ΔΙΕΘΝΕΣ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΤΗΣ ΕΛΛΑΔΟΣ

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

HTML Scrapping με χρήση .NET

Του φοιτητη
Κωνσταντίνου Γεωργιάδη
Αρ. Μητρώου: 144164

Επιβλέπων
Ονοματεπώνυμο Μιχαήλ
Σαλαμπασης
Βαθμίδα ……

Ημερομηνία 18-03-2020

Τίτλος Δ.Ε. HTML Scrapping με χρήση .NET
Κωδικός Δ.Ε. 20108
Ονοματεπώνυμο φοιτητή/τών Κωνσταντίνος Γεωργιάδης
Ονοματεπώνυμο εισηγητή Μιχαήλ Σαλαμπασης
Ημερομηνία ανάληψης Δ.Ε. 18-03-2020
Ημερομηνία περάτωσης Δ.Ε. …

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως διπλωματική εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η έγκριση της διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραιτήτως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

Αφιέρωση

Αφιερώνω την πτυχιακή μου εργασία στους γονείς μου και σε όσους ήταν κοντά μου , για την ηθική και οικονομική υποστήριξη που μου παρείχαν κατά την διάρκεια των σπουδών μου.Τους είμαι για πάντα ευγνώμων

Πρόλογος

Η παρούσα πτυχιακή εργασία με τίτλο «HTML Scraping με χρήση .NET» εκπονήθηκε στα πλαίσια της ολοκλήρωσης των προϋποθέσεων, για τη λήψη του πτυχίου μου από το Δ.Ι.Π.Α.Ε Θεσσαλονίκης τμήμα Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων.Ο λόγος που επέλεξα να ασχοληθώ με την συγκεκριμένη πτυχιακή είναι η απόκτηση γνώσεων πάνω στο αντικείμενο Web Scraping , καθώς αποτελεί χρήσιμη και εύκολη μέθοδος άντλησης δεδομένων από σελίδες του διαδικτύου.Εκτός αυτού το θέμα της πτυχιακής είναι επίκαιρο και θα παραμένει για αρκετό καιρό.Η πτυχιακή δεν αποκλείεται στο μέλλον να επεκταθεί και σε άλλες σελίδες , προσφέροντας ακόμα μεγαλύτερη συλλογή πληροφοριών.Υπάρχουν εταιρείες που προσλαμβάνουν προγραμματιστές ειδικά για αυτό τον τομέα , οπότε η πτυχιακή αυτή ίσως είναι μία ευκαιρία , σε ότι αφορά έναν από τους κλάδους που μπορώ να ακολουθήσω στο μέλλον.

Περίληψη

Ο στόχος της πτυχιακής εργασίας είναι η άντληση πληροφοριών, συγκεκριμένα θεατρικών παραστάσεων . Αφού αντλήσουμε την πληροφορία αυτή , την επεξεργαζόμαστε και την εισάγουμε στην βάση. Για να γίνει όμως εφικτό, θα πρέπει να χρησιμοποιηθούν οι κατάλληλες βιβλιοθήκες. Από κεί και πέρα ,μπορούμε να χρησιμοποιήσουμε τα δεδομένα αυτά , εξάγοντας τα σε διάφορες μορφές αρχείων.Για λόγους ευχρηστίας , χρησιμοποιείτε online MySQL βάση για την αποθήκευση των πληροφοριών αυτών.Το project υλοποιείτε σε περιβάλλον Visual Studio 2019 με γλώσσα προγραμματισμού την C#.Η εφαρμογή απαιτεί συνεχή σύνδεση με το διαδίκτυο και την online βάση. Αφού σχεδιάστηκε και υλοποιήθηκε , πραγματοποιήθηκαν τα κατάλληλα test για να διαπιστωθούν τα αποτελέσματα της εφαρμογής. Εξετάζοντας τα αποτελέσματα , μπορεί κανείς να προσέξει , ότι στις περισσότερες περιπτώσεις ακολουθώντας τα κύρια πρότυπα των θεατρικών παραστάσεων του ιστότοπου , οι πληροφορίες συλλέγονται και αποθηκεύονται με ακριβή και ορθό τρόπο.

Η πτυχιακή απευθύνεται σε όσους θέλουν να ασχοληθούν με το Web Scraping και τις τεχνικές που σχετίζονται με αυτό. Στα κεφάλαια που θα αναλυθούν παρακάτω , θα εξηγήσω τόσο τα θεωρητικά μέρη τις πτυχιακής ,όσο και το πρακτικό κομμάτι της.Θα δείξω κάθε εργαλείο που χρησιμοποίησα , τον τρόπο σκέψης μου, οτιδήποτε με βοήθησε στην υλοποίηση του. Υπάρχουν αρκετές εταιρείες που προσλαμβάνουν προγραμματιστές για αυτόν τον τομέα, οπότε η παρούσα πτυχιακή μπορεί να είναι ένα μεγάλο βήμα στην καριέρα μου και μια ευκαιρία να μάθω περισσότερα για αυτήν την τεχνική.

HTML Scrapping using .NET

Konstantinos Georgiadis

Abstract

The aim of the dissertation is to extract information about theatrical performances from viva.gr. After extracting this information, we process the data and insert them to the database. In order to make that possible, we need the appropriate libraries provided by the C# framework, which will be explained below. Beyond that, we can use this data by exporting them to various file formats. For ease of use, use an online MySQL database to store this information. The implementation of the project is on Visual Studio 2019 environment with C # programming language. The project is a desktop application that requires connection to the internet, but also to the online database. After the application was designed and implemented, the appropriate tests were performed to determine the results of the application. Examining the results, one can note that in most cases following the main standards of theatrical performances on the site, the information is collected and stored accurately and right way.

This dissertation is about those who want to learn about Web Scraping and similar techniques. The below chapters will analyze the theoritical part and how we will continue to its structure part. I will show every tool that I used,my way of thinking, whatever helped me to accomplish this application.There are several companies that hire developers for this sector, so this project can be a big step to my job career, and an opportunity to learn more about this technique.

Ευχαριστίες

Οφείλω να εκφράσω τις θερμές μου ευχαριστίες, προς τον επιβλέποντα της εργασίας, Καθηγητή κ. Μιχάλη Σαλαμπάση, για την υπόδειξη του θέματος της πτυχιακής μου εργασίας καθώς και για την καθοδήγησή του .

Περιεχόμενα

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

Συντομογραφίες

Δ.Ε.	Διπλωματική Εργασία
ΔΙΠΑΕ	Διεθνές Πανεπιστήμιο Ελλάδος
Π.Ε.	Πτυχιακή Εργασία
WS	Web Scraping
WR	Web Scraper
WD	Web Scraped
WC	Web Crawler
SS	Screen Scraping
HAP	Html-Agility-Pack

Κεφάλαιο 1ο: Web Scraping

1.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα αναλυθεί το θεωρητικό κομμάτι του Web Scraping , τι είναι, πώς λειτουργεί , ποιοί το χρησιμοποιούν , αλλά και ό,τι σχετίζεται με αυτό .Θα γίνουν συγκρίσεις με άλλες παρόμοιες τεχνικές , παρουσίαση χαρακτηριστικών αλλά και πως μπορούμε να το αποτρέψουμε .

1.2 Web Scraping

Ας πούμε ότι θέλετε να αντιγράψετε μία λίστα ταινιών με τα χαρακτηριστικά τους και να τα αποθηκεύσετε σε ένα αρχείο .Οι περρισότεροι θα το κάνουν ένα προς ένα πράγμα που δεν είναι και πολύ πρακτικό αν η λίστα είναι μεγάλη .Εδώ έρχεται η τεχνική web scraping για να λύσει αυτό το πρόβλημα. Το web scraping είναι μια διαδικασία αυτοματοποίησης της εξαγωγής δεδομένων με αποτελεσματικό και γρήγορο τρόπο. Με τη βοήθεια του web scraping, μπορείτε να εξαγάγετε δεδομένα από οποιονδήποτε ιστότοπο, ανεξάρτητα από το πόσο μεγάλα είναι τα δεδομένα, στον υπολογιστή σας. Ορισμένα δεδομένα που είναι διαθέσιμα στον Ιστό παρουσιάζονται σε μορφή που διευκολύνει τη συλλογή και τη χρήση τους κάνοντας το έργο ενός web scraper ακόμα πιο εύκολο.Επιπλέον, οι ιστότοποι ενδέχεται να έχουν δεδομένα τα οποία δεν μπορείτε να αντιγράψετε και να επικολλήσετε. Το web scraping μπορεί να σας βοηθήσει να εξαγάγετε οποιοδήποτε είδος δεδομένων θέλετε.Εάν όμως θέλετε να εξάγετε αυτήν την πληροφορία σε κάποιο είδος αρχείου τι γίνετε σε αυτή την περίπτωση. Αφού έχουμε αντλήσει την πληροφορία έχουμε την δυνατότητα να την εξάγουμε σε διάφορες μορφές αρχείων της επιλογής μας.Οπότε συνοπτικά μπορούμε να πούμε ότι , απλοποιεί τη διαδικασία εξαγωγής δεδομένων, τα επιταχύνει αυτοματοποιώντας τα και δημιουργεί εύκολη πρόσβαση στα απορριφθέντα δεδομένα παρέχοντάς τα σε μορφή συνήθως CSV,Excel , JSON.

Σχήμα 1.1: Web Scraping flow

1.3 Η διαδικασία

Για να κατανοήσετε το Web Scraping, είναι σημαντικό να κατανοήσετε πρώτα ότι οι ιστοσελίδες είναι χτυσμένες με γλώσσες σήμανσης που βασίζονται σε κείμενο(text-based mark-up language), η πιο κοινή είναι η HTML. Μια mark-up γλώσσα καθορίζει τη δομή του περιεχομένου μιας ιστοσελίδας. Δεδομένου ότι υπάρχουν καθολικά συστατικά και ετικέτες των γλωσσών σήμανσης, αυτό καθιστά πολύ πιο εύκολο

για τον Web Scraper να τραβήξει τις πληροφορίες που χρειάζεται. Η ανάλυση μέσω HTML είναι μόνο το 1/2 της δουλειάς του WR. Μετά από αυτό, ο WR εξάγει στη συνέχεια τα απαραίτητα δεδομένα και τα αποθηκεύει. Οι WR είναι παρόμοιοι με τις διασυνδέσεις προγραμματισμού εφαρμογών ή API, οι οποίες επιτρέπουν σε δύο εφαρμογές να αλληλεπιδρούν μεταξύ τους για πρόσβαση σε δεδομένα.Ένα θέμα που δεν μπορούμε να παραλείψουμε είναι ό,τι οι web scrapers απαιτούν γρήγορη σύνδεση με το internet στις περισσότερες περιπτώσεις καθώς είναι σημαντικός ρόλος στην απόδοση τους.Αν το intenet κάποιου που προσπαθεί να αντλήσει δεδομένα είναι αργό , τότε η διαδικασία γίνεται δύσκολη.

Υπάρχουν 5 βήματα που πρέπει να ολοκληρωθούν για να μπορέσει να επιτευχθεί η διαδικασία Web Scraping :

- Πρώτον , πρέπει η ομάδα η ο προγραμματιστής που θέλει να φτιάξει τέτοιου είδους πρόγραμμα , να βρεί και να αποφασίσει τι δεδομένο θέλει να αντλήσει και απο ποιά/ποίες σελίδες του web επιθυμεί να το κάνει.
- Ορίζεται το Url βάση/πηγή στο οποίο μπορούμε να ανιχνεύσουμε τυχόν υποσελίδες που θέλουμε να αντλήσουμε τα δεδομένα.Αυτό σημαίνει ότι πρέπει να υπάρξει ένα συγκεκριμένο έυρος διευθύνσεων που θα γίνεται η διαδικασία scraping.Η διαδικασία scraping διαρκεί μέχρι το έυρος των διευθύνσεων αυτών καλυφθεί.
- Τα δεδομένα ανακτώνται σε μορφή HTML, τά οποία αναλύονται προσεκτικά για να απεγκλωβιστούν τα ανεπεξέργαστα δεδομένα που θέλετε από το θόρυβο που τα περιβάλλει. Ανάλογα με το project, τα δεδομένα μπορεί να είναι τόσο απλά όσο ένα όνομα και μια διεύθυνση σε ορισμένες περιπτώσεις, και τόσο πολύπλοκα όσο τα δεδομένα πρόβλεψης καιρού η κάτι ανάλογο.
- Τα δεδομένα αποθηκεύονται με τη μορφή και τις ακριβείς προδιαγραφές που εσείς επιλέγεται. Συνήθως οι scraping εφαρμογές χρησιμοποιούν βάσεις δεδομένων για να δουν και να χειριστούν τα δεδομένα ,πράγμα που είναι λογικό αν κάτσουμε και σκεφτούμε πόσο μεγάλος μπορεί να είναι ο όγκος αυτός των δεδομενων που αντλήσαμε.
- Τέλος , έχοντας ήδη πρόσβαση στα δεδομένα μέσω της βάσης, μπορούμε να τα εξάγουμε σε διάφορους τύπους αρχείου πχ Json, Excel , Csv .

1.4 Οι κύριες χρήσεις του Web Scraping

Ολόκληρα επιχειρηματικά μοντέλα έχουν επικεντρωθεί γύρω από την πρακτική του WS, και θα συνεχίσουμε να βλέπουμε όλο και περισσότερα παραδείγματα αυτού στο μέλλον. Παρακάτω είναι 5 από τις πιο εξέχουσες εφαρμογές του WS σήμερα :

Παρακολούθηση τιμών : Το WS μπορεί να χρησιμοποιηθεί από τις εταιρείες για την άντληση των δεδομένων των προϊόντων τους και άλλα ανταγωνιστικά προϊόντα, καθώς και για να δούμε πώς επηρεάζει τις στρατηγικές τιμολόγησης τους. Οι εταιρείες μπορούν να χρησιμοποιήσουν αυτά τα δεδομένα για να καθορίσουν τη βέλτιστη τιμολόγηση για τα προϊόντα τους, ώστε να μπορούν να λάβουν τα μέγιστα έσοδα.

- Δυναμική τιμολόγηση και βελτιστοποίηση εσόδων
- Παρακολούθηση ανταγωνιστών
- Παρακολούθηση τάσης προϊόντος
- Λήψη επενδυτικών αποφάσεων

Εναλλακτικά δεδομένα για χρηματοδότηση : Η διαδικασία λήψης αποφάσεων δεν ήταν ποτέ τόσο ενημερωμένη, όσο και τα δεδομένα τόσο διορατικά . Οι κορυφαίες εταιρείες του κόσμου καταναλώνουν όλο και περισσότερο web scraped δεδομένα που αποκόπτονται από τον ιστό, δεδομένης της απίστευτης στρατηγικής τους αξίας

- Εκτίμηση βασικών αρχών της εταιρείας
- Ενοποιήσεις δημόσιου συναισθήματος
- Παρακολούθηση ειδήσεων

Έρευνα Αγοράς : Η έρευνα αγοράς είναι κρίσιμη και θα πρέπει να καθοδηγείται από τις πιο ακριβείς διαθέσιμες πληροφορίες. Υψηλής ποιότητας, υψηλού όγκου, και ιδιαίτερα διορατικά,τα web scraped δεδομένα κάθε σχήματος και μεγέθους τροφοδοτούν την ανάλυση της αγοράς και την επιχειρηματική νοημοσύνη σε όλο τον κόσμο.

- Ανάλυση Τάσεων Αγοράς
- Τιμολόγηση Αγοράς
- Βελτιστοποίηση σημείου εισόδου
- Έρευνα και Ανάπτυξη

Ακίνητα : Ο ψηφιακός μετασχηματισμός της ακίνητης περιουσίας τα τελευταία είκοσι χρόνια απειλεί να διαταράξει τις παραδοσιακές επιχειρήσεις και να δημιουργήσει ισχυρούς νέους παράγοντες στον κλάδο. Με την ενσωμάτωση των web scraped δεδομένων στην καθημερινή επιχείρηση, οι πράκτορες και οι μεσίτες μπορούν να προστατευτούν από τον online ανταγωνισμό και να λάβουν ενημερωμένες αποφάσεις εντός της αγοράς.

- Εκτίμηση αξίας ακινήτου
- Παρακολούθηση ποσοστών κενών θέσεων
- Εκτίμηση των αποδόσεων ενοικίου
- Κατανόηση της κατεύθυνσης της αγοράς

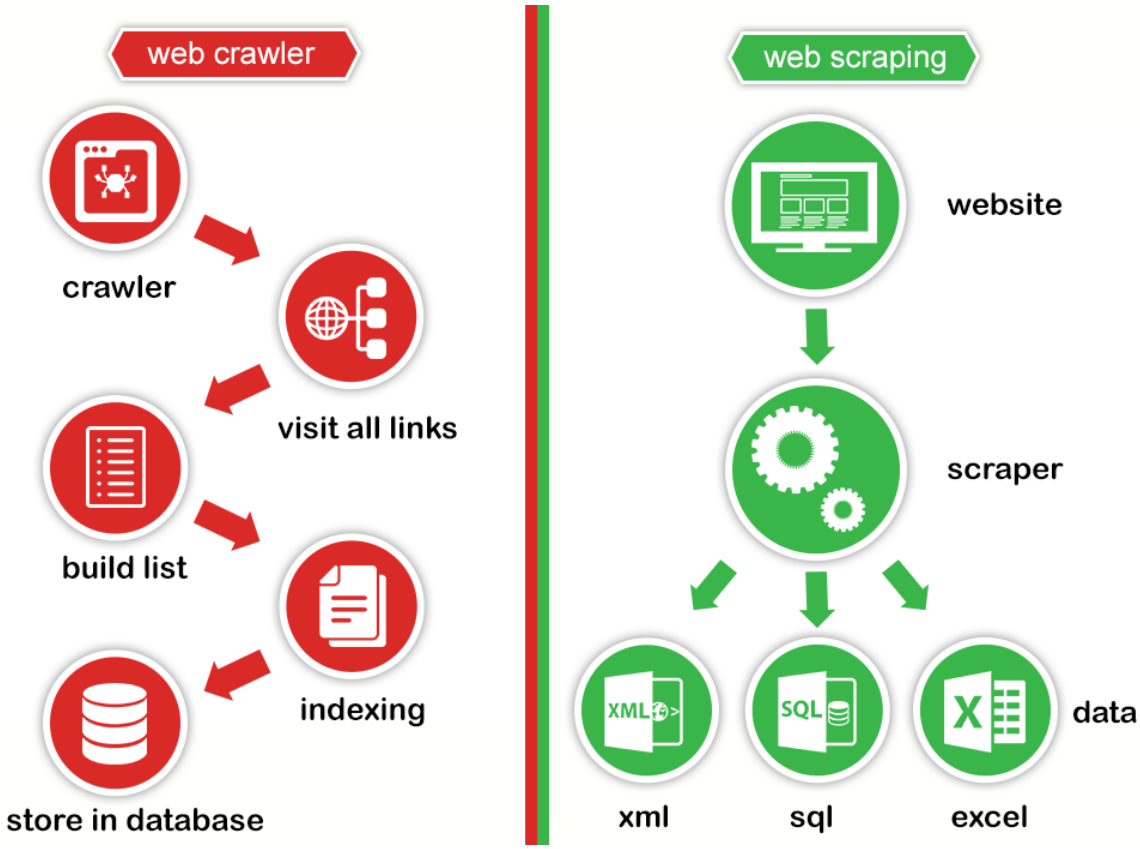
Παρακολούθηση ειδήσεων και περιεχομένου : Τα σύγχρονα μέσα ενημέρωσης μπορούν να δημιουργήσουν εξαιρετική αξία ή υπαρξιακή απειλή για την επιχείρησή σας σε έναν μόνο κύκλο ειδήσεων. Αν είστε μια εταιρεία που εξαρτάται από έγκαιρες αναλύσεις ειδήσεων, ή μια εταιρεία που εμφανίζεται συχνά στις ειδήσεις ,το WS είναι η απόλυτη λύση για την παρακολούθηση, συγκέντρωση και ανάλυση των πιο κρίσιμων ιστοριών από τον κλάδο σας.

- Λήψη επενδυτικών αποφάσεων
- Ηλεκτρονική ανάλυση δημόσιου συναισθήματος
- Παρακολούθηση ανταγωνιστών
- Πολιτικές Εκστρατείες

1.5 Web Scraping vs Web Crawling

Η τεχνική Web Crawling γνωστή ως Ευρετηρίαση, χρησιμοποιείται για την ευρετηρίαση των πληροφοριών σε έναν ιστότοπο χρησιμοποιώντας bots που είναι επίσης γνωστά ως crawlers. Τα Web Crawlers χρησιμοποιούνται από μεγάλες μηχανές αναζήτησης όπως το Google, Bing, YahooTo Bing είναι ένα από τα σημαντικότερα προγράμματα ανίχνευσης ιστού. Εδώ λαμβάνουμε γενικές πληροφορίες, ενώ στην τεχνική WS λαμβάνουμε συγκεκριμένες πληροφορίες.

Ξέροντας απο προηγούμενους ορισμούς τι είναι το WS , καταλήγουμε στο συμπέρασμα ότι η ανίχνευση ιστού απλώς ευρετηριάζει τις πληροφορίες χρησιμοποιώντας bots,ενώ το WS είναι μια αυτοματοποιημένη τεχνική λογισμικού εξαγωγής πληροφοριών από τον ιστό.



Σχήμα 1.2: Web Scraping vs Web Crawling

1.6 Open Data

<<Ανοιχτά είναι εκείνα δεδομένα τα οποία μπορεί ελεύθερα να χρησιμοποιήσει κάποιος, να τα επαναχρησιμοποιήσει και να τα αναδιανέμει από οποιονδήποτε αλλά θα πρέπει να γίνεται αναφορά στους δημιουργούς και να διατίθενται, με τη σειρά τους, υπό τους ίδιους όρους.Τα ανοιχτά δεδομένα είναι:

- Διαθεσιμότητα και Προσβασιμότητα: Τα δεδομένα πρέπει να είναι διαθέσιμα αυτούσια, να έχουν ένα λογικό κόστος αναπαραγωγής, και κατά προτίμηση να είναι διαθέσιμα για λήψη από το Διαδίκτυο. Επίσης, πρέπει να είναι διαθέσιμα σε κάποια μορφή πρακτικά αναγνώσιμη.

- Επαναχρησιμοποίηση και Αναδιανομή: Τα δεδομένα θα πρέπει να είναι διαθέσιμα υπό όρους που επιτρέπουν την επαναχρησιμοποίηση και την αναδιανομή τους, συμπεριλαμβανομένης και της ανάμειξης με άλλα σύνολα δεδομένων.

- Καθολική Συμμετοχή: Καθένας πρέπει να μπορεί να χρησιμοποιήσει, να επαναχρησιμοποιήσει και να αναδιανείμει τα δεδομένα. Δεν πρέπει αυτά να υπόκεινται σε διακρίσεις με βάση τον τομέα δραστηριότητας ή τα πρόσωπα και τις ομάδες. Για παράδειγμα, περιορισμοί για «μη-εμπορική χρήση» ή περιορισμοί για χρήση μόνο για συγκεκριμένους σκοπούς (π.χ. μόνο στην εκπαίδευση) δεν είναι επιτρεπτοί.

Αν κάποιος αναρωτιέται γιατί είναι τόσο σημαντικό να είναι σαφές τι σημαίνει Ανοιχτά Δεδομένα και σε τι είναι χρήσιμος αυτός ο ορισμός, υπάρχει μια απλή απάντηση: η διαλειτουργικότητα.

Η διαλειτουργικότητα δηλώνει τη δυνατότητα διαφορετικών συστημάτων να λειτουργούν μαζί . Σε αυτή τη συγκεκριμένη περίπτωση, γίνεται αναφορά στη δυνατότητα να διαλειτουργούν ή να αναμειγνύουν διαφορετικά σύνολα δεδομένων>>

1.7 Μέθοδοι αποτροπής

Όριο τιμών μεμονωμένων διευθύνσεων IP : Εάν λαμβάνετε χιλιάδες αιτήματα από έναν υπολογιστή, υπάρχει μεγάλη πιθανότητα το άτομο πίσω από αυτόν να υποβάλλει αυτοματοποιημένα αιτήματα στον ιστότοπό σας.Ο αποκλεισμός των αιτήσεων από τους υπολογιστές που τους κάνουν πάρα πολύ γρήγορα είναι συνήθως ένα από τα πρώτα στοιχεία μέτρα που θα χρησιμοποιήσουν για να σταματήσουν τους WR.

Περιορισμός δυνατοτήτων χρήστη : Επιτρέψτε στους χρήστες να κάνουν περιορισμένο αριθμό κινήσεων σε ένα συγκεκριμένο χρονικό διάστημα, τσεκάροντας το απο την IP του.Αυτό έχει ώς αποτέλεσμα να αργούν οι WR μειώνοντας την απόδοση τους.

Αλλαγή HTML κώδικα της ιστοσελίδας τακτικά : Οι Scrapers βασίζονται στην εύρεση μοτίβων με σήμανση HTML μιας ιστοσελίδας και στη συνέχεια χρησιμοποιούν αυτά τα μοτίβα ως ενδείξεις για να βοηθήσουν τα σενάριά τους να βρουν τα σωστά δεδομένα σε κάποιον ιστότοπο.Εάν η σήμανση του ιστότοπού σας αλλάζει συχνά ή είναι εντελώς ασυνεπής, τότε ίσως μπορέσετε να δυσκολέψετε τον Scraper αρκετά ώστε να εγκαταλείψει.

Απαίτηση σύνδεσης του χρήστη με λογαριασμό :Προκειμένου ο scraper να αντλήσει δεδομένα , θα πρέπει να δημιουργηθεί λογαριασμός για να υπάρχει πρόσβαση στην σελίδα.Έτσι λοιπόν ,κάθε φορά που θα στέλνει τα στοιχεία ταυτοποίησης ο scraper, θα φαίνεται ποιός σύνδεεται και τι λειτουργίες κάνει στο site . Αυτή η μέθοδος δεν αποτρέπει το scraping , αλλά δίνει μία εικόνα για το ποίος το κάνει.

Χρήση CAPTCHA όταν είναι απαραίτητο : Οι CAPTCHA σχεδιάζονται ειδικά για να χωρίσουν τους ανθρώπους από τους υπολογιστές με την παρουσίαση των προβλημάτων που οι άνθρωποι βρίσκουν γενικά εύκολα, αλλά οι υπολογιστές δυσκολότερα με βάση το χρόνο.Ενώ οι άνθρωποι τείνουν να βρουν τα προβλήματα εύκολα, τείνουν επίσης να τα βρίσκουν εξαιρετικά ενοχλητικά.Τα Captchas μπορεί να είναι χρήσιμα, αλλά θα πρέπει να χρησιμοποιείται με μέτρο.

Μην δημοσιεύετε τις πληροφορίες στον ιστότοπό σας : Αυτό μπορεί να φαίνεται προφανές, αλλά είναι σίγουρα μια επιλογή, αν ανησυχείτε πως ένας Scraper κλέβει τις πληροφορίες σας. Χωρίς αμφισβήτηση, το WS είναι απλά ένας τρόπος για να αυτοματοποιήσει την πρόσβαση σε μια ιστοσελίδα. Αν νιώθετε καλά στο να μοιράζονται το περιεχόμενό σας με όποιον επισκέπτεται το site σας, τότε ίσως δεν χρειάζεται να ανησυχείται για τους Scrapers.

Το Web Scraping διεθνώς μέχρι και σήμερα δεν έχει χαρακτηριστεί παράνομο , ωστόσο η απόσπαση πληροφοριών από ιστότοπους τρίτων δεν μπορεί να γίνεται ανεξέλεγκτα. Οποιοσδήποτε επιχειρεί πρόσβαση και απόσπαση δεδομένων θα πρέπει, αρχικά, να ακολουθεί και να συμμορφώνεται με τους «όρους χρήσης» των ιστοτόπων αυτών. Η απόσπαση δεδομένων κατά παράβλεψη των παραπάνω κανόνων, ή ακόμα και το WS σε ιστότοπο που δεν έχει αναρτήσει όρους χρήσης ή δεν περιλαμβάνει σε αυτούς περιορισμούς σχετικά με τη συγκομιδή, εκθέτει τον αποσπώντα σε σοβαρούς κινδύνους.

Μετά από όλα, αν προσέξει κανείς προσεκτικά , το Google είναι ο μεγαλύτερος WS στον κόσμο και οι άνθρωποι δεν φαίνεται να ενοχλούνται όταν το Google ευρετηριάζει το περιεχόμενό τους. Οποιαδήποτε

βήματα που παίρνετε για να περιορίσετε τους WS θα βλάψουν πιθανώς επίσης την εμπειρία του μέσου θεατή του ιστότοπου. Οτι απόφαση πάρετε ,στόχος είναι η καλύτερη εμπειρία του χρήστη στον ιστότοπο σας.

1.8 Screen Scraping

Το screen scraping είναι η αυτοματοποιημένη η προγραμματιστική χρήση ενός ιστότοπου αντιγράφει πληροφορίες που εμφανίζονται σε ψηφιακή οθόνη, ώστε να μπορούν να χρησιμοποιηθούν για άλλο οποιαδήποτε σκοπό. Τα οπτικά δεδομένα μπορούν να συλλεχθούν ως ανεπεξέργαστο κείμενο από στοιχεία που εμφανίζονται στην οθόνη, όπως ένα κείμενο ή εικόνες που εμφανίζονται στην επιφάνεια εργασίας, σε μια εφαρμογή ή σε έναν ιστότοπο. Το screen scraping μπορεί να εκτελεστεί αυτόματα με ένα πρόγραμμα scraping ή χειροκίνητα με μεμονωμένα δεδομένα εξαγωγής.

Το screen scraping έχει διάφορες χρήσεις, τόσο ηθικές όσο και ανήθικες. Σύντομα παραδείγματα και των δύο περιλαμβάνουν είτε μια εφαρμογή για τραπεζικές συναλλαγές, για τη συλλογή δεδομένων από πολλούς λογαριασμούς για έναν χρήστη ή για την κλοπή δεδομένων από εφαρμογές. Ένας προγραμματιστής μπορεί να μπεί στον πειρασμό να κλέψει κώδικα από άλλη εφαρμογή για να κάνει τη διαδικασία ανάπτυξης πιο γρήγορη και ευκολότερη για τον εαυτό του.

1.8.1 Σε τι χρησιμεύει όμως;

Οι Screen Scrapers έχουν εφαρμοστεί σε μεγάλο αριθμό πεδίων για διάφορες περιπτώσεις χρήσης. Ορισμένες πιθανές χρήσεις περιλαμβάνουν:

- τραπεζικές εφαρμογές και χρηματοοικονομικές συναλλαγές ·
- αποθήκευση σημαντικών δεδομένων για μελλοντική χρήση.
- για την εκτέλεση ενεργειών που θα έκανε ένας χρήστης σε έναν ιστότοπο.
- να μεταφράσει δεδομένα από μια εφαρμογή παλαιού τύπου σε μια σύγχρονη εφαρμογή.
- για συλλέκτες δεδομένων, όπως ιστότοπους σύγκρισης τιμών ·
- να παρακολουθείτε προφίλ χρηστών για να δείτε διαδικτυακές δραστηριότητες
- για κλοπή δεδομένων.

Μία από τις μεγαλύτερες περιπτώσεις χρήσης του ήταν στον τραπεζικό τομέα. Οι δανειστές μπορεί να θέλουν να χρησιμοποιήσουν το SS για τη συλλογή οικονομικών δεδομένων ενός πελάτη. Οι εφαρμογές που βασίζονται σε οικονομικά στοιχεία μπορούν να χρησιμοποιήσουν το SS για πρόσβαση σε πολλούς λογαριασμούς από έναν χρήστη, συγκεντρώνοντας όλες τις πληροφορίες σε ένα μέρος. Ωστόσο, οι χρήστες θα πρέπει να εμπιστεύονται ρητά την εφαρμογή, καθώς εμπιστεύονται αυτόν τον οργανισμό με τους λογαριασμούς, τα δεδομένα των πελατών και τους κωδικούς πρόσβασης.

Ένας οργανισμός μπορεί επίσης να θέλει να χρησιμοποιήσει το SS για να μεταφράσει μεταξύ προγραμμάτων εφαρμογών παλαιού τύπου και νέων διεπαφών χρήστη (UI), έτσι ώστε η λογική και τα δεδομένα που σχετίζονται με τα παλαιά προγράμματα να μπορούν να συνεχίσουν να χρησιμοποιούνται. Αυτή η επιλογή χρησιμοποιείται σπάνια και θεωρείται μόνο ως επιλογή όταν άλλες μέθοδοι δεν είναι πρακτικές.

1.8.2 Πώς να αποτρέψετε το Screen Scraping

Δυστυχώς, δεν υπάρχει κανένας τρόπος να αποφευχθεί το screen scraping. Ωστόσο, υπάρχουν τρόποι για να το εντοπίσετε . Ένας οργανισμός μπορεί να ανιχνεύσει την χρήση SS μέσω ορισμένων υπογραφών ή να χρησιμοποιήσει συμπεριφορές. Για παράδειγμα, αν εντοπιστεί μια μη τυπική

κατηγορία χρήστη, εάν η JavaScript δεν εκτελέσει την πλευρά του πελάτη ή έχουν δημιουργηθεί αρκετές ακολουθίες αιτήματος σελίδας, μπορεί να είναι ένα σημάδι screen scraping.

Άλλοι μέθοδοι ανίχνευσης :

- Χρησιμοποιήστε κωδικούς πρόσβασης μίας χρήσης, επειδή οι screen scrapers δεν θα μπορούν να δουν έναν κωδικό πρόσβασης έως ότου χρησιμοποιηθεί.

- Μπορείτε να χρησιμοποιήσετε εργαλεία προστασίας τα οποία μπορούν να βοηθήσουν στον εντοπισμό περίεργων ενεργειών στον ιστότοπο η στην διαδικτυακή εφαρμογή.

- Εκτελέστε λογισμικό εντοπισμού απάτης για να πιάσετε πιθανό screenshot ενώ συμβαίνει

- Δημιουργία σελίδων παγιδών οι οποίες δεν είναι ορατές στον χρήστη αλλά στα bots που ψάχνουν όλες τις σελίδες ενός ιστότοπου για να τραβήξουν πληροφορίες.Αυτός ο μηχανισμός αναγνωρίζει την IP του bot και έτσι μπορεί να την μπλοκάρει η να βρεί από που έρχονται οι επιθέσεις αυτές.

1.9 Επίλογος κεφαλαίου

Πήραμε μία γέυση για το τι έστι Web Scraping , είδαμε κάποιες απο τις κύριες χρήσεις του ,και άλλες παρόμοιες τεχνικές που χρησιμοποιούνται σήμερα στο διαδίκτυο.Το πρακτικό κομμάτι θα συνεχιστεί στα επόμενα κεφάλαια.

Κεφάλαιο 2ο: NuGet Scraping Libraries

2.1 Εισαγωγή

Στο παρόν κεφάλαιο γίνεται εκτενής ανάλυση των libraries που χρησιμοποιήθηκαν για την τεχνική Web Scraping.Θα γίνει ανάλυση των selector για τον εντοπισμό των επιθυμητών στοιχείων του ιστότοπου στόχου και πολλά παραδείγματα σχετικά με αναζήτηση τους , με διαφορετικούς τρόπους.

2.2 Τί είναι

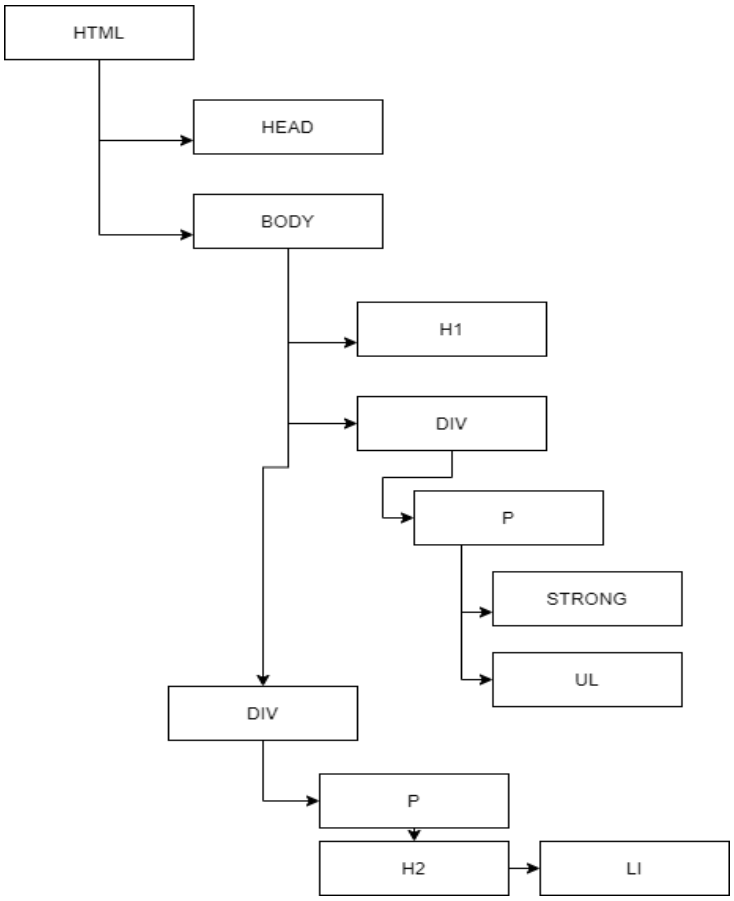
Είναι ένας μηχανισμός μέσω του οποίου οι υπεύθυνοι για την ανάπτυξη μπορούν να δημιουργήσουν, να μοιραστούν, και να καταναλώσουν τον χρήσιμο κώδικα. Για το .NET, ο μηχανισμός που υποστηρίζεται από τη Microsoft για την κοινή χρήση κώδικα είναι το NuGet, το οποίο καθορίζει τον τρόπο δημιουργίας, φιλοξενίας και κατανάλωσης των πακέτων για το .NET και παρέχει τα εργαλεία για κάθε έναν από αυτούς τους ρόλους.

Με απλά λόγια, ένα πακέτο NuGet είναι ένα ενιαίο αρχείο ZIP με την επέκταση .nupkg που περιέχει μεταγλωττισμένο κώδικα (DLL), άλλα αρχεία που σχετίζονται με αυτόν τον κώδικα, και μια περιγραφική διακήρυξη που περιλαμβάνει πληροφορίες όπως τον αριθμό έκδοσης του πακέτου. Προγραμματιστές με κώδικα για κοινή χρήση δημιουργούν πακέτα και τα δημοσιεύουν σε δημόσιο ή ιδιωτικό κεντρικό υπολογιστή. Οι καταναλωτές πακέτων λαμβάνουν αυτά τα πακέτα από κατάλληλους κεντρικούς υπολογιστές, τα προσθέτουν στα έργα τους και, στη συνέχεια, καλούν τη λειτουργικότητα ενός πακέτου στον κωδικό έργου τους. Στη συνέχεια, το ίδιο το NuGet χειρίζεται όλες τις ενδιάμεσες λεπτομέρειες.Επειδή το NuGet υποστηρίζει ιδιωτικούς κεντρικούς υπολογιστές δίπλα στον κεντρικό υπολογιστή nuget.org, μπορείτε να χρησιμοποιήσετε πακέτα NuGet για να κάνετε κοινή χρήση κώδικα που είναι αποκλειστικά για έναν οργανισμό ή μια ομάδα εργασίας. Μπορείτε επίσης να χρησιμοποιήσετε τα πακέτα NuGet ως έναν βολικό τρόπο παράγοντας το δικό σας κωδικό για χρήση. Εν ολίγοις, ένα πακέτο NuGet είναι μια shareable μονάδα κώδικα, αλλά δεν απαιτεί ούτε συνεπάγεται κανένα συγκεκριμένο μέσο κοινής χρήσης.

2.3 Τί πρέπει να έχετε υπόψιν πρίν κάνετε Scraping

Καθώς εξετάζουμε τη δομή της σελίδας θα δούμε ότι κάθε στοιχείο σχετίζεται με ένα άλλο στοιχείο. Αυτό ονομάζεται σχέση γονέα-παιδιού-αδελφού. Ένα στοιχείο που βρίσκεται ακριβώς πάνω από ένα άλλο στοιχείο στην ιεραρχία ονομάζεται γονέας του στοιχείου κάτω από αυτό. Το στοιχείο κάτω από τον γονέα ονομάζεται παιδί. Όταν δύο στοιχεία είναι ίσα στην ιεραρχία, είναι γνωστά ως αδέλφια.

Έτσι λοιπόν , όταν θέλετε να κάνετε Scraping δεδομένα απο μία σελίδα , θα κοιτάξετε τι δομή έχουν οι υποσελίδες της ώστε να ξέρετε περίπου την ιεραρχία της.Μπορώ να πω οτι ακόμα και αν δεν έχετε ιδιαίτερες γνώσεις πάνω στην HTML , δεν θεωρώ δύσκολο να το καταλάβει κάποιος. Η ιεραρχία του διαγράμματος παρακάτω εμφανίζεται απο τα δεξιά στα αριστερά.



Σχήμα 2.1:Ιεραρχία ενός HTML εγγράφου

2.4 HAP

Το HAP είναι ένας HTML αναλυτής γραμμένος σε C# για να δίαβάζει/γράφει αντικείμενα που υπάρχουν σε έναν ιστότοπο και υποστηρίζει γλώσσες εύρεσης των στοιχείων αυτών όπως η XPath και η XSLT. Στο project μου , χρησιμοποιήθηκε η XPath που θα εξηγηθεί παρακάτω με ένα ακόμα παρόμοιο εργαλεί εύρεσης αντικειμένων.

Αν κάποιος θέλει να μάθει το HAP μπορεί άνετα να μπεί στο html-agility-pack.net .Το site περιέχει τις απαραίτητες πληροφορίες που χρειάζεται για να ξεκινήσει το scraping καθώς εκτός από οδηγίες παρέχει μια ατελείωτη λίστα με παραδείγματα κώδικα. Προσωπικά με βοήθησε αρκετά στην εύρεση αντικειμένων μέσα στον ιστότοπο των θεατρικών παραστάσεων.Για να το εγκαταστήσετε αυτό και άλλες βιβλιοθήκες στο project σας θα πατήσετε δεξί κλικ στο όνομα του project στο solution folder του Visual Studio , και στην συνέχεια επιλέγετε <Manage NuGet Packages>.

Όλα ξεκινάνε απο τον HTML Parser .Απο εκεί διαβάζει τον HTML κώδικα και έτσι έχουμε πρόσβαση σε όλα τα περιεχόμενα του ιστότοπου.Αυτό μπορεί να γίνει απο :

- Αρχείο
- String κώδικα που περιέχει HTML
- Το διαδίκτυο
- Ενα πρόγραμμα περιήγησης

Εδώ και ένα απόσπασμα κώδικα με 3 απο τις περιπτώσεις που αναφέρθηκαν :

```
var doc = new HtmlDocument();
doc.Load(filePath);
// Απο συμβολοσειρα
var doc = new HtmlDocument();
doc.LoadHtml(html);
// Απο το διαδίκτυο
var url = "http://html-agility-pack.net/";
var web = new HtmlWeb();
var doc = web.Load(url);
```

Στην συνέχεια έχουμε τους Selectors που παίζουν βασικό ρόλο στο concept καθώς με αυτούς μπορούμε να αντλήσουμε ότι πληροφορία υπάρχει , βρίσκοντας το μονοπάτι του ,είτε παίρνοντας τον κόμβο η ολόκληρη λίστα,είτε παίρνοντας τα παιδιά του κόμβου κλπ.Οι HTML Selectors που επιτρέπονται στο HAP είναι :

- XPATH
- CSS (Coming Soon)
- XDocument
- LINQ
- CSSSelector

Όταν κάνουμε αναζήτηση ενός στοιχείου μέσα σε έναν ιστότοπο , υπάρχει περίπτωση να κάνουμε κάτι λάθος στον selector , η όντως να μην υπάρχει. Αυτό προκαλεί σφάλμα το οποίο οδηγεί σε τερματισμό του προγράμματος . Γι αυτό τον λόγο , το HAP για να κάνουμε safe άντληση στοιχείων , μας παρέχει έναν εύκολο τρόπο ελέγχου αν υπάρχει το στοιχείο η όχι. Αυτός ο τρόπος είναι να συμπληρώσουμε ενα ? πρίν απο την τελεία , σε κάθε τελεία. Παρακάτω το παράδειγμα:

var duration= doc?.DocumentNode?.SelectSingleNode(“//li[@class=’ui-duration’]”);

Η μέθοδος για επιλογή πολλών κόμβων σύμφωνα με τις παραμέτρους που έχουμε δώσει είναι η SelectNodes() .Αντίστοιχα η μέθοδος για να πάρουμε έναν μοναδικό κόμβο , η εναλλακτικά να τον επιλέξουμε σαν κόμβο πατέρα για να πάρουμε τα παιδιά του είναι η μέθοδος SelectNode().Οι επιλογείς αντικειμένων λόγω τις διαφορετικότητας τους παρέχουν διαφορετικές μεθόδους σε κάθε περίπτωση.Γι’αυτό το λόγο θα πρέπει να μάθουμε να τους ξεχωρίζουμε.

Εδώ και ένα απόσπασμα κώδικα με 2 απο τις περιπτώσεις που αναφέρθηκαν :

```
var value = doc.DocumentNode
             .SelectNodes("//html/input")
             .First()
             .Attributes["src"].Value;

            // Με LINQ
var nodes = doc.DocumentNode.Descendants("input")
             .Select(y => y.Descendants()
             .Where(x => x.Attributes["class"].Value == "amaxi"))
           .ToList();
```

Αφού αντλήσουμε το αντικείμενο , θα πρέπει να έχουμε την δυνατότητα να το χειριστούμε . Το HAP μας δίνει αυτή την δυνατότητα με τις εξής ιδιότητες :

- **InnerHtml** : Παίρνει ή ορίζει τον HTML κώδικα μεταξύ των ετικετών έναρξης και τέλους του αντικειμένου.
- **InnerText** : Παίρνει το κείμενο μεταξύ των ετικετών έναρξης και τέλους του αντικειμένου.
- **OuterHtml :** Παίρνει το αντικείμενο και το περιεχόμενό του σε HTML.
- **ParentNode** : Παίρνει τον γονέα ενός κόμβου (για κόμβους που μπορούν να έχουν γονείς).

Επίσης μας δίνει μεθόδους από τους οποίους μπορούμε να κάνουμε διάφορες λειτουργίες με τους κόμβους , να τους πειράξουμε , να προσθέσουμε , να αφαιρέσουμε κλπ. Εδώ αναφέρονται ορισμένες από αυτές :

- AppendChild() : Προσθέτει τον καθορισμένο κόμβο στο τέλος της λίστας των παιδιών αυτού του κόμβου.
- Clone() : Δημιουργεί ένα αντίγραφο του κόμβου
- CloneNode(String) : Δημιουργεί ένα αντίγραφο του κόμβου και αλλάζει το όνομά του ταυτόχρονα.
- InsertAfter():Εισάγει τον καθορισμένο κόμβο αμέσως μετά τον καθορισμένο κόμβο αναφοράς.
- InsertBefore():Εισάγει τον καθορισμένο κόμβο αμέσως πριν από τον καθορισμένο κόμβο αναφοράς.
- PrependChild : Προσθέτει τον καθορισμένο κόμβο στην αρχή της λίστας των παιδιών αυτού του κόμβου.
- Remove : Καταργεί τον κόμβο από τη συλλογή γονέα.
- RemoveAll : Καταργεί όλα τα παιδιά και/ή τα χαρακτηριστικά του τρέχοντος κόμβου.
- RemoveChild(HtmlNode) : Καταργεί τον καθορισμένο κόμβο παιδί.
- ReplaceChild() : Αντικαθιστά τον παλιό κόμβο παιδί με νέο κόμβο παιδί.

Εδώ και ένα απόσπασμα κώδικα με 2 απο τος ιδιότητες που αναφέρθηκαν :

```
var doc = new HtmlDocument();
doc.LoadHtml(html);
var innerHtml = doc.DocumentNode.InnerHtml;
var innerText = doc.DocumentNode.InnerText;
```

Το HAP μας επιτρέπει να διασχίζουμε στους κόμβους HTML παρέχοντας τις εξής ιδιότητες:

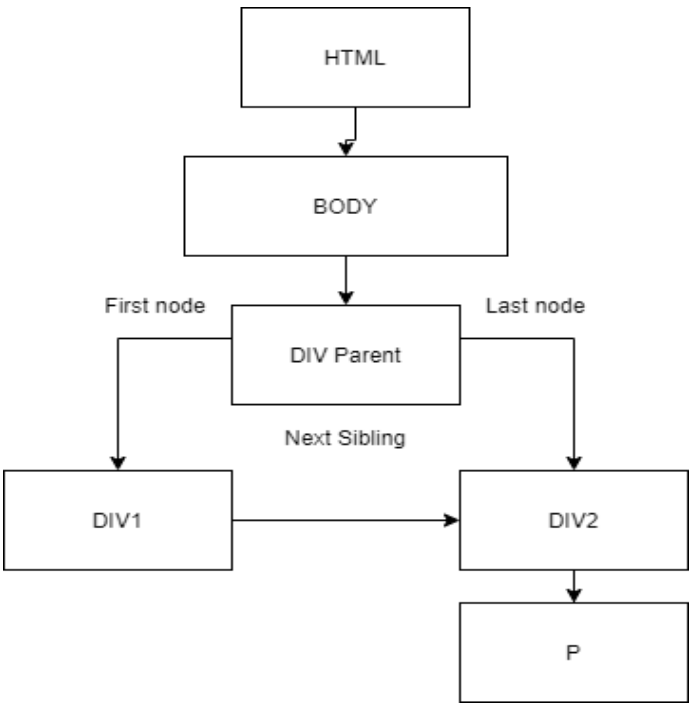
- ChildNodes : Παίρνει όλα τα παιδιά του κόμβου.
- FirstChild : Παίρνει το πρώτο παιδί του κόμβου.
- LastChild : Παίρνει το τελευταίο παιδί του κόμβου.
- NextSibling : Παίρνει τον κόμβο HTML αμέσως μετά από αυτό το στοιχείο.
- ParentNode : Παίρνει τον γονέα αυτού του κόμβου (για κόμβους που μπορούν να έχουν γονείς).

```
var doc = new HtmlDocument();
htmlDoc.LoadHtml(html);
var nodes = doc.DocumentNode.Descendants("input");
```

Μαζί με τις ιδιότητες περιήγησης υπάρχουν και οι μέθοδοι τους:

- Ancestors() : Παίρνει όλους τους προγόνους του κόμβου.
- AncestorsAndSelf() : Παίρνει όλους τους κόμβους προγόνου και τον τρέχοντα κόμβο.
- DescendantNodes : Παίρνει όλους τους κόμβους απογόνων για τον κόμβο και κάθε έναν από τους κόμβους παιδιά.
- Descendants() : Παίρνει όλους τους κόμβους απογόνων σε απαριθμημένη λίστα
- Elements : Παίρνει αντίστοιχους θυγατρικούς κόμβους πρώτης γενιάς που αντιστοιχούν στο όνομα.

Σχήμα 2.2: Σχέσεις γονέα παιδιού



Τέλος , το HAP δίνει την δυνατότητα να πειράξουμε-διαχειριστούμε τα χαρακτηριστικά των στοιχείων.Αυτό γίνεται με τις εξής μεθόδους :

- Add(HtmlAttribute) : Προσθέτει το χαρακτηριστικό σαν αντικείμενο στη συλλογή
- Add(String, String): Προσθέτει ένα νέο χαρακτηριστικό στη συλλογή με τις υπάρχουσες τιμές
- Append(String):Δημιουργεί και εισάγει ένα νέο χαρακτηριστικό ως το τελευταίο χαρακτηριστικό στη συλλογή.
- Remove() : Καταργεί όλα τα χαρακτηριστικά από τη συλλογή
- Remove(HtmlAttribute) : Καταργεί ένα χαρακτηριστικό ίσο με την τιμή της παραμέτρου από τη λίστα
- SetAttributeValue() : Βοηθητική μέθοδος για τον ορισμό της τιμής ενός χαρακτηριστικού αυτού του κόμβου. Εάν το χαρακτηριστικό δεν βρεθεί, θα δημιουργηθεί αυτόματα.

Με την εμπειρία που απέκτησα χρησιμοποιώντας το HAP , μπορώ να πω με σιγουριά πως αν κάποιος θέλει να ξεκινήσει τα βασικά του WS είναι το κατάλληλο για να το κάνει. Υπάρχει όμως ένα μεγάλο μειονέκτημα , το HAP δεν μπορεί να αντλήσει πληροφορίες που παράγονται απο Javascript η Ajax.Τη λύση για το πως θα πάρουμε δεδομένα παραγόμενα από τα αναφερόμενα θα την δείτε αμέσως παρακάτω.

2.5 Selenium

2.6 Τί είναι

Το Selenium είναι ένα σύνολο διαφορετικών εργαλείων λογισμικού το καθένα με διαφορετική προσέγγιση για την υποστήριξη του αυτοματισμού του προγράμματος περιήγησης. Αυτά τα εργαλεία είναι εξαιρετικά ευέλικτα, επιτρέποντας πολλές επιλογές για τον εντοπισμό και τον χειρισμό στοιχείων μέσα σε ένα πρόγραμμα περιήγησης και ένα από τα βασικά χαρακτηριστικά του είναι η υποστήριξη για την αυτοματοποίηση πολλαπλών πλατφόρμων προγράμματος περιήγησης. Το πλεονέκτημα του σε σχέση με το HAP είναι ότι μπορείς να κάνεις πολλά περισσότερα πράγματα αυτόματα , χωρίς να χρειαστεί παρέμβαση την ώρα που εκτελείτε το πρόγραμμα περιήγησης πχ μπορείτε να συνδεθείτε μέσω του web browser του selenium στο email σας , έχοντας βέβαια βάλει τα στοιχεία σας στα πεδία που απαιτούνται.Το μόνο μειονέκτημα είναι οτι η λειτουργία του απαιτεί περισσότερο χρόνο επεξεργασίας, αλλά αυτό είναι φυσιολογικό βλέποντας τις επιπλέον δυνατότητες που μπορεί να μας προσφέρει.

Παρέχει επίσης μια δοκιμαστική γλώσσα για συγκεκριμένους τομείς (Selenium IDE) για τη σύνταξη δοκιμών σε διάφορες δημοφιλείς γλώσσες προγραμματισμού, όπως C#, Groovy, Java, Perl, PHP, Python, Ruby και Scala. Οι δοκιμές μπορούν στη συνέχεια να εκτελεστούν στα πιο σύγχρονα προγράμματα περιήγησης ιστού. Το Selenium εκτελείται σε Windows, Linux και macOS. Είναι λογισμικό ανοιχτού κώδικα που κυκλοφόρησε με το Apache License 2.0.

2.7 Ποιοί το χρησιμοποιούν

Πολλές από τις πιο δημοφιλής εταιρείες στον κόσμο έχουν υιοθετήσει το Selenium για browser-based δοκιμές τους, αντικαθιστώντας χρόνια προσπάθειας χρησιμοποιώντας άλλα ιδιόκτητα εργαλεία. Όσο έχει αυξηθεί η δημοτικότητά του, τόσο πολλαπλασιάστηκαν οι απαιτήσεις και οι προκλήσεις του. Καθώς ο ιστός γίνεται πιο περίπλοκος και προστίθενται νέες τεχνολογίες σε ιστότοπους, στόχος του είναι να ακολουθεί τα σύχρονα πρότυπα και εξελίσσεται ώστε να βοηθάει τους χρήστες εκεί που είναι δυνατόν.

2.8 Web Driver

Το WebDriver τρέχει ένα πρόγραμμα περιήγησης, όπως ένας χρήστης, είτε τοπικά είτε σε απομακρυσμένο μηχάνημα χρησιμοποιώντας το διακομιστή Selenium, σηματοδοτώντας ένα άλμα προς τα μπροστά σε ότι αφορά τον αυτοματισμό των προγραμμάτων περιήγησης.Το WebDriver υποστηρίζει τους πιο διάσημους φυλλομετρητές του κόσμου , αυτοί είναι :

- Chromium/Chrome
- Firefox
- Edge
- Internet Explorer
- Safari
- Opera

2.9 Εντοπισμός στοιχείων

Μία από τις πιο θεμελιώδεις τεχνικές που πρέπει να μάθετε όταν χρησιμοποιείτε το WebDriver είναι ο τρόπος εύρεσης στοιχείων στη σελίδα. Το WebDriver προσφέρει έναν αριθμό ενσωματωμένων τύπων επιλογής, μεταξύ των οποίων βρίσκουν ένα στοιχείο από το χαρακτηριστικό ID:

IWebElement cheese = driver.FindElement(By.Id("cheese"));

Όπως φαίνεται στο παράδειγμα, ο εντοπισμός στοιχείων στο WebDriver γίνεται στο αντικείμενο cheese.Το αντικείμενο αυτό επιστρέφει όλες τις πληροφορίες των αντικειμένων που έχουν σαν id το “cheese”.

Το WebDriver αντιπροσωπεύει το πρόγραμμα περιήγησης.

Το WebElement αντιπροσωπεύει έναν συγκεκριμένο κόμβο DOM(πχ ένα κουμπί, ένα πεδίο)

2.10 Web Element

Αντιπροσωπεύει ένα στοιχείο HTML. Γενικά, όλες οι ενδιαφέρουσες λειτουργίες που σχετίζονται με την αλληλεπίδραση με μια σελίδα θα εκτελούνται μέσω αυτής της διεπαφής.Όλες οι κλήσεις μεθόδου θα κάνουν έλεγχο φρεσκάδας για να διασφαλίσουν ότι η αναφορά στοιχείων εξακολουθεί να ισχύει.

Αυτό καθορίζει ουσιαστικά αν το στοιχείο εξακολουθεί να είναι συνδεδεμένο στο DOM. Εάν αυτή η δοκιμή αποτύχει, τότε ένα StaleElementReferenceException ενεργοποιείται και όλες οι μελλοντικές κλήσεις σε αυτήν την περίπτωση θα αποτύχουν.Οι μέθοδοι που μπορούν να χρησιμοποιηθούν από ένα Web Element είναι :

Find Element : Χρησιμοποιείται για την εύρεση ενός στοιχείου και επιστρέφει μια πρώτη αντίστοιχη αναφορά WebElement, η οποία μπορεί να χρησιμοποιηθεί για μελλοντικές ενέργειες στοιχείων.

```
IWebElement searchBox = driver.FindElement(By.TagName("q"));
```

Find Elements : Παρόμοιο με το "Find Element", αλλά επιστρέφει μια λίστα με τα αντίστοιχα WebElements. Για να χρησιμοποιήσετε ένα συγκεκριμένο WebElement από τη λίστα, πρέπει να περιηγηθείτε στη λίστα στοιχείων για να εκτελέσετε την ενέργεια στο επιλεγμένο στοιχείο.

```
driver.Navigate().GoToUrl("https://example.com");

IList<IWebElement> elements = driver.FindElements(By.TagName("p"));

foreach (IWebElement e in elements)

{

  Console.WriteLine(e.Text);

}
```

Find Element From Element : Χρησιμοποιείται για την εύρεση ενός child στοιχείου στο πλαίσιο του γονικού στοιχείου. Για να επιτευχθεί αυτό, το parent WebElement είναι δεμένο με το «findElement» για πρόσβαση στο στοιχείο παιδί.

```
IWebDriver driver2 = new ChromeDriver();

driver.Url = "http://www.google.com";

IWebElement searchForm = driver.FindElement(By.TagName("form"));

IWebElement searchbox = searchForm.FindElement(By.Name("q"));
```

Find Elements From Element : Χρησιμοποιείται για την εύρεση της λίστας των αντίστοιχων παιδιών WebElements στο πλαίσιο του γονικού στοιχείου. Για να επιτευχθεί αυτό, το γονικό WebElement είναι δεμένο με «findElements» για πρόσβαση σε στοιχεία παιδιά.

```
IWebElement element = driver.FindElement(By.TagName("div"));

IList<IWebElement> elements = element.FindElements(By.TagName("p"));

foreach (IWebElement e in elements)

{

  System.Console.WriteLine(e.Text);

}
```

Get Active Element : Χρησιμοποιείται για την παρακολούθηση (ή) εύρεση στοιχείου DOM που έχει την ιδιότητα εστίασης(focus) στο τρέχον περιβάλλον περιήγησης.

```
string attr = driver.SwitchTo().ActiveElement().GetAttribute("title");
```

Is Element Enabled : Αυτή η μέθοδος χρησιμοποιείται για να ελέγξει εάν το συνδεδεμένο στοιχείο είναι ενεργοποιημένο ή απενεργοποιημένο σε μια ιστοσελίδα. Επιστρέφει μια τιμή boolean, True εάν το συνδεδεμένο στοιχείο είναι ενεργοποιημένο στο τρέχον περιβάλλον περιήγησης αλλιώς επιστρέφει false.

IWebElement element = driver.FindElement(By.Name("btnK"));

Console.WriteLine(element.Enabled);

Is Element Selected : Αυτή η μέθοδος καθορίζει εάν το στοιχείο είναι επιλεγμένο ή όχι. Αυτή η μέθοδος χρησιμοποιείται ευρέως σε πλαίσια ελέγχου, κουμπιά επιλογής, στοιχεία εισαγωγής και στοιχεία επιλογής.

Bool value =driver.FindElement(By.CssSelector("input[type='checkbox']:last-of-type")).Selected;

Get Element TagName : Χρησιμοποιείται για την ανάκτηση του ονόματος ετικέτας του στοιχείου που έχει την εστίαση(focus) στο τρέχον περιβάλλον περιήγησης.

string attr = driver.FindElement(By.CssSelector("h1")).TagName;

Get Element Rect : Χρησιμοποιείται για τη λήψη των διαστάσεων και των συντεταγμένων του στοιχείου αναφοράς.

var res = driver.FindElement(By.CssSelector("h1"));

// Επιστρέφει την θέση x και y του αναφερόμενου στοιχείου

Console.WriteLine(res.Location);

// Επιστρέφει το ύψος και το πλάτος του αναφερόμενου στοιχείου

Console.WriteLine(res.Size);

Get Element CSS Value : Ανακτά την τιμή της καθορισμένης ιδιότητας υπολογισμένου στυλ ενός στοιχείου στο τρέχον περιβάλλον περιήγησης.

string cssValue = driver.FindElement(By.LinkText("More information...")).GetCssValue("color");

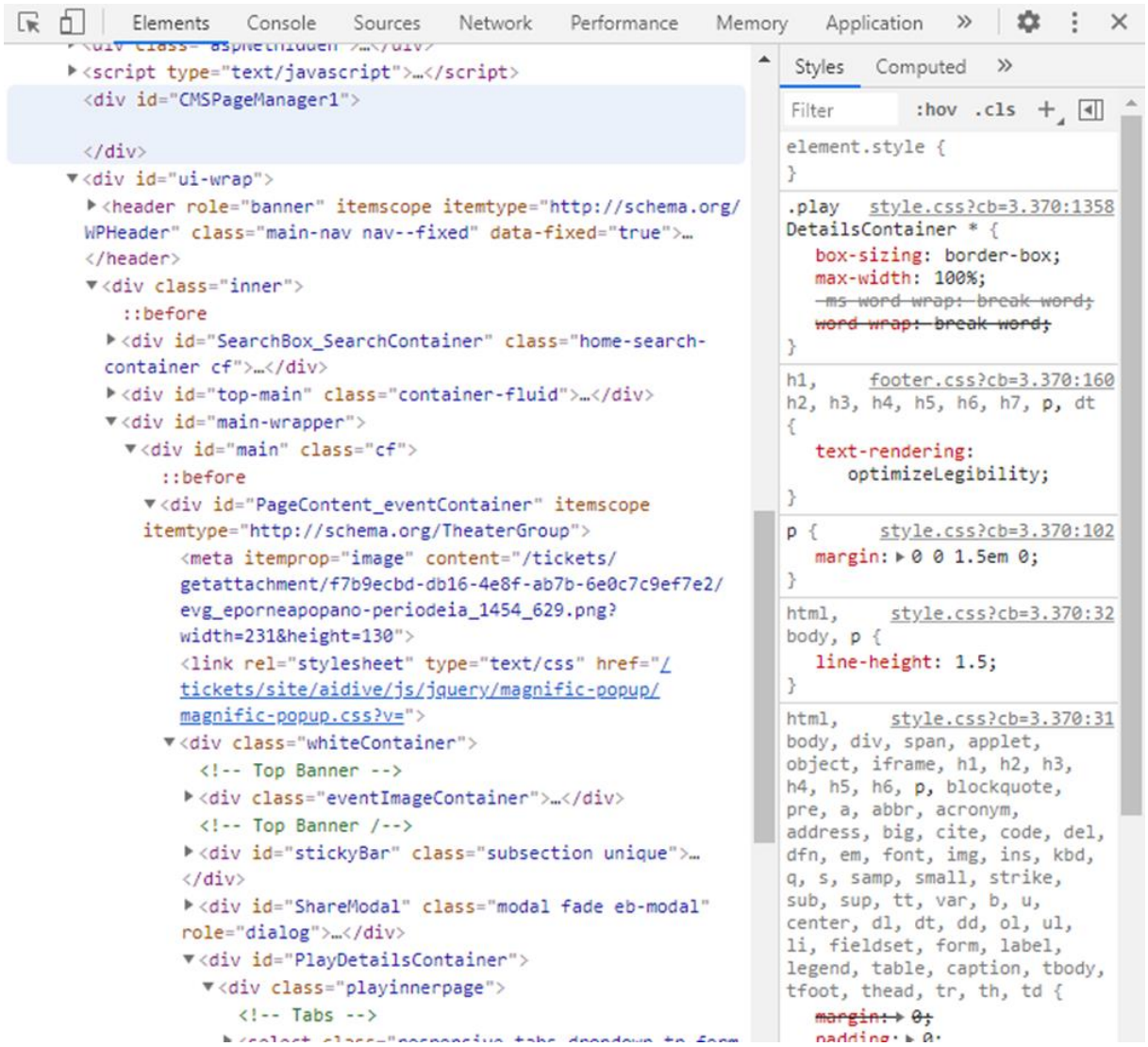
Get Element Text : Ανακτά το κείμενο του καθορισμένου στοιχείου.

string media = driver.FindElement(By.Id("openMedia")).Text;

2.11 Συμβουλές για χρήση selectors

Ο ιδανικός τρόπος για να δεις όλα τα στοιχεία ενός ιστότοπου απο την HTML πλευρά , είναι να πατήσεις το κουμπί F12 στον φυλλομετρητή η πατώντας δεξί κλίκ έλεγχος. Εκείνη την στιγμή ο φυλλομετρητής θα ανοίξει δεξιά στο πλάι ένα παράθυρο όπου φαίνεται ο HTML κώδικας.

Υπάρχουν αντικείμενα μέσα σε έναν ιστότοπο που βρίσκονται είτε εύκολα είτε δύσκολα .Ο τρόπος σκέψης όμως παίζει τον μεγαλύτερο ρόλο στην άντληση των στοιχείων αυτών , καθώς μπορείς να πάρεις ένα αντικείμενο , όχι μόνο με τους βασικούς τρόπους Class,Id, TagName, αλλά και με τα χαρακτηριστικά τους.Αυτά μπορεί να είναι το κείμενο ανάμεσα στην σήμανση τους, το μέγεθος τους , την θέση τους. Στο μυαλό σας να προσπαθείτε να βρείτε σύνθετους τρόπους , εκτός απο αυτούς που ήδη υπάρχουν.



Σχήμα 2.3: Εντοπισμός στοιχείων

Ένας καλογραμμένος επιλογέας CSS είναι η προτιμώμενη μέθοδος εντοπισμού ενός στοιχείου. Το XPath λειτουργεί και ως επιλογείς CSS, αλλά η σύνταξη είναι περίπλοκη και συχνά δύσκολο να εντοπιστεί. Αν και οι επιλογείς XPath είναι πολύ ευέλικτοι, συνήθως δεν δοκιμάζονται οι επιδόσεις από τους προμηθευτές του προγράμματος περιήγησης και τείνουν να είναι αρκετά αργοί. Προσωπικά χρησιμοποίησα XPath στο HAP ενώ στο selenium τους Selectors By.ClassName,By.ID, By.TagName.

Το By.TagName μπορεί να είναι ένας επικίνδυνος τρόπος εντοπισμού στοιχείων. Υπάρχουν συχνά πολλά στοιχεία της ίδιας ετικέτας σε μια σελίδα.Παρόλα αυτά είναι κυρίως χρήσιμο όταν καλείτε τη μέθοδο findElements (By) που επιστρέφει μια συλλογή στοιχείων. Η πρόταση μου είναι να διατηρήσετε τους Selector σας όσο το δυνατόν πιο συμπαγείς και ευανάγνωστους. Το να ζητάτε από το WebDriver να διασχίσει τη δομή DOM είναι μια δαπανηρή λειτουργία και όσο περισσότερο μπορείτε να περιορίσετε το εύρος της αναζήτησής σας, τόσο το καλύτερο.

2.12 Διαχείριση Cookies

Ένα cookie HTTP αποτελείται από πληροφορίες σχετικά με τον χρήστη και τις προτιμήσεις του. Αποθηκεύει πληροφορίες χρησιμοποιώντας ένα ζεύγος κλειδιού-τιμής. Πρόκειται για ένα μικρό κομμάτι δεδομένων που αποστέλλεται από την Εφαρμογή Ιστού και αποθηκεύεται στο Πρόγραμμα περιήγησης στο Web, ενώ ο χρήστης περιηγείται στον ιστότοπο.

Προσπαθώντας να αντλήσω στοιχεία από το WebDriver υπήρξε ένα θέμα καθώς ο selector μου δεν έβρισκε το αντικείμενο που ήθελα.Μετά απο πολύωρη αναζήτηση κατάλαβα ότι πρέπει να διαχειριστώ τα cookies του ιστότοπου.Οι επιλογές που έχει κάποιος για τα cookies είναι η να τα δεχτεί , η να τα απορρίψει. Αποφάσισα να τα δεχτώ , έτσι λοιπόν με τρείς γραμμές κώδικα λύθηκε το πρόβημα μου :

ChromeDriver driver = new ChromeDriver(); // δημιουργώ τον chrome φυλλομετρητη που φορτώνει //την σελίδα

var cookies = driver.Manage().Cookies.AllCookies;

//δίνω εντολή στον driver , να βρεί το στοιχείο με κλάση ‘cc-btn—accept’ και να το κάνει κλικ

driver.FindElement(By.XPath("//a[contains(@class,'cc-btn-accept')]")).Click();

Βρίσκοντας με την XPath την επιλογή αποδοχής Cookie και με την εντολή Click() που παρέχει ο WebDriver αποδέχτηκα τα cookies και η αναζήτηση του στοιχείου έγινε επιτυχώς.

2.13 Χρόνος αναμονής

Αν κάποιος προσέξει, οι πληροφορίες που φορτώνουν σε έναν Web Browser δεν γίνονται πάντα με γρήγορο τρόπο. Αυτό έχει ως συνέπεια την δημιουργία λαθών , σε ειδικές περιπτώσεις που χρειάζεται να αντλήσουμε πληροφορίες που φορτώνουν με javascript. Γι αυτό το λόγο το Selenium μας παρέχει κατάσταση αναμονής που μπορεί να προσαρμοστεί ανάλογα με τις ανάγκες σας. Μερικές φορές δεν είναι απαραίτητο να περιμένουμε ολόκληρο το προεπιλεγμένο χρονικό όριο, καθώς η ποινή για μη επίτευξη επιτυχούς κατάστασης μπορεί να είναι δαπανηρή.Ο παρακάτω κώδικας δείχνει ότι περιμένει 3 δευτερόλεπτα αφού ξεκινήσει το φόρτωμα της σελίδας , να εμφανιστεί το αντικείμενο a με γραμματοσειρά h3 :

NewWebDriverWait(driver,
Duration.ofSeconds(3)).until(ExpectedConditions.elementToBeClickable(By.xpath("//a/h3")));

Υπάρχει ένας δεύτερος τύπος αναμονής που διαφέρει από τη παραπάνω αναμονή που ονομάζεται σιωπηρή αναμονή(implicit wait). Περιμένοντας σιωπηρά, το WebDriver εξετάζει το DOM για μια συγκεκριμένη διάρκεια προσπαθώντας να βρει οποιοδήποτε στοιχείο. Αυτό μπορεί να είναι χρήσιμο

όταν ορισμένα στοιχεία στην ιστοσελίδα δεν είναι άμεσα διαθέσιμα και χρειάζονται λίγο χρόνο για φόρτωση.Παράδειγμα κώδικα με την σιωπηρή μέθοδο :

```
IWebDriver driver = new ChromeDriver();

driver.Manage().Timeouts().ImplicitWait = TimeSpan.FromSeconds(10);

driver.Url = "http://somedomain/url_that_delays_loading";

IWebElement dynamicElement = driver.FindElement(By.Name("dynamicElement"));
```

Ο τρόπος που χρησιμοποίησα εδώ είναι μία βιβλιοθήκη που μου παρέχει το .net framework τα Thread για τον έλεγχο νημάτων.Χρησιμοποιώ την μέθοδο που μου παρέχει , συγκεκριμένα την Sleep, για να δώσει χρόνο στο WebDriver ώστε να φορτώσει η javascript .Σαν παράμετρο η μέθοδος παίρνει τιμές σε millisecond , οπότε αν θέλουμε πχ 2 δευτερόλεπτα παύσης θα βάλουμε 2000.

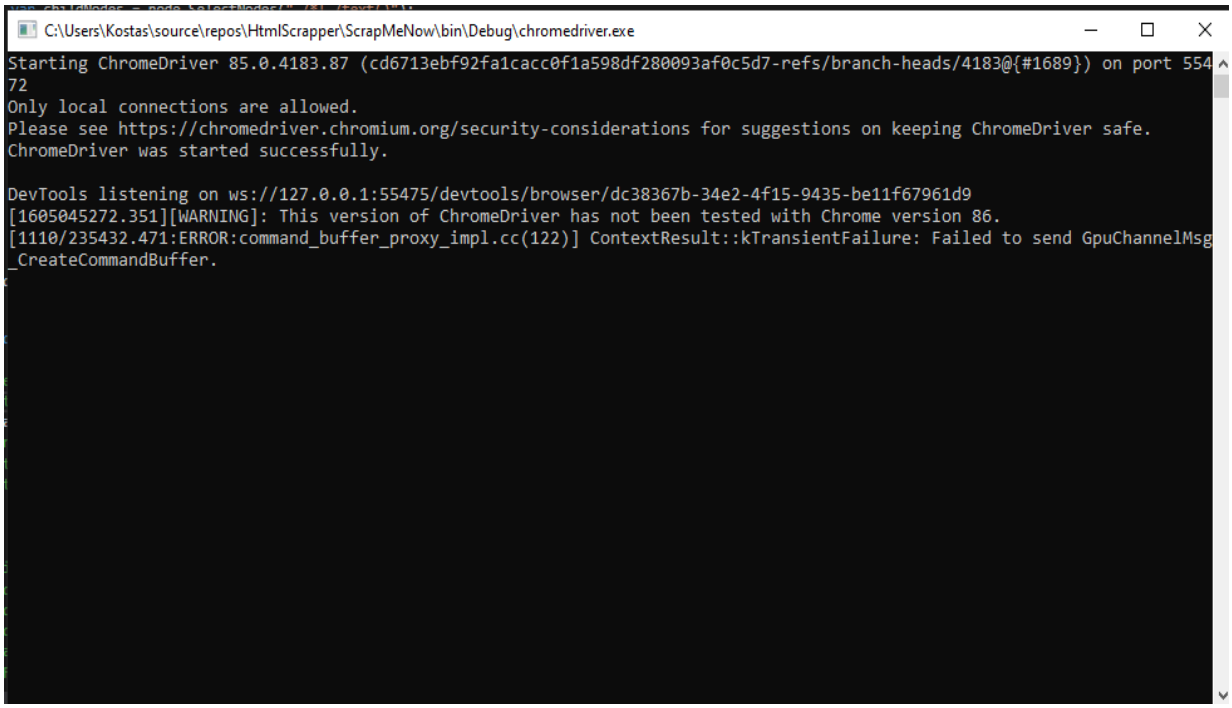
```
driver.FindElement(By.Id("openMedia")).Click();

Thread.Sleep(3000);

Boolean isPresent = driver.FindElements(By.ClassName("mfp-img")).ToString().Length > 0;
```

2.14 Chrome Driver

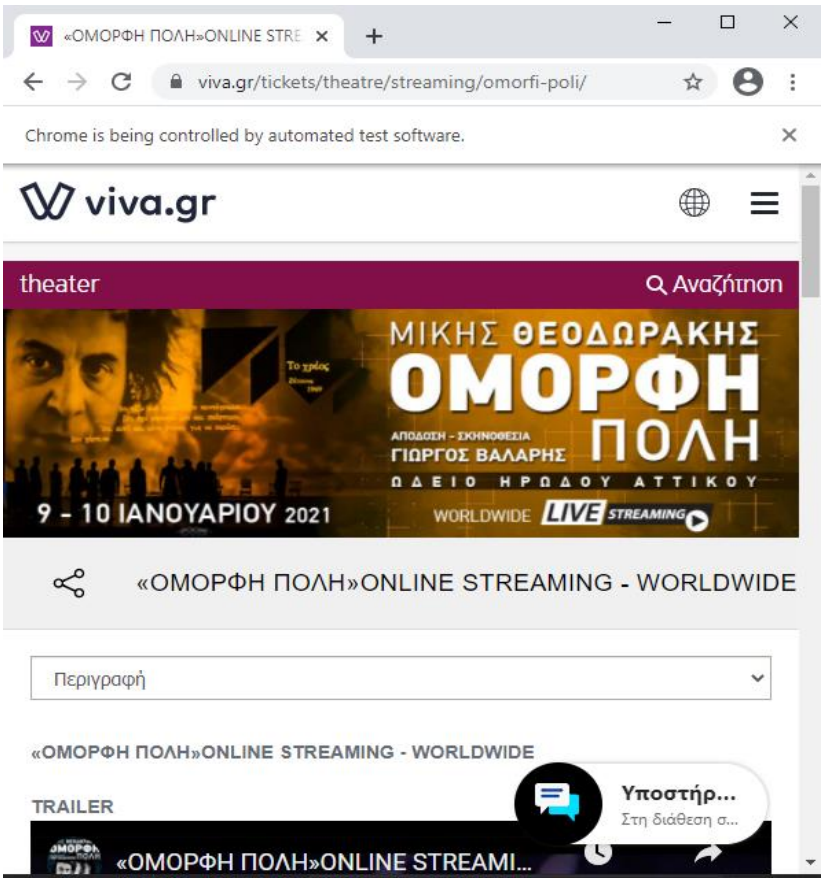
Επιλέγοντας να δουλέψω με το Chrome Driver , κλάση του οποίου μου προσφέρει διάφορες σημαντικές ιδιότητες που χρησιμοποίησα για καλύτερη εμπειρία του χρήστη στο project .Η λίστα με τις ιδιότητες που παρέχει η κλάση αυτή είναι :

- start-maximized: Ανοίγει το Chrome σε λειτουργία μεγιστοποίησης
- incognito: Ανοίγει το Chrome σε κατάσταση ανώνυμης περιήγησης
- headless: Ανοίγει το Chrome σε λειτουργία χωρίς κεφαλή
- disable-extensions: Απενεργοποιεί τις υπάρχουσες επεκτάσεις στο πρόγραμμα περιήγησης Chrome
- disable-popup-blocking: Απενεργοποιεί τα αναδυόμενα παράθυρα που εμφανίζονται στο πρόγραμμα περιήγησης Chrome
- make-default-browser: Κάνει το Chrome προεπιλεγμένο πρόγραμμα περιήγησης
- version: Εκτυπώνει την έκδοση του προγράμματος περιήγησης chrome
- disable-infobars: Αποτρέπει το Chrome από την εμφάνιση της ειδοποίησης. Το Chrome ελέγχεται από αυτοματοποιημένο λογισμικό

Η ιδιότητα που φάνηκε αρκετά χρήσιμη από όλες είναι η headless καθώς ο Web Browser εκτελείται στο παρασκήνιο. Δεν θα δείτε το πρόγραμμα περιήγησης ή τις λειτουργίες που εκτελούνται σε αυτό. Από προγραμματιστική άποψη , θεώρησα σωστό να χρησιμοποιήσω την ιδιότητα καθώς , κάθε φορά που θα χρειάζοταν το Selinium να τραβήξει δεδομένα , ο Web Browser τίθεται στην λειτουργία , και η εμφανίση ενός παραθύρου Chrome από το πουθενά δεν θα ήταν και ότι καλύτερο. Αντ' αυτού, εμφανίζεται ένα command prompt που αφού κάνει την δουλειά που πρέπει και με χρήση της εντολής driver.Quit() σταματάει το τρέξιμο και κλείνει.



Σχήμα 2.4: Headless mode prompt του Selenium



Σχήμα 2.5: Selenium Chrome Driver

2.15 Οι πληροφορίες που γίνεται το Scraping

Ρόλος-Μέλος παράστασης : Εντοπίζοντας το στοιχείο(με inspect element) και βλέποντας τον κόμβο πατέρα , σπάω την κάθε γραμμή σε δύο string όπου το αριστερά αναφέρεται στους ρόλους και το δεξιά στα μέλη.Έχει γίνει χρήση Regex (Regex.IsMatch(line, @"[\w]{1,}(.*):(.*)[\w]{1,}"))

Σχήμα 2.6: Υπόδειγμα Ρόλου-Μέλους παράστασης στον ιστότοπο

Ημερομηνία,Θέατρο , Τιμές : Τιμές που είναι απαραίτητες για την εισαγωγή στον πίνακα events.

List<IWebElement> date=driver.FindElements(By.XPath("//div[contains(@class,'events-container__item-date')]")).ToList();

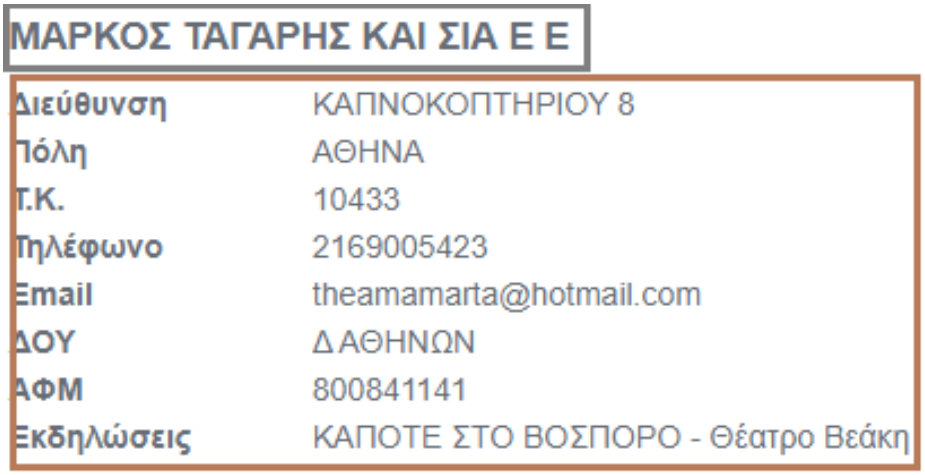
var money = driver.FindElements(By.CssSelector(".events-container__item-prices")).ToList();

Σχήμα 2.7: Υπόδειγμα προβολών παραστάστων στον ιστότοπο

Διοργανωτής :Για τον εντοπισμό των χαρακτηριστικών του διοργανωτή ,παρατηρώ ότι όλα ανήκουν στην κλάση 'field-group'.Έτσι πολύ απλά δημιουργώ μία λίστα που

μαζέυει αυτούς τους κόμβους .

var fields = doc?.DocumentNode?.SelectNodes("//div[@class='field']").ToList();



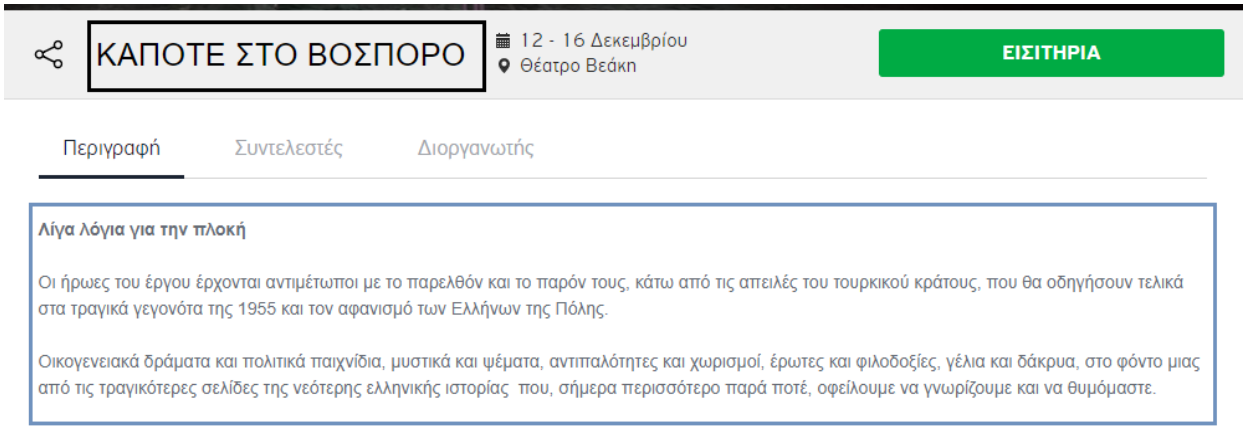
Σχήμα 2.8: Υπόδειγμα οργανωτή παράστασης στον ιστότοπο

Τίτλος – Περιγραφή :Εύκολα μπορείς με την χρήση του εντοπισμού στοιχείου να βρείς το αντικείμενο του τίτλου.Αυτή την φορά θέλουμε επιλογή ενός στοιχείου και όχι συνόλου.Οπότε χρησιμοποιούμε την εξής εντολή.

var title = doc?.DocumentNode?.SelectSingleNode("//h1[@id='playTitle']");

Η περιγραφή συνήθως αποτελείται απο πολλούς παραγράφους <p> οπότε μιλάμε για πολλά στοιχεία .Παρόλα αυτά πάντα κάνουμε ασφαλή αναζήτηση.

var desc = doc?.DocumentNode?.SelectNodes("//div[@itemprop='description']")?.ToList();



Σχήμα 2.9: Υπόδειγμα Τίτλου-Περιγραφής παράστασης στον ιστότοπο

2.16 Επίλογος

Μέσα από το κεφάλαιο αυτό πήραμε μία ιδέα για τα εργαλεία web scraping καθώς και τα χαρακτηριστικά τους.Επίσης είδαμε τί είδους στοιχεία που αντλούμε και πως τα αντλούμε απο κάθε παράσταση με παραδείγματα κώδικα και εικόνες.

Κεφάλαιο 3ο: SQL

3.1 Εισαγωγή

Στο παρακάτω κεφάλαιο θα ειπωθούν κάποια βασικά πράγματα σχετικά με την γλώσσα των βάσεων δεδομένων , επίσης θα γίνει μία ανάλυση της βάσης που project αλλά και κάποιων χαρακτηριστικών της ,τα οποία είναι απαραίτητο να εξηγηθούν για να υπάρχει μία καθαρά και ξεκάθαρη εικόνα .

3.2 Τι είναι

Η SQL (από το Structured Query Language)είναι μία γλώσσα στις βάσεις δεδομένων, που σχεδιάστηκε για τη διαχείριση δεδομένων, σε ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων , η οποία αρχικά βασίστηκε στη σχεσιακή άλγεβρα. Η γλώσσα περιλαμβάνει δυνατότητες ανάκτησης και ενημέρωσης δεδομένων, δημιουργίας και τροποποίησης σχημάτων και σχεσιακών πινάκων, αλλά και ελέγχου πρόσβασης στα δεδομένα.

Τα γλωσσικά στοιχεία

- Clauses, οι οποίες είναι σε μερικές περιπτώσεις προαιρετικές, αλλά απαραίτητα συστατικά των δηλώσεων και ερωτήσεων.
- Expressions που μπορούν να παραγάγουν είτε τις κλιμακωτές τιμές είτε πίνακες που αποτελούνται από στήλες και σειρές στοιχείων.
- Predicates που διευκρινίζουν τους όρους που μπορούν να αξιολογηθούν σαν σωστό ή λάθος.
- Queries που ανακτούν τα στοιχεία βασισμένα σε ειδικά κριτήρια.
- Statements που μπορούν να έχουν μια επίδραση στα σχήματα και τα στοιχεία, ή που μπορούν να ελέγξουν τη ροή του προγράμματος και τις συνδέσεις από άλλα προγράμματα.
- Το κενό αγνοείται γενικά στα Statements και τα SQL queries. Ένα κενό είναι όμως απαραίτητο για να ξεχωρίζει Statements όπως και στην κανονική γραφή κειμένων.

3.3 Βάση δεδομένων

Μια βάση δεδομένων είναι μια συλλογή οργανωμένων εγγραφών, τις οποίες μπορούμε να αποθηκεύσουμε, να επεξεργαστούμε. Οι βάσεις δεδομένων μπορούν να αποθηκεύουν πληροφορίες σχετικά με άτομα, προϊόντα, παραγγελίες ή οτιδήποτε άλλο. Πολλές βάσεις δεδομένων ξεκινούν ως μια λίστα σε ένα πρόγραμμα επεξεργασίας κειμένου . Καθώς η λίστα μεγαλώνει, αρχίζουν να εμφανίζονται επαναλήψεις και ασυνέπειες. Τα δεδομένα γίνεται δύσκολο να κατανοηθούν σε μορφή λίστας και υπάρχουν περιορισμένοι τρόποι για την αναζήτηση ή την άντληση υποσυνόλων δεδομένων για αναθεώρηση. Μόλις αρχίσουν να εμφανίζονται αυτά τα προβλήματα, συνιστάται να μεταφέρετε τα δεδομένα σε μια βάση δεδομένων που δημιουργήθηκε από ένα σύστημα διαχείρισης βάσεων δεδομένων , όπως η MySQL η οποιαδήποτε άλλη.

Μία βάση δεδομένων μπορεί να περιέχει περισσότερους από έναν πίνακες. Για παράδειγμα, ένα σύστημα καταχώρησης δανειστών βιβλίων , περιέχει ξεχωριστούς πίνακες για τους πελάτες , τα βιβλία , τις ενοικιάσεις των βιβλίων. Αυτα τα σύνολα είναι διαφορετικά μεταξύ τους και το μόνο που μπορεί να τα συνδέσει είναι οι τυχόν συσχετίσεις που λαμβάνουν χώρα στον πίνακα των ενοικιάσεων , όπου κάθε πίνακας έχει απο ένα κύριο κλειδί για λόγους μοναδικότητας.

Οι συνηθισμένοι τύποι αρχείων βάσεων δεδομένων καταλήγουν σε .accdb ,.mbd.db.mdf,.sql .

3.4 MySQL

Η MySQL είναι ένα σύστημα διαχείρησης σχεσιακών βάσεων που χρησιμοποιεί την sql , για πρόσβαση και επεξεργασία των δεδομένων της βάσης. Κάθε DBMS ουσιαστικά δέχεται και εκτελεί ένα σετ εντολών SQL για να διαχειριστεί τα δεδομένα του. Η MySQL είναι ανοικτού κώδικα και μπορεί να την κατεβάσει ο καθένας .Χαρακτηρίζεται για την ταχύτητα, την ευχρηστία , την αξιοπιστία που παρέχει.Ως μία απο τις πιο δημοφιλής βάσεις δεδομένων , η MySQL χρησιμοποιείται απο ιστοσελίδες αλλά και διαδικτυακά προγράμματα.Παρακάτω θα δούμε τους κύριους τύπους της και τα χαρακτηριστικά τους.

Οι τύποι μπορεί να είναι Αριθμητικοί, Αλφαριθμητικοί, Ημερομηνίες και Ώρες:

Αριθμητικοί:

- INT : ακέραιος με μήκος έως 11 ψηφία.
- FLOAT: κινητής υποδιαστολής έως 24 ψηφία.
- DOUBLE: διπλής ακρίβειας έως 53 ψηφία.
- BIGINT: ακέραιος με μήκος έως 20 ψηφία.
- TINYINT: ακέραιος με μήκος έως 4 ψηφία.
- SMALLINT: ακέραιος με μήκος έως 5 ψηφία.
- MEDIUMINT: ακέραιος με μήκος έως 9 ψηφία.

Αλφαριθμητικοί:

- CHAR: Έχει σταθερό μέγεθος από 1-255 χαρακτήρες.
- VARCHAR: Έχει μεταβλητό μέγεθος από 1-255 χαρακτήρες.
- TEXT: Έχει μέγιστο μήκος 65535.
- TINYTEXT: Έχει μέγιστο μήκος 255.
- MEDIUMTEXT: Έχει μέγιστο μήκος 16777215.
- LONGTEXT: Έχει μέγιστο μήκος 4294967295

Ημερομηνίες και Ώρες:

- DATE: Ημερομηνία στη μορφή ΕΕΕΕ-ΜΜ-ΗΗ
- TIME: Ώρα στη μορφή ΩΩ:ΛΛ:ΔΔ
- YEAR: Έτος στη διψήφια μορφή από 1970-2069.
  Έτος στην τετραψήφια μορφή 1901-2155.
- DATETIME: Ημερομηνία και ώρα στη μορφή ΕΕΕΕ-ΜΜ-ΗΗ ΩΩ:ΛΛ:ΔΔ

3.4.1 Πως δημιουργώ τη δική μου βάση δεδομένων σε MySQL?

Μπορείς να δουλέψεις τοπικά στον υπολογιστή σου , κατεβάζοντας τοπικό server όπως Apache,Xampp η στην περίπτωση μας , μέσα από το CPanel να επικοινωνείτε με μία online βάση.Τα αναφερόμενα συστήματα είναι δωρεάν και υπάρχουν στο διαδίκτυο στην διάθεση μας.

3.5 Triggers

Ενα Trigger(έναυσμα) είναι ένα αποθηκευμένο πρόγραμμα που καλείται αυτόματα όταν γίνεται ένα συμβάν όπως εισαγωγή, ενημέρωση ή διαγραφή που εμφανίζεται στον σχετικό πίνακα. Για παράδειγμα, μπορείτε να ορίσετε μια σκανδάλη που καλείται αυτόματα προτού εισαχθεί μια νέα σειρά σε έναν

πίνακα. Η MySQL υποστηρίζει Triggers που ενεργοποιούνται ως απάντηση στο συμβάν INSERT, UPDATE ή DELETE.

Το πρότυπο SQL ορίζει δύο τύπους trigger: triggers επιπέδου γραμμής και triggers επιπέδου δήλωσης.

- Ένας trigger επίπεδου-γραμμής ενεργοποιείται για κάθε σειρά που εισάγεται, ενημερώνεται ή διαγράφεται. Για παράδειγμα, εάν ένας πίνακας έχει 50 σειρές που έχουν εισαχθεί, ενημερωθεί ή διαγραφεί, ο κανόνας ενεργοποιείται αυτόματα 50 φορές για τις 50 σειρές που επηρεάζονται.
- Ένας trigger επίπεδου-δήλωσης εκτελείται μία φορά για κάθε συναλλαγή, ανεξάρτητα από το πόσες σειρές εισάγονται, ενημερώνονται ή διαγράφονται.

Η MySQL υποστηρίζει μόνο triggers επιπέδου γραμμής. Δεν υποστηρίζει ενεργοποιήσεις σε επίπεδο δήλωσης.

Πλεονεκτήματα των trigger

- Οι triggers παρέχουν έναν εναλλακτικό τρόπο ελέγχου ακεραιότητας των δεδομένων.
- Οι triggers χειρίζονται σφάλματα στο επίπεδο της βάσης δεδομένων.
- Οι triggers δίνουν έναν εναλλακτικό τρόπο εκτέλεσης προγραμματισμένων εργασιών. Χρησιμοποιώντας κανόνες ετικέτας, δεν χρειάζεται να περιμένετε να εκτελεστούν τα προγραμματισμένα συμβάντα, επειδή οι κανόνες ενεργοποιούνται αυτόματα πριν ή μετά από μια αλλαγή. Οι κανόνες ετικέτας μπορούν να είναι χρήσιμοι για τον έλεγχο των αλλαγών δεδομένων στους πίνακες. τα δεδομένα σε έναν πίνακα.

Μειονεκτήματα των trigger

- Οι triggers μπορούν να παρέχουν μόνο εκτεταμένες επικυρώσεις και όχι όλες. Για απλές επικυρώσεις, μπορείτε να χρησιμοποιήσετε τους περιορισμούς NOT NULL, UNIQUE, CHECK και FOREIGN KEY.
- Μπορεί να είναι δύσκολο να βρεις κάποιο σφάλμα στους triggers, επειδή εκτελούνται αυτόματα στη βάση δεδομένων.
- Οι triggers ενδέχεται να αυξήσουν τα γενικά έξοδα του διακομιστή MySQL.

Η ανάλυση των trigger έγινε σκόπιμα καθώς έχει φτιαχτεί ένας τέτοιος αλγόριθμος που ενημερώνει τον πίνακα changelog σε κάθε αλλαγή, εισαγωγή , διαγραφή που γίνεται στην βάση.

3.6 Prepared Statements

Ένα prepared statement είναι μια δυνατότητα που χρησιμοποιείται για την εκτέλεση των ίδιων (ή παρόμοιων) δηλώσεων SQL επανειλημμένα με υψηλή απόδοση. Λειτουργούν βασικά ως εξής :

1. Προετοιμασία: Ένα πρότυπο δήλωσης SQL δημιουργείται και αποστέλλεται στη βάση δεδομένων. Ορισμένες τιμές παραμένουν μη καθορισμένες, που ονομάζονται παράμετροι (με την ένδειξη "?" η κάποιον άλλον ειδικό χαρακτήρα που μπορεί να ορίσει ο χρήστης).

2. Η βάση δεδομένων αναλύει, μεταγλωττίζει και εκτελεί βελτιστοποίηση ερωτήματος στο πρότυπο δήλωσης SQL και αποθηκεύει το αποτέλεσμα χωρίς να το εκτελεί.

3. Εκτέλεση: Αργότερα, η εφαρμογή δεσμεύει τις τιμές στις παραμέτρους και η βάση δεδομένων εκτελεί τη δήλωση. Η εφαρμογή μπορεί να εκτελέσει τη δήλωση όσες φορές θέλει με διαφορετικές τιμές.

Σε σύγκριση με την άμεση εκτέλεση δηλώσεων SQL, τα prepared statements έχουν τρία κύρια πλεονεκτήματα:

- Τα prepared statement μειώνουν τον χρόνο ανάλυσης καθώς η προετοιμασία στο ερώτημα γίνεται μόνο μία φορά (αν και η δήλωση εκτελείται πολλές φορές)

- Τα prepared statement ελαχιστοποιούν το εύρος ζώνης στο διακομιστή καθώς χρειάζεστε να στέλνετε μόνο τις παραμέτρους κάθε φορά και όχι ολόκληρο το ερώτημα

- Τα prepared statement είναι πολύ χρήσιμα έναντι των SQL injections, επειδή οι τιμές των παραμέτρων, οι οποίες μεταδίδονται αργότερα χρησιμοποιώντας διαφορετικό πρωτόκολλο, δεν χρειάζεται να ξεφύγουν. Εάν το αρχικό πρότυπο δήλωσης δεν προέρχεται από εξωτερική είσοδο, δεν μπορεί να πραγματοποιηθεί SQL injection.

Παράδειγμα prepared statement με εισαγωγή εγγραφής στο Venue table

```
MySqlCommand cmd = con.CreateCommand();

cmd.CommandText = @"INSERT INTO venue ( Title,Address,SystemID) VALUES(@Ttl, @Ads, @SysID) ON DUPLICATE KEY UPDATE score = score + @score";

cmd.Parameters.AddWithValue("@Ttl ", MySqlDbType.VarChar, 32);

cmd.Parameters.AddWithValue("@Ads ", MySqlDbType.Int32);

cmd.Parameters.AddWithValue("@SysID ", MySqlDbType.VarChar, 1);

cmd.ExecuteNonQuery();
```

3.7 Ανάλυση βάσης project

Μιάς και μιλάμε για βάσεις δεδομένων , ήρθε η ώρα να γίνει ανάλυση της βάσης του project μου. Προς ενημέρωση του αναγνώστη ,η βάση μοιράζεται καθώς το θέμα της πτυχιακής υλοποιείτε σε άλλη γλώσσα απο συμφοιτητή του τμήματος , οπότε είναι κοινή . Οι πίνακες της βάσης είναι 9 και είναι οι εξής :

System table: Επειδή η βάση χρησιμοποιείται απο δύο συστήματα, καλή ιδέα είναι να ξέρουμε από ποιό σύστημα γίνονται οι εγγραφές. Οι πιθανές τιμές(2,'Python'),(3,'C#').

Πίνακας 3.1: System Table

Όνομα	**Τύπος**	**Πρόσθετα**
ID	int(10)	AUTO_INCREMENT
Name	varchar(60)	Python | C#

Venue table: Για τα χαρακτηριστικά των θεάτρων/αιθουσών αποφασίσαμε σαν πεδία να βάλουμε το ID σαν αναγνωριστικό και κύριο κλειδί του πίνακα.Κάθε θέατρο έχει ένα όνομα εξού και ο τίτλος.Το Address μπορεί να περιλαμβάνει την πόλη, περιοχή ακόμα και την ακριβή διεύθυνση του θεάτρου.Το SystemID που δείχνει από ποιό σύστημα έγινε η εγγραφή και το Timestamp που είναι η ημερομηνία εισαγωγής της εγγραφής στον πίνακα.

Πίνακας 3.2: Venue Table

Όνομα	**Τύπος**	**Πρόσθετα**
ID	Int(11)	AUTO_INCREMENT
Title	varchar(60)	
Address	varchar(60)	
SystemID	Int(10)	Python | C#
Timestamp	timestamp	On update CURRENT_TIMESTAMP

Persons table: Ο πίνακας για την αποθήκευση μελών παράστασης. Τα πεδία που υπάρχουν είναι το ID σαν αναγνωστιστικό και κύριο κλειδί του πίνακα. Fullname είναι το ονομ/μο του μέλους παράστασης,

Το SystemID που δείχνει από ποιό σύστημα έγινε η εγγραφή και το Timestamp είναι η ημερομηνία εισαγωγής της εγγραφής στον πίνακα

Πίνακας 3.3: Persons Table

Όνομα	**Τύπος**	**Πρόσθετα**
ID	Int(11)	AUTO_INCREMENT
Fullname	varchar(150)	
SystemID	int(10)	Python | C#
Timestamp	timestamp	ON UPDATE CURRENT_TIMESTAMP

Roles table: Ο πίνακας για την αποθήκευση ρόλων των μελών της παράστασης , αποτελείται απο το πεδίο ID ως κύριο κλειδί και το όνομα του ρόλου Role,το SystemID που δείχνει από ποιό σύστημα έγινε η εγγραφή και το Timestamp είναι η ημερομηνία εισαγωγής της εγγραφής στον πίνακα.

Πίνακας 3.4: Roles Table

Όνομα	**Τύπος**	**Πρόσθετα**
ID	int(11)	AUTO_INCREMENT
Role	varchar(150)	
SystemID	int(10)	Python | C#
timestamp	Timestamp	ON UPDATE CURRENT_TIMESTAMP

Organizer table : Ο πίνακας που αποθηκεύει τους διοργανωτές με ID ως κύριο κλειδί , Name το όνομα του διοργανωτή , Address η διεύθυνση της διοργάνωσης , Town η πόλη στην οποία βρίσκεται , postcode ο ταχυδρομικός κώδικας , Phone τηλέφωνο επικοινωνίας, Email (ηλεκρονική διεύθυνση) επικοινωνίας , Doy η ΔΟΥ , Afm το ΑΦΜ του διοργανωτή , και τα πεδία SystemID & timestamp.

Πίνακας 3.5: Organizer Table

Όνομα	Τύπος	Πρόσθετα
ID	int(11)	AUTO_INCREMENT
Name	varchar(80)	
Address	varchar(50)	
Town	varchar(30)	
postcode	varchar(20)	
Phone	varchar(30)	
Email	varchar(30)	
Doy	varchar(30)	
Afm	varchar(30)	
SystemID	int(10)	Python | C#
timestamp	timestamp	ON UPDATE CURRENT_TIMESTAMP

Production table: Ο πίνακας στον οποίο αποθηκεύονται οι πληροφορίες για τις παραστάσεις. Αυτές οι πληροφορίες είναι Title , ο τίτλος της παράστασης , η περιγραφή Description αν έχει , το Url της σελίδας στην οποία φιλοξενεί την συγκεκριμένη παράσταση (Το πεδίο χρησιμοποιηθήκε για έλεγχο πριν την εισαγωγή ώστε να μην μπαίνουν ίδιες παραστάσεις ξανά και ξανά), Producer οπού είναι το όνομα της παραγωγής της παράστασης , MediaURL είναι το url που μπορεί να έχει μία παράσταση , αυτό είναι βίντεο η εικόνα. Η διάρκεια της παράστασης duration , και τα γνωστά πεδία SystemID και Timestamp.Το ID της παράστασης ορίζεται ως κύριο κλειδί . Εδώ έχουμε ξένο κλειδί το ID του διοργανωτή της παράστασης.

Πίνακας 3.6: Production Table

Όνομα	Τύπος	Πρόσθετα
ID	int(11)	AUTO_INCREMENT
ProductionID	int(11)	
VenueID	int(11)	
DateEvent	datetime	
PriceRange	varchar(30)	
SystemID	int(10)	Python | C#
timestamp	timestamp	ON UPDATE CURRENT_TIMESTAMP

Contributions table: Ο πίνακας ο οποίος δείχνει την συμμετοχή κάθε μέλους παράστασης αλλά και τον ρόλο του, την παράσταση στην οποία είναι μέλος είναι πλήρως απαραίτητη και σημαντική η ύπαρξη του. ID κύριο κλειδί, PeopleID το ξένο κλειδί που αναφέρεται στον άνθρωπο μέλος, RoleID το ξένο κλειδί που αναφέρεται στον ρόλο του μέλους, subRole είναι το πεδίο το οποίο προστέθηκε για να περιοριστούν οι θεατρικοί ρόλοι που μπορεί να υποδύεται κάποιος ηθοποιός. Τέλος τα γνωστά πεδία SystemID & timestamp.

Πίνακας 3.7: Contributions Table

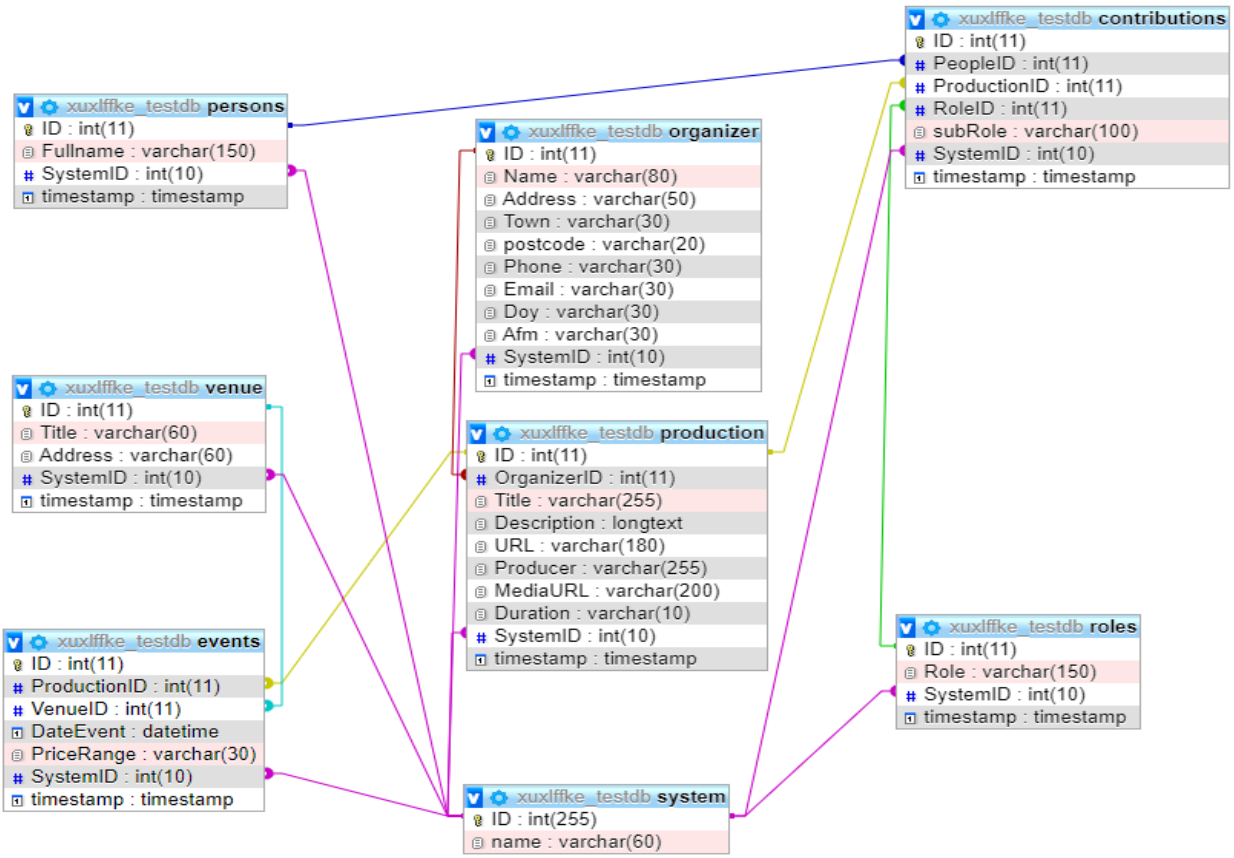
Όνομα	Τύπος	Πρόσθετα
ID	int(11)	AUTO_INCREMENT
PeopleID	int(11)	
ProductionID	int(11)	
RoleID	int(11)	
subRole	varchar(100)	
SystemID	int(10)	Python | C#
timestamp	timestamp	ON UPDATE CURRENT_TIMESTAMP

Changelog table :Ο πίνακας στον οποίο αποθηκεύονται οι αλλαγές , διαγραφές , εισαγωγές στην βάση. Έχει το ID σαν κύριο κλειδί , EventType πεδίο το οποίο δείχνει τι είδους αλλαγή έγινε στην εγγραφή , Value η τιμή που άλλαξε , CollumnName το πεδίο που άλλαξε , timestamp η χρονική στιγμή που έγινε η εγγραφή.

Πίνακας 3.8: Change log Table

Όνομα	Τύπος	Πρόσθετα
ID	int(11)	AUTO_INCREMENT
EventType	varchar(100)	
Value	varchar(100)	
CollumnName	varchar(100)	
timestamp	timestamp	ON UPDATE CURRENT_TIMESTAMP

Σχήμα 3.1: Στιγμιότυπο βάσης MySQL με τους πίνακες και τις μεταξύ τους σχέσεις



3.8 Επίλογος

Κλείνοντας το κεφάλαιο μάθαμε τα βασικά πράγματα που αφορούν τις βάσεις δεδομένων και την γλώσσα SQL, είδαμε την βάση που χρησιμοποιούμε για την αποθήκευση των web scraped δεδομένων έχοντας μία ολοκληρωμένη εικόνα.

Κεφάλαιο 4ο: GitHub

4.1 Εισαγωγή

Σε αυτό το κεφάλαιο εξηγώ λίγα πράγματα για το GitHub , τις εντολές που παρέχει , και τον τρόπο δημιουργίας του project της πτυχιακής , όπως και το ανέβασμα του στον λογαριασμό μου στο GitHub μέσω του Visual Studio 2019.

4.2 Τι είναι

Το GitHub είναι μια διαδικτυακή πλατφόρμα παρέχει φιλοξενία για ανάπτυξη λογισμικού και έλεγχο έκδοσης χρησιμοποιώντας το Git. Προσφέρει τη λειτουργική διαχείριση κατανεμημένων εκδόσεων και διαχείρισης πηγαίου κώδικα (SCM) του Git, καθώς και τις δικές της δυνατότητες. Παρέχει έλεγχο πρόσβασης και πολλές δυνατότητες συνεργασίας, όπως παρακολούθηση σφαλμάτων, αιτήματα λειτουργιών, διαχείριση εργασιών, συνεχή ενσωμάτωση για κάθε έργο. Υπάρχει και σαν desktop εφαρμογή , αν θέλετε να χρησιμοποιήσετε τις λειτουργίες του . Πολλές ομάδες developer δουλεύουν με git η GitHub , γεγονός που το κάνει μέρα με την μέρα ακόμα πιο δημοφιλές , με χιλιάδες developers να θέλουν να το μάθουν.

4.3 Branch & Repository

Ένα repository περιλαμβάνει ολόκληρη τη συλλογή αρχείων και φακέλων που σχετίζονται με ένα έργο, μαζί με το ιστορικό αναθεωρήσεων κάθε αρχείου. Το ιστορικό αρχείων εμφανίζεται ως στιγμιότυπα στο χρόνο που ονομάζονται commits(δεσμεύσεις) και τα commits υπάρχουν ως σχέση συνδεδεμένης λίστας και μπορούν να οργανωθούν σε πολλές γραμμές ανάπτυξης που ονομάζονται branches.

Ένα branch είναι ουσιαστικά ένα μοναδικό σύνολο αλλαγών κώδικα με ένα μοναδικό όνομα. Κάθε repository μπορεί να έχει ένα ή περισσότερα branch. Ο κύριος branch είναι αυτός όπου όλες οι αλλαγές τελικά συγχωνεύονται .

4.4 Οι βασικές εντολές του Git

Για να χρησιμοποιήσουν το Git, οι προγραμματιστές χρησιμοποιούν συγκεκριμένες εντολές για αντιγραφή, δημιουργία, αλλαγή και συνδυασμό κώδικα. Αυτές οι εντολές μπορούν να εκτελεστούν απευθείας από τη γραμμή εντολών ή χρησιμοποιώντας μια εφαρμογή όπως το GitHub Desktop ή το Git Kraken. Ακολουθούν ορισμένες κοινές εντολές για τη χρήση του Git:

Το git init δημιουργεί ένα ολοκαίνουργιο Git repository και αρχίζει να παρακολουθεί έναν υπάρχοντα κατάλογο. Προσθέτει έναν κρυφό υποφάκελο στον υπάρχοντα κατάλογο που φιλοξενεί την εσωτερική δομή δεδομένων που απαιτείται για τον έλεγχο έκδοσης.

Το git clone δημιουργεί ένα τοπικό αντίγραφο ενός έργου που υπάρχει ήδη από απόσταση. Ο κλώνος περιλαμβάνει όλα τα αρχεία, το ιστορικό και τα branches.

Το git add. Το Git παρακολουθεί τις αλλαγές στη βάση κώδικα ενός προγραμματιστή, αλλά είναι σημαντικό κάθε αλλαγή που κάνετε καθώς προγραμματίζεται να είναι σαν ένα στάδιο , έτσι ώστε να τις συμπεριλάβετε στο ιστορικό του έργου. Αυτή η εντολή εκτελεί στάδια, το πρώτο μέρος αυτής της διαδικασίας δύο βημάτων. Τυχόν αλλαγές που πραγματοποιούνται θα γίνουν μέρος του επόμενου

στιγμιότυπου και μέρος του ιστορικού του έργου. Η σταδιοποίηση και το committing χωριστά δίνει στους προγραμματιστές πλήρη έλεγχο του ιστορικού του έργου τους χωρίς να αλλάζουν τον τρόπο με τον οποίο γράφουν κώδικα και λειτουργούν

Το git commit αποθηκεύει το στιγμιότυπο στο ιστορικό έργου και ολοκληρώνει τη διαδικασία παρακολούθησης αλλαγών. Εν ολίγοις, μια δέσμευση λειτουργεί όπως η λήψη φωτογραφίας. Οτιδήποτε έχει πραγματοποιηθεί με git add θα γίνει μέρος του στιγμιότυπου με git commit.

Η εντολή status git εμφανίζει την κατάσταση του καταλόγου εργασίας και τα στάδια ολοκλήρωσης ενός έργου. Σας επιτρέπει να δείτε ποιές αλλαγές έχουν πραγματοποιηθεί, ποιές όχι και ποιά αρχεία δεν παρακολουθούνται από το Git.

Το git branch δείχνει τα branches που δουλεύουν τοπικά.

Το git merge συγχωνεύει γραμμές ανάπτυξης μαζί. Αυτή η εντολή χρησιμοποιείται συνήθως για το συνδυασμό αλλαγών που γίνονται σε δύο ξεχωριστούς κλάδους. Για παράδειγμα, ένας προγραμματιστής θα συγχωνευθεί όταν θέλει να συνδυάσει αλλαγές από έναν κλάδο λειτουργιών στον κύριο κλάδο για ανάπτυξη.

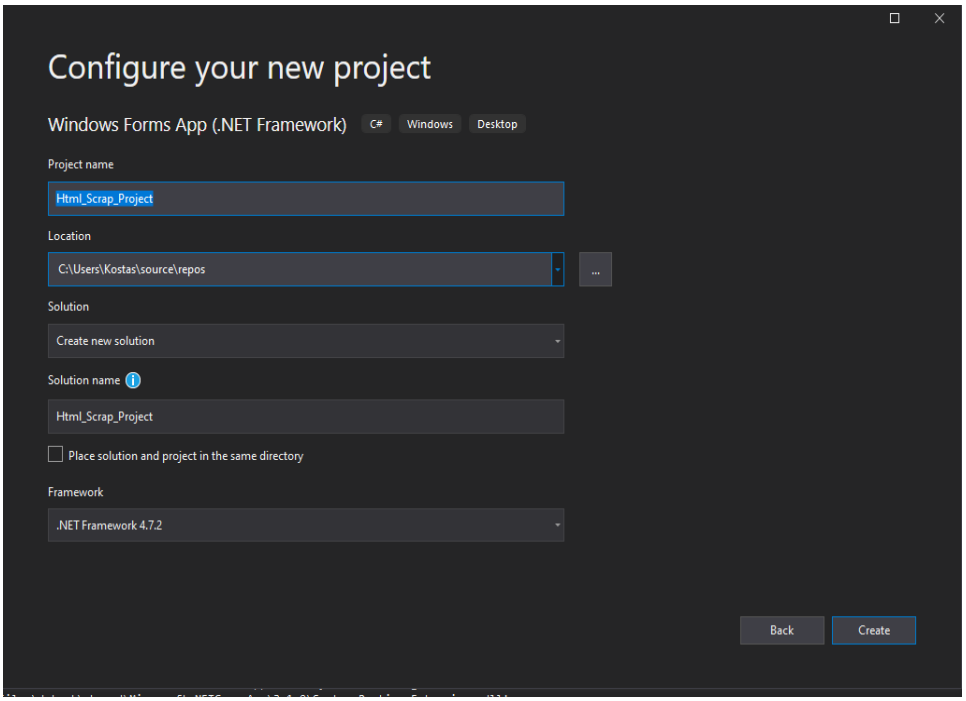
Το git pull ενημερώνει την τοπική γραμμή ανάπτυξης με ενημερώσεις από το απομακρυσμένο αντίστοιχο. Οι προγραμματιστές χρησιμοποιούν αυτήν την εντολή εάν ένας συμπαίκτης έχει δεσμευτεί σε ένα υποκατάστημα σε ένα τηλεχειριστήριο και θα ήθελε να αντικατοπτρίζει αυτές τις αλλαγές στο τοπικό τους περιβάλλον.

Το git push ενημερώνει το απομακρυσμένο αποθετήριο με οποιεσδήποτε δεσμεύσεις γίνονται τοπικά σε ένα υποκατάστημα

Τα βήματα που ακολούθησα για να φτιάξω το project και να το κάνω publish στον GitHub λογαριασμό μου :

Βήμα 1 : Δημιουργία του project . Είμαι μέσα στο visual studio και επιλέγω να ξεκινήσω ένα νέο project.

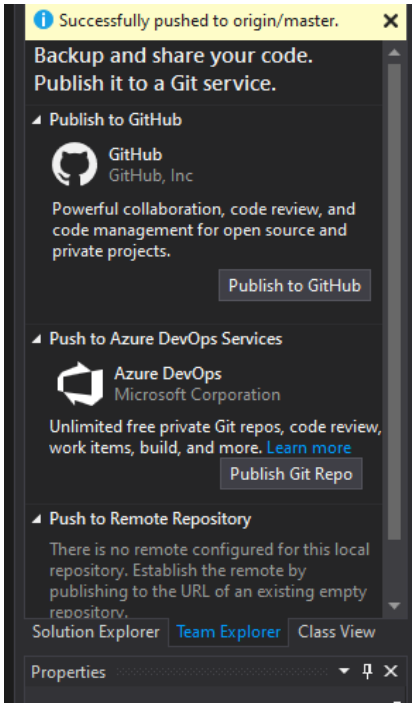
Σχήμα 4.1: New project form



Βήμα 2 : Αφού δημιουργηθεί το project και έχοντας πάντα εγκατεστημένο το GitHub extension και το git , δημιουργώ το repository πατώντας κάτω στην γωνία “add to source control” και επιλέγω git.

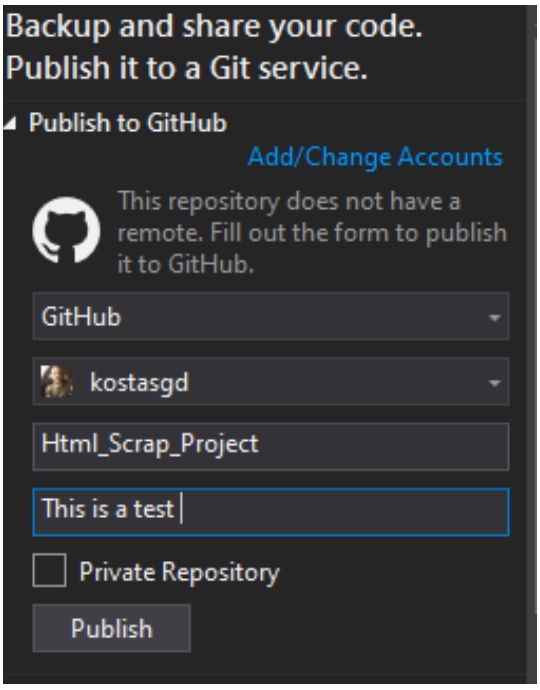
Βήμα 3: Αφού κάνω τα παραπάνω , μου ζητάει το git να κάνω publish το repository στον GitHub λογαριασμό μου.

Σχήμα 4.2: Publish στο Github



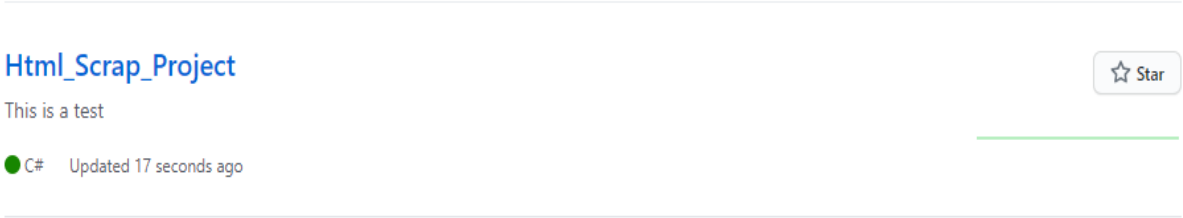
Με σύνδεση στον λογαριασμό μου δίνει την δυνατότητα για άμεση δημοσίευση του project , και να γράψω μία περιγραφή .

Σχήμα 4.3: Παράμετροι πριν το ανέβασμα στο github



Ακολουθώντας λοιπόν αυτά τα απλά βήματα , ανέβασα το Project στο προσωπικό μου λογαριασμό στο GitHub , και για επιβεβαίωση των βημάτων , το repository εμφανίστηκε στον κατάλογο που έχω όλα τα repositories.

Σχήμα 4.4: Επιβεβαίωση ανεβάσματος



Με την χρήση δικαιωμάτων μπορώ να προσθέσω κάποιο άτομο στο project , ώστε να παρακολουθεί την πορεία του project η ακόμα και να επέμβει αν χρειαστεί η να γράψει κώδικα.

Μέσα στα πλαίσια υλοποίησης ενός project , μπορεί να έρθεται σε αδιέξοδο σε κάποιο σημείο , είτε αυτό είναι κάποιο σφάλμα που προκλήθηκε κατά την πορεία η κάτι άλλο.Κάνοντας όμως τακτικά commit , μπορείτε να κάνετε **Revert** σε ένα προηγούμενο commit πού λογικά δεν θα έχει προκληθεί ακόμα αυτό το πρόβλημα.

4.5 Επίλογος

Απο αυτό το κεφάλαιο κρατάμε τον τρόπο με τον οποίο μπορείς να δουλέψεις ένα project τόσο έυκολα και απλά χρησιμοποιώντας τις βασικές εντολές του github , αλλά και την δημιουργία ενός τοπικού repository σε περίπτωση που δεν υπάρχει github λογαριασμός.

Κεφάλαιο 5ο: C#

5.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα γίνει ανάλυση της γλώσσας C# όπου θα εξηγηθούν οι τύποι , οι μεταβλητές , οι δομές δεδομένων,θα μάθουμε για το περιββάλον το οποίο αναπτύχθηκε το project ,αλλά και την λογική που δουλεύουν οι κύριοι μέθοδοι για να παραχθεί το επιθυμητό αποτέλεσμα.

5.2 C#

Η C# είναι μία ολοκληρωμένη αντικειμενοστραφής γλώσσα προγραμματισμού η οποία δημιουργήθηκε μέσα απο την πλατφόρμα .NET.Η γλώσσα με βάση τα χαρακτηριστικά της , τείνει να μοιάζει περισσότερο με την Cκαι C++.Το όνομα της είναι εμπνευσμενο απο μια μουσική σημειογραφία. Η C# μπορεί να χρησιμοποιηθεί για τη δημιουργία διαφόρων τύπων εφαρμογών, όπως web, windows, εφαρμογές κονσόλας ή άλλους τύπους εφαρμογών χρησιμοποιώντας το Visual studio. Ένα πρόγραμμα C # αποτελείται από τα ακόλουθα μέρη :

- Namespaces
- Κλάσεις
- Μέθοδοι κλάσης
- Χαρακτηριστικά κλάσης
- Μια κύρια μέθοδο
- Δηλώσεις και εκφράσεις
- Σχόλια

Ένα απλό πρόγραμμα στην C# έχει την ακόλουθη δομή :

```
using System;

namespace helloWorldApp{

        class helloWord{

                static void Main(String[] args){

                /* εκτύπωση μηνύματος  κονσόλας*/

                Console.WriteLine(“hello world”);

                }

        }

}
```

5.3 Visual studio

Η εφαρμογή που δημιούργησα είναι γραμμένη σε C# με την χρήση του προγράμματος Visual Studio 2019. Το Microsoft Visual Studio είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης της Microsoft. Χρησιμοποιείται για την ανάπτυξη προγραμμάτων ηλεκτρονικών υπολογιστών για λειτουργικά συστήματα, καθώς και web sites, εφαρμογών και υπηρεσιών web. Visual Studio χρησιμοποιεί πλατφόρμες ανάπτυξης λογισμικού της Microsoft, όπως τα Windows API, Windows

Forms(Αναφέρθηκα παραπάνω). Μπορεί να παράγει τόσο εγγενή κώδικα και διαχειριζόμενο κώδικα. Άλλα ενσωματωμένα εργαλεία που περιλαμβάνουν έναν σχεδιαστή έντυπα για τη δημιουργία εφαρμογών GUI, web designer, τάξη σχεδιαστής, σχεδιαστής και σχήματος βάσης δεδομένων. Δέχεται plugins που ενισχύουν τη λειτουργικότητα σχεδόν σε κάθε επίπεδο, συμπεριλαμβανομένων προσθέτοντας υποστήριξη για συστήματα πηγή ελέγχου (όπως η ανατροπή) και την προσθήκη νέων toolsets όπως εκδότες και visual designers για συγκεκριμένους τομείς γλώσσες ή toolsets για άλλες πτυχές του κύκλου ανάπτυξης λογισμικού (όπως τον πελάτη Team Foundation Server: Team Explorer). Το Visual Studio υποστηρίζει διαφορετικές γλώσσες προγραμματισμού και επιτρέπει το πρόγραμμα επεξεργασίας κώδικα και εντοπισμού σφαλμάτων για την υποστήριξη (σε διάφορους βαθμούς) σχεδόν οποιαδήποτε γλώσσα προγραμματισμού, εφόσον υπάρχει μια γλώσσα-ειδική υπηρεσία. Built-in γλώσσες περιλαμβάνουν C, C ++ και C ++ / CLI (μέσω Visual C ++), VB.NET (μέσω της Visual Basic .NET), C # (μέσω της Visual C #),.

5.4 Windows Forms

Οι φόρμες των Windows (WinForms) είναι μια δωρεάν βιβλιοθήκη ανοιχτού κώδικα γραφικών (GUI) που περιλαμβάνεται ως μέρος του Microsoft .NET Framework,παρέχοντας μια πλατφόρμα για την δημιουργία εφαρμογών για desktop,φορητά και tablet συστήματα.Η η μετεξέλιξη τους θεωρείται πως είναι οι WPF φόρμες οι οποίες είναι πιο σύγχρονες και περισσότερο συμβατές με τα σημερινά πρότυπα.

Για την υλοποίηση του project χρησιμοποίησα Windows Form και για να την κάνω πιο ελκυστική στον χρήστη χρησιμοποίησα μια βιβλιοθήκη του .NET Framework που ονομάζεται Material Design.Η βιβλιοθήκη μας παρέχει ορισμένα δικά της εργαλεία για να προσθέσουμε στην φόρμα. Οι επιλογές για το θέμα της φόρμας είναι ανοιχτό η σκοτεινό. Έπειτα δημιουργούμε ένα σύνολο χρωματικών επιλογών που καλύπτει κάθε αντικείμενο μέσα στην φόρμα προσέχοντας φυσικά να είναι όμορφα σαν σύνολο και εμφάνιση. Ο παρακάτω κώδικας τα κάνει όλα αυτά.

```
var skinmanager = MaterialSkin.MaterialSkinManager.Instance;

skinmanager.AddFormToManage(this);

skinmanager.Theme = MaterialSkin.MaterialSkinManager.Themes.DARK;

skinmanager.ColorScheme = new MaterialSkin.ColorScheme(MaterialSkin.Primary.BlueGrey800,
MaterialSkin.Primary.BlueGrey900,MaterialSkin.Primary.BlueGrey500,
MaterialSkin.Accent.LightBlue200, MaterialSkin.TextShade.WHITE);
```

5.5 Τύποι δεδομένων C#

Πίνακας 5.1: Τύποι δεδομένων και χαρακτηριστικά

Τύπος δεδομένων	Μέγεθος	Περιγραφή
int	4 bytes	Αποθηκεύει αριθμούς από -2.147.483.648 έως 2.147.483.647
long	8 bytes	Αποθηκεύει αριθμούς από -9.223.372.036.854.775.808 έως 9.223.372.036.854.775.807
float	4 bytes	Αποθηκεύει κλασματικούς αριθμούς. Αρκεί για την αποθήκευση 6 έως 7 δεκαδικών ψηφίων
double	8 bytes	Αποθηκεύει κλασματικούς αριθμούς. Αρκεί για την αποθήκευση 15 δεκαδικών ψηφίων
bool	1 bit	Αποθηκεύει αληθινές ή ψευδείς τιμές
char	2 bytes	Αποθηκεύει έναν μόνο χαρακτήρα/γράμμα, που περιβάλλεται από μεμονωμένα εισαγωγικά
string	2 bytes per character	Αποθηκεύει μια ακολουθία χαρακτήρων, που περιβάλλεται από διπλά εισαγωγικά

5.6 Δομές δεδομένων

Αφού η φάση της άντλησης πληροφοριών έχει τελειώσει , το ζήτημα πλέον που υπήρχε ήταν το πού θα αποθηκεύσω τις πληροφορίες αυτές.Η C# περιλαμβάνει εξειδικευμένες κλάσεις που αποθηκεύουν σειρά τιμών ή αντικειμένων που ονομάζονται συλλογές. Υπάρχουν δύο τύποι συλλογών στη C#: μη γενικές συλλογές και γενικές συλλογές. Το namespace System.Collections περιέχει τους μη γενικούς τύπους συλλογής και το System.Collections. Γενικός χώρος ονομάτων περιλαμβάνει γενικούς τύπους συλλογής. Στις περισσότερες περιπτώσεις, συνιστάται η χρήση των γενικών συλλογών, επειδή

αποδίδουν ταχύτερα από τις μη γενικές συλλογές και επίσης ελαχιστοποιούν τις εξαιρέσεις δίνοντας σφάλματα χρόνου μεταγλώττισης.

Αποφάσισα να αποθηκεύσω τις πληροφορίες αυτές σε μία List<string> καθώς οι περισσότερες πληροφορίες είναι σε μορφή κειμένου , αλλά και η αλλαγή σε άλλο τύπο είναι εύκολη. Μια List<T> περιέχει στοιχεία καθορισμένου τύπου πχ int , string,char. Αυξάνεται αυτόματα καθώς προσθέτετε στοιχεία σε αυτό. Υπάρχει μία άλλη δομή οι πίνακες που χρησιμοποιούνται για την αποθήκευση πολλαπλών τιμών σε μία μόνο μεταβλητή, αντί να δηλώνουν ξεχωριστές μεταβλητές για κάθε τιμή. Συνήθως τις αποφεύγουμε αλλά σε κάποιες περιπτώσεις μπορούν να φανούν χρήσιμες. Εκτός απο τις λίστες υπάρχουν και οι παρακάτω τύποι .

Πίνακας 5.2:Δομές δεδομένων C#

Γενικές συλλογές	**Περιγραφή**
Dictionary<TKey,TValue>	Το Dictionary <TKey, TValue> περιέχει ζεύγη τιμών-κλειδιών.
SortedList<TKey,TValue>	Η SortedList αποθηκεύει ζεύγη κλειδιών και τιμών. Προσθέτει αυτόματα τα στοιχεία σε αύξουσα σειρά του κλειδιού από προεπιλογή.
Queue<T>	Η Queue(ουρά) <T> αποθηκεύει τις τιμές σε στυλ FIFO (First In First Out). Διατηρεί τη σειρά με την οποία προστέθηκαν οι τιμές. Παρέχει μια μέθοδο Enqueue() για την προσθήκη τιμών και μια μέθοδο Dequeue() για την ανάκτηση τιμών από τη συλλογή.
Stack<T>	Η Stack(στοίβα) <T> αποθηκεύει τις τιμές ως LIFO (Last In First Out). Παρέχει μια μέθοδο Push() για να προσθέσετε μια τιμή και Pop() & Peek() μεθόδους για να ανακτήσετε τιμές.

5.7 Regular Expressions και επεξεργασία κειμένου

Μια κανονική έκφραση είναι μια ακολουθία χαρακτήρων που ορίζουν ένα μοτίβο αναζήτησης.Τέτοια μοτίβα λοιπόν χρησιμοποιούνται απο αλγόρυιμους για την αναζήτηση συμβολοσειρών.Μία από τις μεγαλύτερες δυσκολίες που αντιμετώπισα κατά την υλοποίηση του project , ήταν η σωστή εισαγωγή δεδομένων. Θέλοντας να πάρω τους συντελεστές μίας παράστασης , έπρεπε να δημιουργήσω μία έκφραση σε regular expression που θα έπαιρνε τις περιπτώσεις στις οποίες θα ήταν του στυλ [ρόλος] : [μέλος παράστασης] .Το regular expression που χρησιμοποίησα έχει την μορφή @"[\w]{1,}(.*):(.*)[\w]{1,}" όπου ουσιαστικά ταίριαξε με γραμμή που περιέχει μία η παραπάνω λέξεις πριν από την : (άνω κάτω τελεία) και με μία η παραπάνω λέξεις μετά την : .

Αποθηκεύω τον ρόλο και το μέλος παράστασης σε ξεχωριστές λίστες , αλλά προηγουμένως , στο loop που ελέγχει την κάθε γραμμή string έχω δημιουργήσει έναν πίνακα με λέξεις κλειδιά που θέλω να αγνοήσω. Έτσι με την μέθοδο Contains ελέγχω αν υπάρχει η ανεπιθύμητη λέξη αλλιώς γίνεται η εισαγωγή στις λίστες.

Το κείμενο που γινόταν η αποθήκευση , περιείχε άχρηστες εκφράσεις μπορούμε να τις ονομάσουμε ουσιαστικά σκουπίδια , τα οποία εμπεριεχόταν μαζί με τις πληροφορίες που γινόταν το Scraping. Μία ιδέα που εφαρμόστηκε και είχε αποτελέσματα ήταν η δημιουργία μίας μεθόδου που θα αφαιρούσε ότι Html σήμανση υπήρχε , και επέστρεφε σε string το κείμενο των συντελεστών με τους ρόλους τους.Ελέγχω αν το string είναι κενό αλλίως περνάω το link της παράστασης ώστε να φορτώσει το HtmlDocument το url. Δημιουργώ μία δομή ουράς που επιλέγει κάθε κόμβο που περιέχει κείμενο . Το μόνο tag που επιτρέπεται είναι το p που αντιστοιχεί στις παραγράφους , τις οποίες βρίσκονται στις πιο πολλές περιπτώσεις ενσωματωμένες οι πληροφορίες που χρειαζόμαστε.

5.8 LINQ

Το LINQ είναι μια ομοιόμορφη σύνταξη ερωτήματος σε C# και VB.NET για την ανάκτηση δεδομένων από διαφορετικές πηγές και μορφές.. Η LINQ επεκτείνει τη C# με την προσθήκη εκφράσεων ερωτήματος, οι οποίες είναι παρόμοιες με τις δηλώσεις SQL, και μπορεί να χρησιμοποιηθεί για την εύκολη εξαγωγή και επεξεργασία δεδομένων από πίνακες, αναρίθμητες τάξεις, έγγραφα XML, σχετικές βάσεις δεδομένων και πηγές δεδομένων τρίτων. Άλλες χρήσεις, οι οποίες χρησιμοποιούν εκφράσεις ερωτημάτων ως γενικό πλαίσιο για την ευανάγνωστη σύνθεση αυθαίρετων υπολογισμών, περιλαμβάνουν την κατασκευή χειριστών συμβάντων. Οι μέθοδοι LINQ που μου φάνηκαν ιδιαίτερα χρήσιμοι βρίσκονται στον παρακάτω πίνακα.

Πίνακας 5.3: Linq μέθοδοι

Μέθοδος	**Περιγραφή**
Contains()	Ελέγχει εάν υπάρχει συγκεκριμένο στοιχείο στη συλλογή αλλιώς επιστρέφει ένα δυαδικό.
Any()	Ελέγχει εάν κάποιο στοιχείο ικανοποιεί δεδομένη συνθήκη ή όχι
Count()	Ο τελεστής καταμέτρησης επιστρέφει τον αριθμό των στοιχείων στη συλλογή ή ταριθμό των στοιχείων που έχουν ικανοποιήσει τη δεδομένη συνθήκη.
Distinct()	Διαγράφει τα διπλότυπα απο μία συλλογή
ElementAt()	Ο τελεστής ElementAt ανακτά το στοιχείο σε ένα δεδομένο σημείο στη συλλογή.
ToList()	Μετατρέπει τη συλλογή σε λίστα

5.9 List View & Grid View

Τα C# Forms μας δίνουν δύο tools για παρουσίαση δεδομένων , το Grid View και το List View . Το GridView χρησιμοποιείται για την εμφάνιση των τιμών μιας πηγής δεδομένων σε έναν πίνακα. Κάθε στήλη αντιπροσωπεύει ένα πεδίο, ενώ κάθε σειρά αντιπροσωπεύει μια εγγραφή. Το στοιχείο ελέγχου GridView υποστηρίζει τις ακόλουθες δυνατότητες:

- Σύνδεση με στοιχεία ελέγχου πηγής δεδομένων, όπως SqlDataSource.
- Ενσωματωμένες δυνατότητες ταξινόμησης.
- Ενσωματωμένες δυνατότητες ενημέρωσης και διαγραφής.
- Ενσωματωμένες δυνατότητες σελιδοποίησης.
- Ενσωματωμένες δυνατότητες επιλογής σειράς.
- Πρόσβαση μέσω προγραμματισμού στο μοντέλο αντικειμένου GridView για δυναμικό ορισμό ιδιοτήτων, διαχείριση συμβάντων και ούτω καθεξής.
- Πολλαπλά βασικά πεδία.
- Πολλαπλά πεδία δεδομένων για τις στήλες υπερσύνδεσης.
- Προσαρμόσιμη εμφάνιση μέσω θεμάτων και στυλ.

Το στοιχείο ελέγχου ListView χρησιμοποιείται για την εμφάνιση των τιμών από μια πηγή δεδομένων. Μοιάζει με το στοιχείο ελέγχου Grid View, εκτός από το ότι εμφανίζει δεδομένα χρησιμοποιώντας πρότυπα που καθορίζονται από τον χρήστη αντί για πεδία σειράς. Η δημιουργία των δικών σας προτύπων σας δίνει μεγαλύτερη ευελιξία στον έλεγχο του τρόπου εμφάνισης των δεδομένων.

Το στοιχείο ελέγχου ListView υποστηρίζει τις ακόλουθες δυνατότητες:

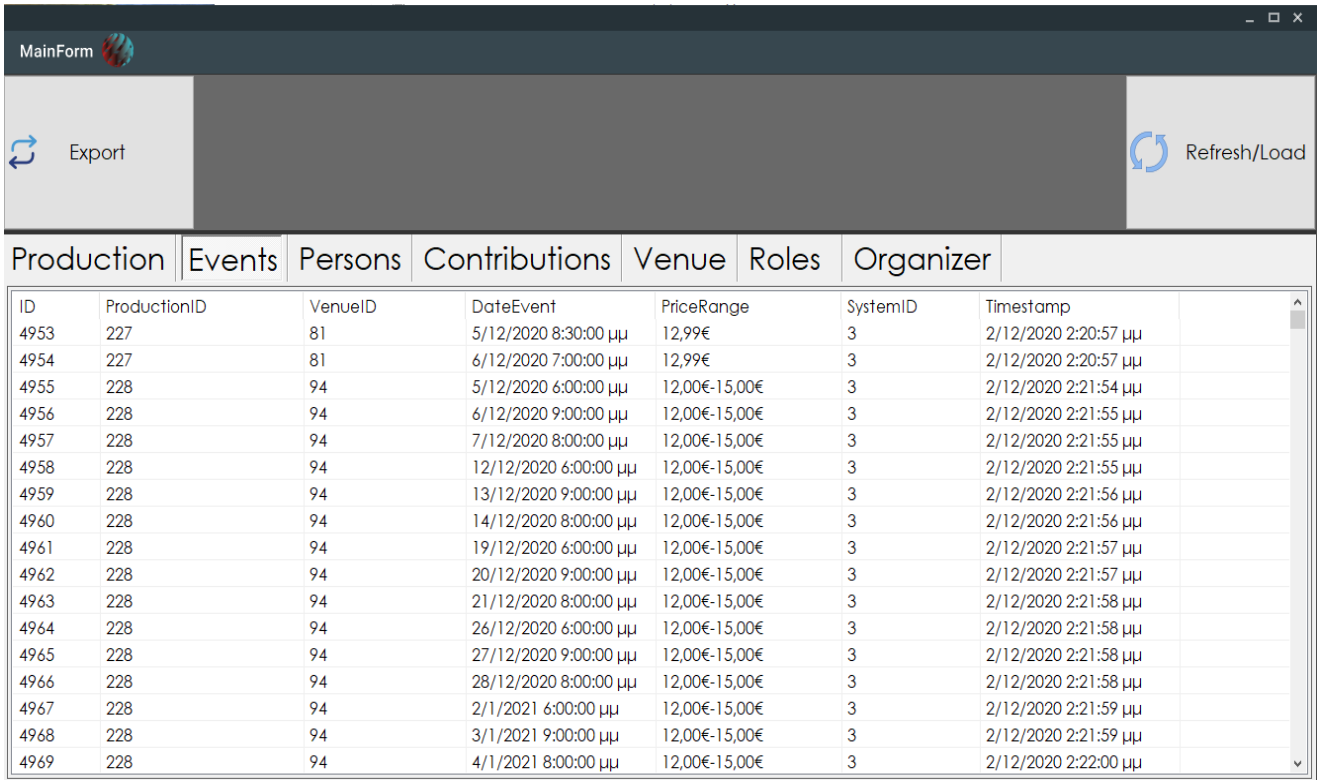
- Υποστήριξη για σύνδεση σε στοιχεία ελέγχου πηγής δεδομένων, όπως SqlDataSource, LinqDataSource και ObjectDataSource.
- Προσαρμόσιμη εμφάνιση μέσω καθορισμένων από το χρήστη προτύπων και στυλ.
- Ενσωματωμένες δυνατότητες ταξινόμησης.
- Ενσωματωμένες δυνατότητες ενημέρωσης και διαγραφής.
- Ενσωματωμένες δυνατότητες εισαγωγής.
- Υποστήριξη για δυνατότητες τηλεειδοποίησης χρησιμοποιώντας ένα στοιχείο ελέγχου DataPager.
- Ενσωματωμένες δυνατότητες επιλογής στοιχείων.
- Πρόσβαση μέσω προγραμματισμού στο μοντέλο αντικειμένου ListView για δυναμικό ορισμό ιδιοτήτων, διαχείριση συμβάντων και ούτω καθεξής.
- Πολλαπλά βασικά πεδία.

Εδώ έρχεται το ερώτημα , ποιό από τα δύο είναι καλύτερα. Η απάντηση είναι κανένα , γιατί τα χρησιμοποιούμε ανάλογα με το τι δυνατότητες θέλουμε να δώσουμε στον χρήστη. Αν θέλουμε να πειράξουμε τις εγγραφές , να προσθέσουμε μια , να διαγράψουμε , τότε το Grid View είναι η επιλογή μας. Αν θέλουμε κάτι γρήγορο , απλά να παρουσιάσουμε τα δεδομένα από την βάση η οπουδήποτε , τότε επιλέγουμε List View.

Λόγω του ότι κάποια πεδία περιέχουν μεγάλο μέγεθος κειμένου προς ανάγνωση , πράγμα που σημαίνει ότι πρέπει να υπάρξει επεξεργασία του List View ώστε η ανάγνωση των πεδίων να είναι ευχάριστη.Γι αυτό τον λόγο δημιούργησα μία μέθοδο που μεγαλώνει το πλάτος και το μήκος (Τα συγκεκριμένα χαρακτηριστικά δεν αναφέρονται στις ιδιότητες που υπάρχουν σε όλα τα controls του Visual Studio) εκεί που υπάρχει ανάγκη(Στο List View Production λόγω του πεδίου Description).

```
private void SetHeight(ListView listView, int height)
    {
       ImageList imgLst = new ImageList();
       imgLst.ImageSize = new Size(60, height);
       listView.SmallImageList = imgLst;
    }
```

Εδώ το ListView που περιέχει τις εγγραφές του Event table.



ID	ProductionID	VenueID	DateEvent	PriceRange	SystemID	Timestamp
4953	227	81	5/12/2020 8:30:00 μμ	12,99€	3	2/12/2020 2:20:57 μμ
4954	227	81	6/12/2020 7:00:00 μμ	12,99€	3	2/12/2020 2:20:57 μμ
4955	228	94	5/12/2020 6:00:00 μμ	12,00€-15,00€	3	2/12/2020 2:21:54 μμ
4956	228	94	6/12/2020 9:00:00 μμ	12,00€-15,00€	3	2/12/2020 2:21:55 μμ
4957	228	94	7/12/2020 8:00:00 μμ	12,00€-15,00€	3	2/12/2020 2:21:55 μμ
4958	228	94	12/12/2020 6:00:00 μμ	12,00€-15,00€	3	2/12/2020 2:21:55 μμ
4959	228	94	13/12/2020 9:00:00 μμ	12,00€-15,00€	3	2/12/2020 2:21:56 μμ
4960	228	94	14/12/2020 8:00:00 μμ	12,00€-15,00€	3	2/12/2020 2:21:56 μμ
4961	228	94	19/12/2020 6:00:00 μμ	12,00€-15,00€	3	2/12/2020 2:21:57 μμ
4962	228	94	20/12/2020 9:00:00 μμ	12,00€-15,00€	3	2/12/2020 2:21:57 μμ
4963	228	94	21/12/2020 8:00:00 μμ	12,00€-15,00€	3	2/12/2020 2:21:58 μμ
4964	228	94	26/12/2020 6:00:00 μμ	12,00€-15,00€	3	2/12/2020 2:21:58 μμ
4965	228	94	27/12/2020 9:00:00 μμ	12,00€-15,00€	3	2/12/2020 2:21:58 μμ
4966	228	94	28/12/2020 8:00:00 μμ	12,00€-15,00€	3	2/12/2020 2:21:58 μμ
4967	228	94	2/1/2021 6:00:00 μμ	12,00€-15,00€	3	2/12/2020 2:21:59 μμ
4968	228	94	3/1/2021 9:00:00 μμ	12,00€-15,00€	3	2/12/2020 2:21:59 μμ
4969	228	94	4/1/2021 8:00:00 μμ	12,00€-15,00€	3	2/12/2020 2:22:00 μμ

Σχήμα 5.1: List View Events

5.10 Προαπαιτούμενα του project

Καλώς η κακώς , τα περισσότερα προγράμματα που χρησιμοποιούμε στην καθημερινότητα μας απαιτούν κάποιες απαραίτητες προϋπόθεσεις για να μπορούν να λειτουργήσουν. Αυτό γίνεται για την αποφυγή σφαλμάτων πριν αλλά κατά την ώρα εκτέλεσης τους η για τον προφανές λόγο ότι είναι απαραίτητες. Το project που δημιούργησα δεν μπορεί να ξεκινήσει αν δεν υπάρχει σύνδεση με το διαδίκτυο , αλλά και επικοινωνία με την MySQL online βάση. Η online βάση μάλιστα πρέπει να ξέρει την IP του router για να μπορεί να δουλέψει η Remote MySQL.Αν έστω κάποιο απο τα δύο δεν μπορεί να λειτουργήσει , το πρόγραμμα πάυει να λειτουργεί .

Ο πιο απλός έλεγχος για να δείτε αν ο υπολογιστής σας επικοινωνεί με το διαδίκτυο είναι η παρακάτω μέθοδος :

```
private bool Ping()
    {
System.Net.NetworkInformation.Ping pingSender = new        System.Net.NetworkInformation.Ping();
System.Net.NetworkInformation.PingReply reply =   pingSender.Send("www.google.com");
if (reply.Status == System.Net.NetworkInformation.IPStatus.Success)
     {
         return true;
     }
     else
     {
           return false;
        }
}
```

Αντίστοιχα η πιό απλή μέθοδος για να δείτε αν υπάρχει επικοινωνία με την βάση (στην περίπτωση μας online βάση)

```
private bool Sql()
   {
     try
      {
  using (var connection = new MySqlConnection("…."))
      {
          connection.Open();
          return true;
      }
     }
     catch (MySql.Data.MySqlClient.MySqlException ex)
     {
        return false;
     }
   }
```

Το πρόγραμμα χρειάζεται συνεχής σύνδεση με το internet καθώς ελέγχει περιοδικά για νέες παραστάσεις.Για τον λόγο αυτό μέσα στον constructor της κεντρικής φόρμας , πρόσθεσα ένα κομμάτι κώδικα ο οποίος προσθέτει το ScrapMeNow στα προγράμματα που ξεκινάνε κατά την εκκίνηση.

```
var path = @"SOFTWARE\Microsoft\Windows\CurrentVersion\Run";
RegistryKey key = Registry.CurrentUser.OpenSubKey(path, true);
key.SetValue("ScrapMeNow", Application.ExecutablePath.ToString());
```

Αντικείμενα RegistryKey που αντιπροσωπεύουν τα ριζικά κλειδιά στο μητρώο των Windows και στατικές μεθόδους για πρόσβαση σε ζεύγη κλειδιών/τιμών.

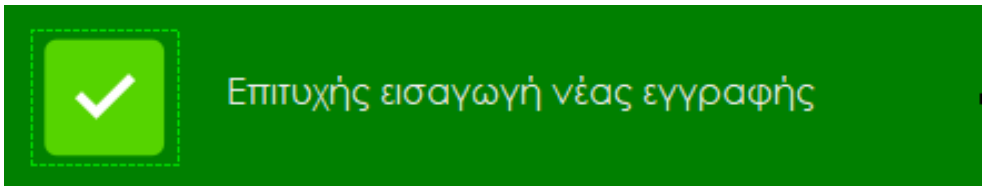
5.10.1 Project Loader

Πολλά προγράμματα της αγοράς αλλά και ιστοσελίδες κατά την εκκίνηση τους εμφανίζεται ένας loader. Είναι ένα από τα βασικά στάδια της διαδικασίας έναρξης ενός προγράμματος, καθώς τοποθετεί προγράμματα στη μνήμη και τα προετοιμάζει για εκτέλεση. Μόλις ολοκληρωθεί η φόρτωση, το λειτουργικό σύστημα ξεκινά το πρόγραμμα μεταβιβάζοντας τον έλεγχο στον φορτωμένο κωδικό προγράμματος. Αποφάσισα να φτιάξω έναν loader για το project θέλοντας το να δείχνει πιο όμορφο και προσεγμένο. Στην παρακάτω εικόνα φαίνεται ο loader του project :

Σχήμα 5.2 Application Loader

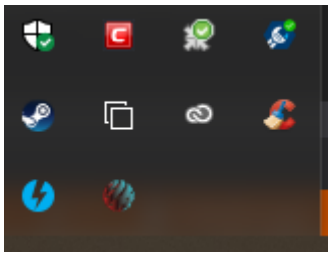
5.11 Notification Popup

Για τις ανάγκες ενημέρωσης του χρήστη για τυχόν νέα εισαγωγή στην βάση , δημιούργηθηκε ένα custom popup παράθυρο , το οποίο καλείται αφού τελειώσει η διαδικασία scraping και ενημερωθεί η βάση με νέα εγγραφή . Το παράθυρο αυτό εμφανίζεται κάτω δεξιά στην οθόνη του υπολογιστή.



Σχήμα 5.3 Επιβεβαίωση εισαγωγής εγγραφής

Σαν επιπλέον δυνατότητα , κάνοντας ελαχιστοποίηση του κεντρικού παραθύρου του project ,προστίθεται στην γραμμή εντολών το εικονίδιο της εφαρμογής , όπου κάνοντας το διπλό αριστερό κλικ το επαναφέρει στην κανονική του μορφή .



Σχήμα 5.4: Εικονίδιο ScrapMeNow στο hidden bar

5.12 Η λογική και εξήγηση μεθόδων με περίπτωση χρήσης

Ας πάρουμε την βασική περίπτωση εισαγωγής . Έχουμε με γεμίσει την MySQL βάση με όλες τις παραστάσεις. Υπάρχει μία μέθοδος που λέγεται checkNewLinks() και ελέγχει περιοδικά κάθε μία ώρα τα link που υπάρχουν στην σελίδα viva.gr/theater|theatre|stand_up_commedy με χρήση timer. Το στοιχείο Timer(χρονοδιακόπτη) είναι ένας χρονοδιακόπτης που δημιουργεί ένα συμβάν που έχει παρέλθει στην εφαρμογή μας μετά την παρέλευση του αριθμού χιλιοστών του δευτερολέπτου. Άν βρεί οτι ένα link δεν υπάρχει μέσα στην βάση προχωράει στην κλήση της μεθόδου insertProduction().Η μέθοδος για κάθε link που δεν υπάρχει στην βάση ,επιστρέφει μία λίστα με τα links αυτά. Έστω ότι βρίσκουμε ένα καινούργιο link τι θα συμβεί στην συνέχεια;Γίνεται έλεγχος αν το μέγεθος της checkNewLinks() είναι μεγαλύτερο απο το μηδέν και εφόσον είναι τότε καλείται η insertProduction() και ξεκινάει η εγγραφή.

Ξεκινώντας απο την insertProduction(), επειδή ο πίνακας Production έχει ξένο κλειδί το ID του πίνακα Organizer, ο πίνακας Organizer έχει προτεραιότητα εγγραφής στην βάση. Δημιουργώ ένα MySQL ερώτημα για το αν υπάρχει το ID του Organizer στην βάση ελέγχοντας το από τον τίτλο του .Άν υπάρχει τότε αποθηκεύω το ID αυτό σε μεταβλητή για να το χρησιμοποιήσω στην εισαγωγή το πίνακα Production , αλλιώς αν δεν υπάρχει , γίνεται η εισαγωγή της νέας εγγραφής στον πίνακα Organizer και παίρνω το MAX(ID) του πίνακα αυτού για να το χρησιμοποιήσω στην εισαγωγή του Production.

Έπειτα αφού γίνει η παραπάνω εισαγωγή , ελέγχεται αν η παράσταση υπάρχει , ελέγχοντας το αυτό από το url με MySQL ερώτημα.Εφόσον δεν υπάρχει τότε προχωράει στην εισαγωγή της παράστασης.Η άντληση δεδομένων τις παραπάνω περιπτώσεις(organizer , production)γίνεται με το HAP. Υπάρχουν πεδία που γεμίζουν από μεθόδους. Το Duration(διάρκεια) και το Media Url. Η getDuration(link) είναι μία απλή μέθοδος που επιστρέφει σαν string την διάρκεια του έργου σε λεπτά.Για να πάρω την εικόνα μιας παράστασης , στην GetMediaUrl γίνεται χρήση Selenium καθώς το HAP δεν μπορεί να έχει πρόσβαση σε δεδομένα παραγόμενα απο javascript.

Σε περίπτωση που υπάρχει βίντεο ,οπως ανέφερα πιο πάνω στις συμβουλές για χρήση Selector σε έναν ιστότοπο , αν δεν μπορείς πρέπει να σκεφτείς κάποιον εναλλακτικό τρόπο. Θέλωντας να βρω το link των παραστάσεων που έχουν video , ακολούθησα μία λογική η οποία δεν έβγαζε πουθενά. Έτσι αποφάσισα να χρησιμοποιήσω το HAP.Αυτό που παρατήρησα είναι ότι με την χρήση της μεθόδου της XPath text() μπορώ να πάρω τα πάντα από τον ιστότοπο.Όχι μόνο το κείμενο , την HTML , αλλά και το Script text που περιέχει. Μέσα στην JavaScript εμπεριέχονται πληροφορίες σχετικά με την παράσταση , μέσα σε αυτές και το link του βίντεο.Αλλά αυτό μόνο δεν είναι αρκετό, πρέπει με κάποιον

τρόπο να πάρω το link , το οποίο βρίσκεται μέσα σε string.Έτσι με την χρήση regex απομονώνω το link που ξεκινάει από youtube και το επιστρέφω.

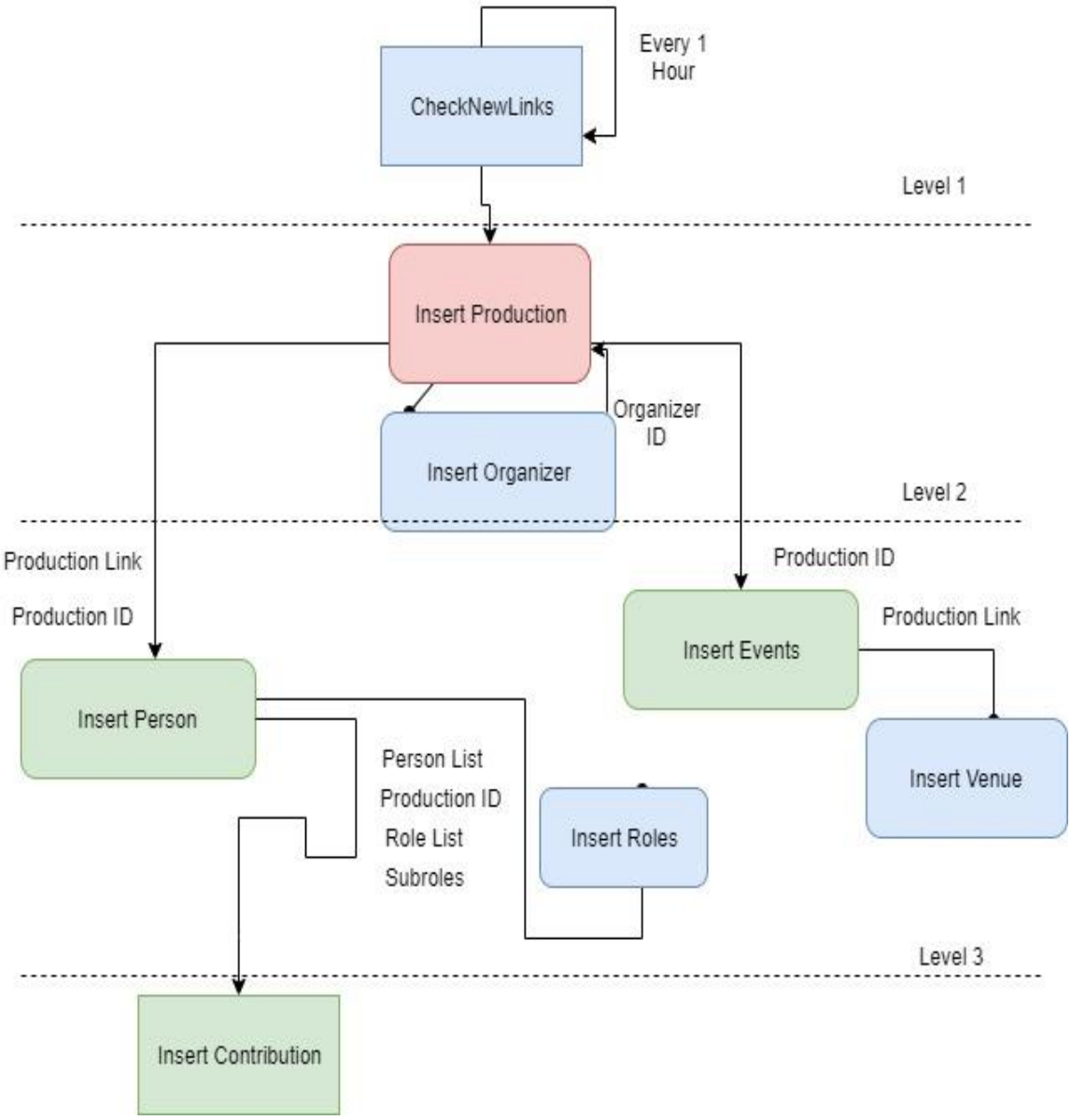
Μέθοδος για εισαγωγή Events :Η μέθοδος κάνει χρήση του Selenium Web Driver γιατί οι προβολές των παραστάσεων φορτώνουν με javascript .Με τις εντολές που έχω χρησιμοποιήσει σε προηγούμενα παραδείγματα και με ένα σύνολο λιστών παίρνω τα δεδομένα που θέλω. Γίνεται η επεξεργασία σε μερικά απο αυτά(ημερομηνία) και στην συνέχεια τα εισάγουμε στην βάση. Μόλις τελείωσει αυτό κλείνω τον Web Driver και συνεχίζει η εισαγωγή προβολών της επόμενης παράστασης. Γίνεται επίσης εισαγωγή στον πίνακα venue αν κάποιο θέατρο δεν είναι περασμένο στην βάση, καθώς το ID του θεάτρου είναι πεδίο του πίνακα events και μάλιστα ξένο κλειδί.

Σειρά έχει η insertPersons() όπου με την χρήση regular expression απομονωνονται οι ρόλοι με τα μέλη της παράστασης σε ξεχωριστές λίστες.Στην συνέχεια γίνεται η κατάλληλη επεξεργασία των λιστών αυτών ,όπως έλεγχος μοναδικότητας και έλεγχος ώστε να εισάγονται τα σωστά δεδομένα , με χρήση λέξεων κλειδιών προς αποφυγή.Και στην περίπτωση των ρόλων και των μέλων παράστασης , υπάρχει έλεγχος ώστε να μην μπαίνουν οι ίδιες εγγραφές παραπάνω απο μία φορά στην βάση.

Αφού γίνουν οι παραπάνω εισαγωγές τότε καλείται μέσα απο την insertPersons() η insertContributions().Μέσω παραμέτρων περνάμε τους ρόλους,τους ύπορολους, και τα μέλη σε λίστες.Εκεί υπάρχει ένα for loop όπου για κάθε ρόλο εισάγει το μέλος και τον υπορόλο αν υπάρχει αυτός.

Η ανίχνευση υπορόλων σε γενικές γραμμές είναι ακριβής αλλά όχι απόλυτα.Υπάρχει ένας αλγόριθμος ο οποίος ελέγχει αν υπάρχει ο ρόλος στην βάση , αλλιώς τον εισάγει σαν υπορόλο.Έχοντας έτσι περάσει τους βασικούς ρόλους που μπορούν να υπάρξουν σε ένα μεγάλο σύνολο παραστάσεων , αποφεύγετε η εισαγωγή <θεατρικών ρόλων> οι οποίοι θα γέμιζαν την βάση με επιπλέον ρόλους που η χρήση τους είναι αρκετά περιορισμένη .

Αφού περάσει και το τελευταίο στάδιο εισαγωγής στην βάση, σειρά έχει το notification popup το οποίο ενημερώνει τον χρήστη για την νέα εισαγωγή . Για να σβήσει το παράθυρο αυτό ο χρήστης αρκεί να πατήσει πάνω του.



Σχήμα 5.5 Ιεραρχία μεθόδων του προγράμματος

5.13 Επίλογος

Κλείνοντας το κεφάλαιο, μάθαμε το περιββάλον και την γλώσσα προγραμματισμού στην οποία υλοποιήθηκε η πτυχιακή.Πήραμε μία ιδέα με τις μεθόδους που έπαιξαν σημαντικό ρόλο , αλλά και τον τρόπο λειτουργίας ολόκληρου το application με τον οποίο επιτυγχάνουμε τον αρχικό μας στόχο.

Κεφάλαιο 6ο: Data Exporting

6.1 Εισαγωγή

Στο κεφάλαιο αυτό θα εξηγηθεί με λίγα λόγια τι είναι η εξαγωγή δεδομένων , σε τι μας χρησιμέυει ,θα παρουσιαστούν οι 3 τύποι που μπορεί να εξάγει το πρόγραμμα μας αλλά φυσικά και τα αντίστοιχα χαρακτηριστικά τους.

6.2 Data Exporting

Η εξαγωγή είναι η διαδικασία με την οποία μπορείς να εξάγεις δεδομένα μίας μορφής σε μία άλλη εύκολα και γρήγορα. Η εξαγωγή μπορεί να χρησιμοποιηθεί ως μέθοδος δημιουργίας αντιγράφων ασφαλείας σημαντικών δεδομένων ή μεταφοράς δεδομένων μεταξύ δύο διαφορετικών εκδόσεων προγραμμάτων. Για παράδειγμα, χρησιμοποιώντας το Microsoft Word μπορείτε να εξάγεται ένα αρχείο με κατάληξη .docx σε pdf , σε παλαιότερης έκδοσης αρχείο σε περίπτωση που χρειαστεί να το μεταφέρετε σε άλλον υπολογιστή που το χρησιμοποιεί.Ένας εξαγωγέας συχνά δεν είναι ένα ολόκληρο πρόγραμμα από μόνο του, αλλά μια επέκταση σε ένα άλλο πρόγραμμα, που εφαρμόζεται ως plug-in. Όταν υλοποιείται με αυτόν τον τρόπο, ο εξαγωγέας μετατρέπει την εγγενή μορφή της εφαρμογής αυτής , στην επιθυμητή μορφή και την γράφει σε αρχείο.

6.2.1 Παραδείγματα εξαγωγής δεδομένων

Υπάρχουν πολλοί λόγοι για τους οποίους μπορεί να χρειαστεί να εξαγάγετε δεδομένα από ένα πρόγραμμα. Ακολουθεί μια λίστα με τα πιο συνηθισμένα σενάρια εξαγωγής δεδομένων.

- Δημιουργία αντιγράφων ασφαλείας δεδομένων σε πρόγραμμα ή βάση δεδομένων. Για παράδειγμα, η εξαγωγή του βιβλίου διευθύνσεων ή του ηλεκτρονικού σας ταχυδρομείου από ένα πρόγραμμα ηλεκτρονικού ταχυδρομείου, ώστε να μπορεί να αποκατασταθεί εάν χαθεί.

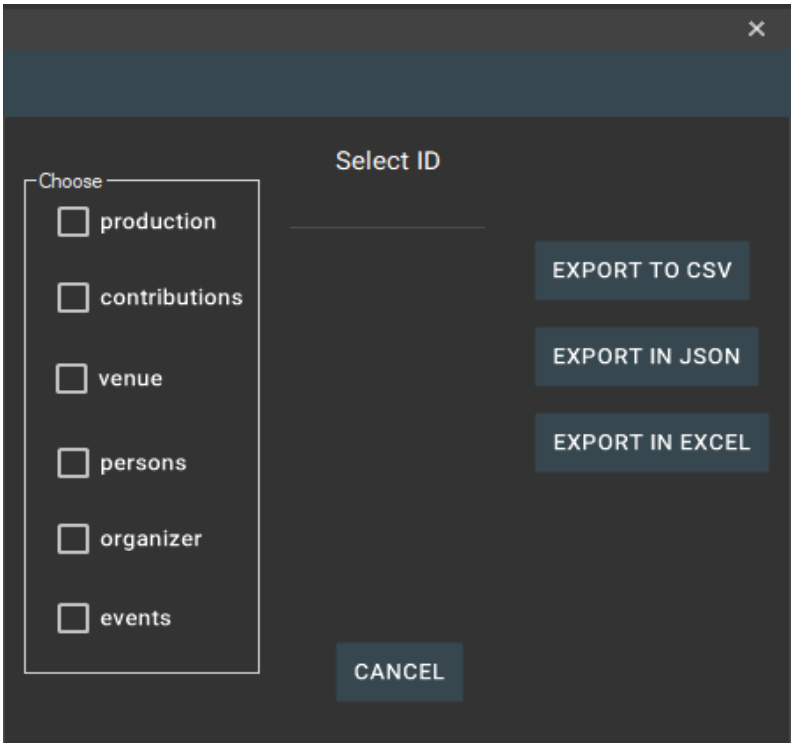
- Μετακίνηση δεδομένων από το ένα πρόγραμμα στο άλλο. Για παράδειγμα, κατά την αναβάθμιση σε νέα έκδοση προγράμματος ή την αλλαγή σε διαφορετικό προγραμματιστή λογισμικού (π.χ. αλλαγή προγράμματος περιήγησης).

- Αποθήκευση δεδομένων σε ένα πρόγραμμα σε μορφή συμβατή με διαφορετικό πρόγραμμα. Για παράδειγμα, η αποθήκευση ενός υπολογιστικού φύλλου Excel ως αρχείο CSV που μπορεί να εισαχθεί σε μια βάση δεδομένων.

- Αποθήκευση τμημάτων ενός μεγαλύτερου συνόλου δεδομένων. Για παράδειγμα, με μια βάση δεδομένων που περιέχει εκατομμύρια πελάτες, μπορείτε να εξαγάγετε δεδομένα για όλους τους χρήστες μιας συγκεκριμένης πολιτείας, ώστε να μπορούν να δημιουργηθούν αλληλογραφίες.

6.3 Export Form

Στην κεντρική φόρμα του project , υπάρχει ένα κουμπί που λέγεται Export. Πατώντας το εμφανίζεται μία φόρμα με checkboxes , το καθένα από τα οποία έχει ένα όνομα των πινάκων της βάσης. Υπάρχει επίσης ένα text πεδίο που πρέπει κάποιος να εισάγει το ID της εγγραφής που θέλει να κάνει εξαγωγή μίας εγγραφής. Η φόρμα είναι κάπως έτσι



Σχήμα 6.1 Export Form

Για να μπορεί όμως να γίνει το extracting έχω δημιουργήσει κάποιες μεθόδους που συμβάλουν στην διαδικασία αυτή. Η GetTable συγκεκριμένα παίρνει το τσεκαρισμένο checkbox επιστρέφοντας σε DataTable τύπο την Γραμμή του ID που ζητήθηκε. Ένα άλλο στοιχείο που μας παρέχει η C# είναι το SaveFileDialog που επιτρέπει στους χρήστες να περιηγούνται στο σύστημα αρχείων και να επιλέγουν αρχεία που θα αποθηκευτούν. Το πλαίσιο διαλόγου επιστρέφει τη διαδρομή και το όνομα του αρχείου που έχει επιλέξει ο χρήστης στο πλαίσιο διαλόγου.

Ένα αντικείμενο DataTable μοιάζει με έναν πίνακα βάσης δεδομένων και έχει μια συλλογή DataColumns (πεδία) και DataRows (Εγγραφές). Μπορεί επίσης να έχει ένα πρωτεύον κλειδί που βασίζεται σε μία ή περισσότερες στήλες και μια συλλογή αντικειμένων περιορισμού που είναι χρήσιμα για την επιβολή της μοναδικότητας των τιμών σε μια στήλη. Η πλειονότητα των εφαρμογών θα συμπληρώσει ένα Data Table απευθείας από μια βάση δεδομένων, ωστόσο είναι δυνατή η συμπλήρωση ενός πίνακα δεδομένων χρησιμοποιώντας κώδικα.

Μία μέθοδο για λόγους ελέγχου αλλά και για να ξέρουμε ποιό checkbox έχει τσεκαριστεί , ώστε να εκτελεστούν ενέργειες για το συγκεκριμένο, είναι η getCheckboxesChecked() η οποία για κάθε checkbox που υπάρχει στο group , ελέγχει και επιστρέφει το όνομα του checkbox που είναι checked.

6.4 Json

JSON είναι μια μορφή δεδομένων που βασίζεται σε κείμενο, ακολουθώντας τη σύνταξη αντικειμένων JavaScript. Διαδόθηκε από τον Douglas Crockford. Είναι βασισμένη σε ένα υποσύνολο της Javascript, αλλά μπορεί να χρησιμοποιηθεί τελειώς ανεξάρτητα από τη JavaScript σε πολλά περιβάλλοντα προγραμματισμού διαθέτουν τη δυνατότητα ανάγνωσης και δημιουργίας JSON. Χρήσιμο όταν θέλετε να μεταδώσετε δεδομένα σε ένα δίκτυο. Πρέπει να μετατραπεί σε εγγενές αντικείμενο JavaScript όταν θέλετε να αποκτήσετε πρόσβαση στα δεδομένα.

Χαρακτηριστικά της JSON μορφής :

- Η JSON είναι καθαρά μια συμβολοσειρά με καθορισμένη μορφή δεδομένων, περιέχει μόνο ιδιότητες, χωρίς μεθόδους.

- Η JSON απαιτεί διπλά εισαγωγικά για συμβολοσειρές και ονόματα ιδιοτήτων. Τα μεμονωμένα εισαγωγικά δεν είναι έγκυρα εκτός από το να περιβάλλουν ολόκληρη τη συμβολοσειρά JSON

- Ακόμη και ένα κόμμα ή άνω τελεία με λάθος θέση μπορεί να προκαλέσει λάθος στο αρχείο JSON και να μην λειτουργεί. Πρέπει να είστε προσεκτικοί για να επικυρώσετε τυχόν δεδομένα που προσπαθείτε να χρησιμοποιήσετε ,αν και το JSON που δημιουργείται από υπολογιστή είναι λιγότερο πιθανό να περιλαμβάνει σφάλματα, αρκεί το πρόγραμμα που το παράγει να λειτουργεί σωστά.

- Η JSON μπορεί στην πραγματικότητα να λάβει τη μορφή οποιουδήποτε τύπου δεδομένων που ισχύει για συμπερίληψη μέσα στο JSON, όχι μόνο πίνακες ή αντικείμενα. Έτσι, για παράδειγμα, μια μεμονωμένη συμβολοσειρά ή αριθμός θα ήταν έγκυρη JSON.

- Σε αντίθεση με τον κώδικα JavaScript στον οποίο οι ιδιότητες αντικειμένων ενδέχεται να είναι χωρίς εισαγωγικά, στο JSON μπορούν να χρησιμοποιηθούν μόνο συμβολοσειρές που αναφέρονται ως ιδιότητες.

Αυτή η μορφή είναι που αποθηκεύεται σε αρχείο , τα ονόματα των πεδίων μέσα σε " ", οι τιμές τους και αύτες σε " " με το σύμβολο : να τις ορίζει , χωρισμένα με κόμμα για να τα ξεχωρίζουμε.

[{"ID":99,"Name":"Α Μ ΤΕΧΝΗΧΩΡΟΣ ΕΤΕΡΟΡΡΥΘΜΗ ΕΤΑΙΡΙΑ",
"Address":"ΙΠΠΟΚΡΑΤΟΥΣ 7","Town":"ΑΘΗΝΑ",
"postcode":"10679",
"Phone":"-",
"Email":"texnixoros@gmail.com",
"Doy":"Α ΑΘΗΝΩΝ",
"Afm":"801009461",
"SystemID":3,
"timestamp":"2020-10-23T17:41:31"}]

Η μέθοδος που εξάγει αυτό το αρχείο είναι η DataTableToJson.Η μετατροπή γίνεται με χρήση library του .NET που ονομάζεται Newtonsoft.Json.

```
public string DataTableToJson(DataTable objDataTable)
    {
       string jsonString = string.Empty;
       jsonString = JsonConvert.SerializeObject(objDataTable);
       return jsonString;}
```

Η παραπάνω μέθοδος καλείται από την onClick μέθοδο του κουμπιού με όνομα "Write to JSON",Γεμίζουμε ουσιαστικά το DataTable με το αποτέλεσμα που επιστρέφει το SQL ερώτημα, η SerializeObject το παίρνει σαν μορφή αντικειμένου,το επεξεργάζεται και το επιστρέφει σαν JSON text.

6.5 CSV

Ένα αρχείο CSV είναι ένα αρχείο απλού κειμένου που περιέχει μια λίστα δεδομένων. Αυτά τα αρχεία χρησιμοποιούνται συχνά για την ανταλλαγή δεδομένων μεταξύ διαφορετικών εφαρμογών. Για παράδειγμα, οι βάσεις δεδομένων και οι διαχειριστές επαφών συχνά υποστηρίζουν αρχεία CSV. Αυτά τα αρχεία μπορεί μερικές φορές να ονομάζονται τιμές διαχωρισμένες με χαρακτήρες ή αρχεία οριοθετημένα με κόμματα. Χρησιμοποιούν τον χαρακτήρα κόμμα για να διαχωρίσουν δεδομένα, αλλά μερικές φορές χρησιμοποιούν άλλους χαρακτήρες, όπως τα ερωτηματικά. Η ιδέα είναι ότι μπορείτε να εξαγάγετε πολύπλοκα δεδομένα από μία εφαρμογή σε αρχείο CSV και στη συνέχεια, να εισαγάγετε τα δεδομένα σε αυτό το αρχείο CSV σε άλλη εφαρμογή.

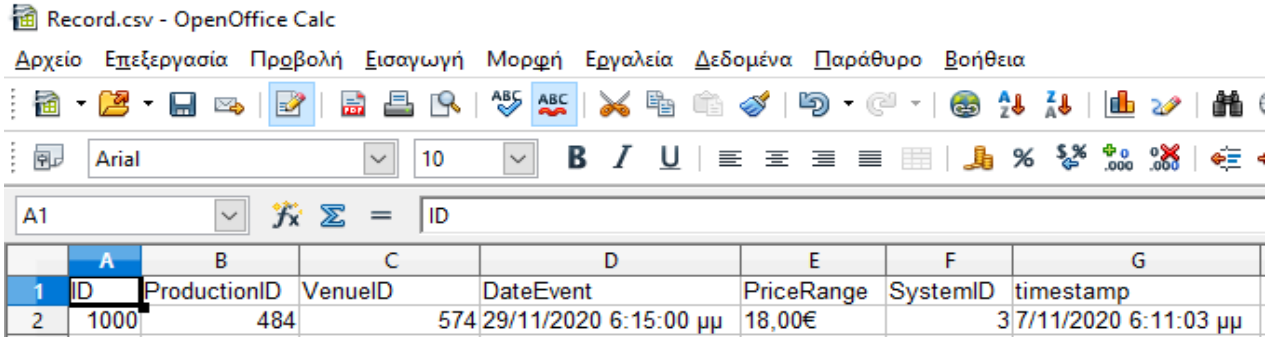
6.5.1 Η δομή ενός αρχείου CSV

Ένα αρχείο CSV έχει μια αρκετά απλή δομή. Πρόκειται για μια λίστα δεδομένων που διαχωρίζονται με κόμμα. Για παράδειγμα, ας υποθέσουμε ότι έχετε μερικές επαφές σε έναν διαχειριστή επαφών και τις εξάγετε ως αρχείο CSV. Θα λάβετε ένα αρχείο που περιέχει κείμενο ως εξής:

- Όνομα, email, αριθμός τηλεφώνου, διεύθυνση
- Kostas Georgiadis, kon_geo@gmail.com, 123-333-4444,123 Antipoleos 56
- Mike Zampidis, mikez@hotmail.com, 098-765-4321,321 Athinas 4

Όταν πατήσουμε το κουμπί που γράφει "Export to csv" τότε τρέχει κώδικας ο οποίος εφόσον του δώσουμε έγκυρο όνομα αρχείου , καλεί την WriteToCsvFile και αποθηκεύει το αποτέλεσμα της στο αρχείο που εμείς έχουμε ορίσει.

Συνιστάται , όταν ανοίγουμε αρχεία CSV να το κάνουμε με το Open Office η άλλο πρόγραμμα που αποδέχεται και μπορεί να προβάλει τέτοιου είδους αρχεία. Το αρχείο φαίνεται κάπως έτσι.



	A	B	C	D	E	F	G
1	ID	ProductionID	VenueID	DateEvent	PriceRange	SystemID	timestamp
2	1000	484	574	29/11/2020 6:15:00 μμ	18,00€	3	7/11/2020 6:11:03 μμ

Σχήμα 6.2 Csv αρχείο

6.6 Excel

Τα αρχεία με κατάληξη .xlsx και .xls είναι υπολογιστικά φύλλα Microsoft Excel που χρησιμοποιούνται συνήθως για την αποθήκευση οικονομικών δεδομένων και τη δημιουργία μαθηματικών μοντέλων. Τα αρχεία αποθηκεύουν δεδομένα σε φύλλα εργασίας που περιέχουν κελιά διατεταγμένα ως πλέγμα γραμμών και στηλών. Τα υπολογιστικά φύλλα του Excel μπορεί επίσης να περιέχουν γραφήματα, μαθηματικές συναρτήσεις και διάφορα είδη μορφοποίησης κειμένου. Τα υπολογιστικά φύλλα χρησιμοποιούνται συχνά σε επιχειρηματικά περιβάλλοντα για την αποθήκευση οικονομικών δεδομένων και για την εκτέλεση μαθηματικών υπολογισμών.

Αν θέλουμε να κάνουμε εξαγωγή δεδομένων σε excel τύπο , τότε ακολουθούμε τα βήματα όπως στις προηγούμενες περιπτώσεις.Στην κεντρική φόρμα επιλέγουμε το κουμπί στα αριστερά με όνομα export.Επιλέγουμε ένα απο τα checkboxes και συμπληρώνουμε το ID που επιθυμούμαι να εξάγουμε.Έτσι,αφού γίνονται οι απαραίτητοι έλεγχοι για τα checkboxes , σειρά έχει να γεμίσουμε με το SQL ερώτημα που θα επιστρέψει την σειρά για την οποία το ID είναι ίσο με την τιμή που δώσαμε στο text πεδίο. Έχουμε δώσει όνομα και τοποθεσία αρχείου για να ξέρουμε που βρίσκεται.Κάπως έτσι φαίνεται το αποτέλεσμα της μεθόδου με την χρήση του Microsoft Excel.

	A	B	C	D	E	F	G	H	I	J	K
1	ID	Name	Address	Town	postcode	Phone	Email	Doy	Afm	System	timestamp
2	90	ΘΕΑΤΡΙΚΕΣ ΣΚΗΝΕΣ ΑΕ	Αμερικής 9	ΑΘΗΝΑ	10672	2103639343	info@theatrikesskines.gr	Φ.Α.Ε ΑΘΗΝΩΝ	800744003	3	6/10/2020 14:17

Σχήμα 6.3 Αρχείο excel

6.7 Προτάσεις βελτίωσης Data Exporting

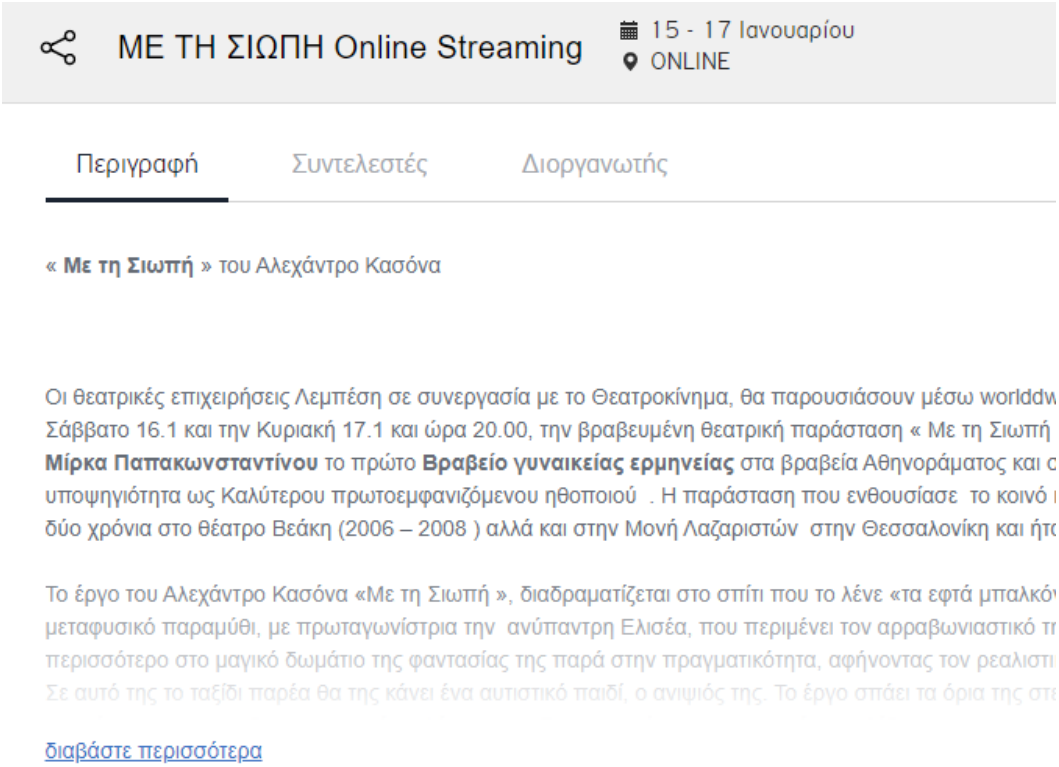
Ως πρώτη σκέψη για την εφαρμογή είναι η προσθήκη ακόμα περισσότερων τύπων αρχείου για εξαγωγή πράγμα που θα το κάνει ακόμα πιο ελκυστικό.Μία δεύτερη σκέψη ειναι η δημιουργία μίας μεθόδου όπου θα ενσωματώνει όλα τα χαρακτηριστικά μιας συγκεκριμένης παράστασης σε ένα κείμενο/συμβολοσειρά χωρίς να χρειαστεί να τσεκάρουμε πληροφορίες που σχετίζονται με εκείνη την παράσταση σε ξεχωριστά αρχεία

6.8 Επίλογος

Αυτό το κεφάλαιο δείχνει το πόσο χρήσιμη είναι η διαδικασία εξαγωγής δεδομένων , καθώς είναι η τελική φάση έχοντας ήδη αποθηκεύσει τα δεδομένα στην βάση.Εξηγήθηκαν οι τρείς τύποι εξαγωγής δεδομένων καθώς και τα επιμέρους χαρακτηριστικά τους.

Κεφάλαιο 7ο: Εξέταση αποτελεσμάτων

Φτάσαμε λοιπόν στο σημείο που θα πάρουμε μία περίπτωση εισαγώγής παράστασης για να δούμε την ακρίβεια των αποτελεσμάτων της Web Scraping διαδικασίας.Πιο συγκεκριμένα θα εξετάσουμε την παράσταση με url : https://www.viva.gr/tickets/theatre/streaming/me-ti-siopi/ . Αφού η μέθοδος checkNewLinks() ελέγχει ό,τι δεν υπάρχει το συγκεκριμένο link στην βάση , προχωράει στην εισαγωγή της παράστασης.

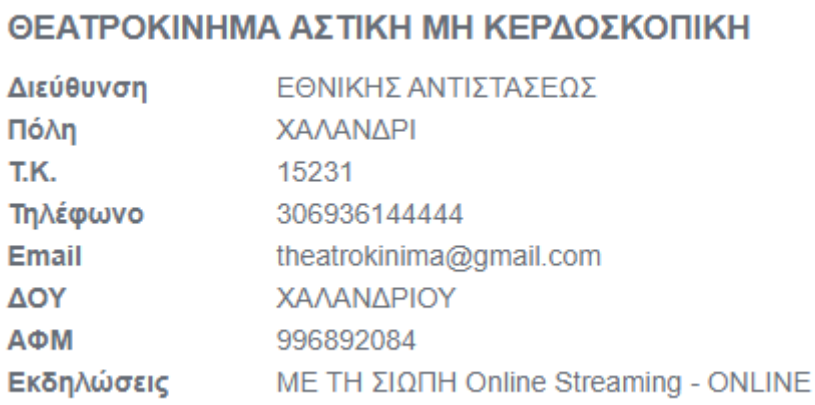


Σχήμα 7.1 Εικόνα παράστασης στον ιστότοπο

ID	OrganizerID	Title	Description	URL	Producer	MediaURL	Duration	SystemID	timestamp
400	149	ΜΕ ΤΗ ΣΙΩΠΗ Online Streaming	Με τη Σιωπή του Αλεχάντρο Κασόνα Οι θεατρικές	https://www.viva.gr/tickets/theatre/streaming/me-t...	ΘΕΑΤΡΟΚΙΝΗΜΑ ΑΣΤΙΚΗ ΜΗ ΚΕΡΔΟΣΚΟΠΙΚΗ		Not found	3	2021-01-07 15:38

Σχήμα 7.2 Εικόνα παράστασης στην βάση

Η παράσταση που θέλουμε να εξετάσουμε έχει το ID 400.Όπως μπορείτε να δείτε , τα πεδία Title ,Description , Url γεμίζουν απο τους κόμβους που έχουν βρεθεί χρησιμοποιώντας τον εντοπισμό στοιχείων.Το MediaUrl πεδίο είναι κενό καθώς δεν υπάρχει κάποιο Media για την συγκεκριμένη παράσταση.Επίσης το πεδίο duration δεν βρήκε διάρκεια της παράστασης ,πεδίο το οποίο γεμίζει σε σπάνιες περιπτώσεις.Οσο αφορά το OrganizerID και Producer ,τα πεδία αυτά έχουν προκύψει απο την εύρεση του ID του οργανωτή της παράστασης ,παίρνοντας έτσι το όνομα οργανωτή και το ID του ,αποθηκεύοντας τα σε μεταβλητές που στην συνέχεια τις ορίζουμε στις παραμέτρους εισαγωγής.

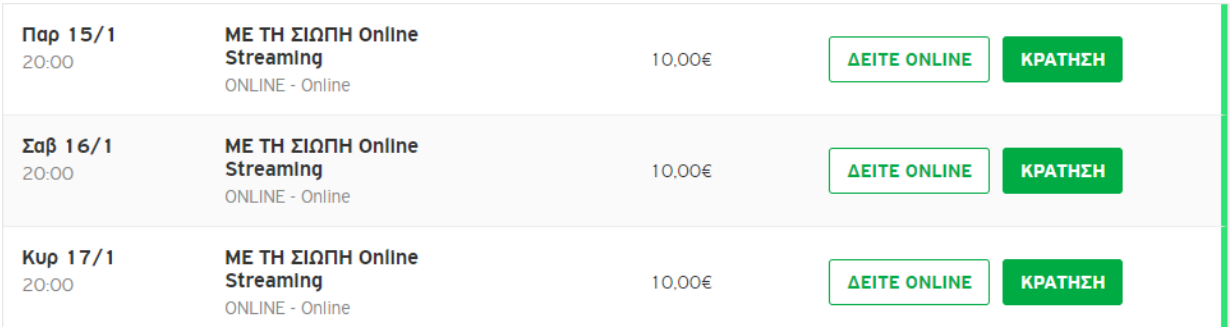


Σχήμα 7.3 Εικόνα οργανωτή στον ιστότοπο

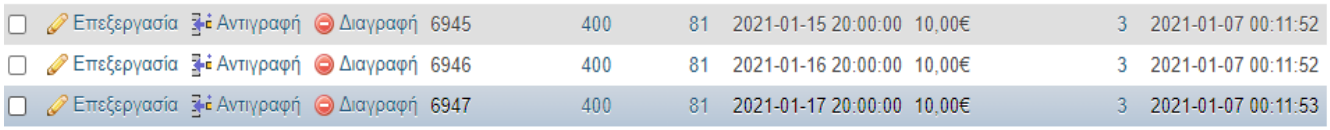
ID	Name	Address	Town	postcode	Phone	Email	Doy	Afm	SystemID	timestamp
149	ΘΕΑΤΡΟΚΙΝΗΜΑ ΑΣΤΙΚΗ ΜΗ ΚΕΡΔΟΣΚΟΠΙΚΗ	ΕΘΝΙΚΗΣ ΑΝΤΙΣΤΑΣΕΩΣ	ΧΑΛΑΝΔΡΙ	15231	306936144444	theatrokinima@gmail.com	ΧΑΛΑΝΔΡΙΟΥ	996892084	3	2021-01-06 21:40:53

Σχήμα 7.4 Εικόνα οργανωτή στην βάση

Προχωράμε για να εξετάσουμε τις προβολές των παραστάσεων της παράστασης.Το πεδίο με την ημερομηνία της παράστασης προφανώς περιέχει την χρονιά που θα γίνει. Τα άλλα πεδία εκτός απο το ID των αιθουσών είναι εμφανές ο τρόπος που γεμίζουν , η τιμή (10 ευρώ), οι ημερομηνίες (15,16,17/1).

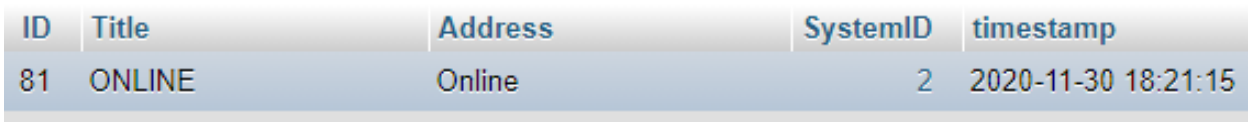


Σχήμα 7.5 Εικόνα προβολών στον ιστότοπο



Σχήμα 7.6 Εικόνα προβολών στην βάση

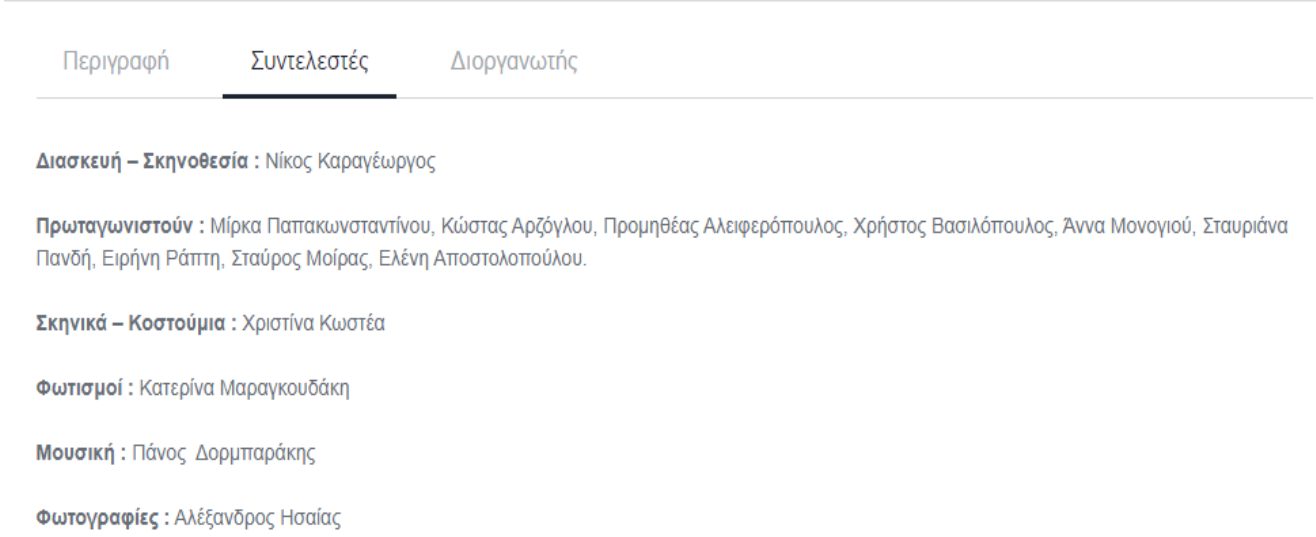
Το πεδίο με τον αριθμό 81 περιγράφει το ID του θετρικού χώρου που γίνεται η προβολή της παράστασης.Εδώ φαίνεται η εγγραφή με το ID που αναφέρθηκε προηγουμένως.



ID	Title	Address	SystemID	timestamp
81	ONLINE	Online	2	2020-11-30 18:21:15

Σχήμα 7.7 Εικόνα αίθουσας προβολής στην βάση

Στην συνέχεια έχουμε τα μέλη της παράστασης που εισάγονται στην βάση,γίνεται και η εισαγωγή των ρόλων.Στους δύο αυτούς πίνακες, αλλά και σε άλλους ύπαρχει έλεγχος ώστε να μην μπαίνουν ίδιες εγγραφές . Παίρνοντας μία περίπτωση ενός ατόμου με όνομα πχ Νίκος Παπαδόπουλος . Υπάρχει πιθανότητα αυτό το άτομο να έχει παραπάνω απο έναν ρόλους σε μία παράσταση. Μία ακόμα περίπτωση μπορεί να είναι η ύπαρξη ίδιου ονοματεπώνυμου σε διαφορετικές παραστάσεις.Έτσι άμα χρειαστεί να αναφερθεί το όνομα Νίκος Παπαδόπουλος σε παραπάνω απο μία παραστάσεις , χρησιμοποιούμε ένα sql query για να πάρουμε το ID του και να το χρησιμοποιήσουμε σε κάθε περίπτωση.



Σχήμα 7.8 Εικόνα συντελεστών και ρόλων στον ιστότοπο

Στήν βάση υπάρχουν απο προηγούμενες εισαγωγές , οι ρόλοι της πάρουσας παράστασης προς εξέταση.Εδώ θα τους βρούμε με τα ID τους.

	ID	Role	SystemID	timestamp
Επεξεργασία Αντιγραφή Διαγραφή	300	Μουσική	3	2020-10-08 14:07:29
Επεξεργασία Αντιγραφή Διαγραφή	332	Φωτισμοί	3	2020-10-10 23:27:45
Επεξεργασία Αντιγραφή Διαγραφή	333	Ηθοποιός	3	2020-10-10 23:27:47
Επεξεργασία Αντιγραφή Διαγραφή	351	Φωτογραφίες	3	2020-10-13 14:24:15
Επεξεργασία Αντιγραφή Διαγραφή	571	Σκηνικά , Κοστούμια	3	2020-10-26 19:11:11
Επεξεργασία Αντιγραφή Διαγραφή	837	Διασκευή , Σκηνοθεσία	3	2021-01-06 21:41:19

Σχήμα 7.9 Εικόνα ρόλων στην βάση

	ID	Fullname	SystemID	timestamp
Επεξεργασία		Μίρκα Παπακωνσταντίνου	3	2020-12-01 18:06:39
Επεξεργασία Αντιγραφή Διαγραφή	2400	Χριστίνα Κωστέα	3	2020-12-02 14:48:38
Επεξεργασία Αντιγραφή Διαγραφή	2883	Κατερίνα Μαραγκουδάκη	3	2020-12-02 15:29:07
Επεξεργασία Αντιγραφή Διαγραφή	3116	Νίκος Καραγέωργος	3	2021-01-06 21:41:19
Επεξεργασία Αντιγραφή Διαγραφή	3117	Πάνος Δορμπαράκης	3	2021-01-06 21:41:19
Επεξεργασία Αντιγραφή Διαγραφή	3118	Αλέξανδρος Ησαίας	3	2021-01-06 21:41:19
Επεξεργασία Αντιγραφή Διαγραφή	3129	Κώστας Αρζόγλου	3	2021-01-06 22:34:46
Επεξεργασία Αντιγραφή Διαγραφή	3130	Προμηθέας Αλειφερόπουλος	3	2021-01-06 22:34:47
Επεξεργασία Αντιγραφή Διαγραφή	3131	Χρήστος Βασιλόπουλος	3	2021-01-06 22:34:47
Επεξεργασία Αντιγραφή Διαγραφή	3132	Άννα Μονογιού	3	2021-01-06 22:34:47
Επεξεργασία Αντιγραφή Διαγραφή	3133	Σταυριάνα Πανδή	3	2021-01-06 22:34:47
Επεξεργασία Αντιγραφή Διαγραφή	3134	Ειρήνη Ράπτη	3	2021-01-06 22:34:47
Επεξεργασία Αντιγραφή Διαγραφή	3135	Σταύρος Μοίρας	3	2021-01-06 22:34:47
Επεξεργασία Αντιγραφή Διαγραφή	3136	Ελένη Αποστολοπούλου.	3	2021-01-06 22:34:47

Πατήστε το προς τα κάτω βέλος για εναλλαγή της εμφάνισης στηλών.

Σχήμα 7.10 Εικόνα person στην βάση

Ακολουθεί μία απο τις δύσκολες διαδικασίες η οποία <ταιριάζει> ουσιαστικά τους ρόλους με τα μέλη της παράστασης για να έχουμε μία ολοκληρωμένη εικόνα .Όπως ανάφερα ένα άτομο μπορεί να έχει παραπάνω απο έναν ρόλο.Το ίδιο συμβαίνει εξίσου με τους ρόλους καθώς ένας ρόλος μπορεί να έχει πολλούς ανθρώπους .Μία συνηθισμένη περίπτωση είναι οι ηθοποιοί σαν γενική έννοια.

	ID	PeopleID	ProductionID	RoleID	subRole	SystemID	timestamp
Επεξεργασία Αντιγραφή Διαγραφή	4399	3116	400	837		3	2021-01-07 00:11:58
Επεξεργασία Αντιγραφή Διαγραφή	4400	2192	400	333		3	2021-01-07 00:11:59
Επεξεργασία Αντιγραφή Διαγραφή	4401	3129	400	333		3	2021-01-07 00:11:59
Επεξεργασία Αντιγραφή Διαγραφή	4402	3130	400	333		3	2021-01-07 00:11:59
Επεξεργασία Αντιγραφή Διαγραφή	4403	3131	400	333		3	2021-01-07 00:12:00
Επεξεργασία Αντιγραφή Διαγραφή	4404	3132	400	333		3	2021-01-07 00:12:00
Επεξεργασία Αντιγραφή Διαγραφή	4405	3133	400	333		3	2021-01-07 00:12:00
Επεξεργασία Αντιγραφή Διαγραφή	4406	3134	400	333		3	2021-01-07 00:12:01
Επεξεργασία Αντιγραφή Διαγραφή	4407	3135	400	333		3	2021-01-07 00:12:01
Επεξεργασία Αντιγραφή Διαγραφή	4408	3136	400	333		3	2021-01-07 00:12:01
Επεξεργασία Αντιγραφή Διαγραφή	4409	2400	400	571		3	2021-01-07 00:12:02
Επεξεργασία Αντιγραφή Διαγραφή	4410	2883	400	332		3	2021-01-07 00:12:02
Επεξεργασία Αντιγραφή Διαγραφή	4411	3117	400	300		3	2021-01-07 00:12:02
Επεξεργασία Αντιγραφή Διαγραφή	4412	3118	400	351		3	2021-01-07 00:12:03

Σχήμα 7.11 Εικόνα contributions στην βάση

7.1 Επίλογος

Εδώ κάπου τελείωνει η εξέταση των αποτελεσμάτων.Βλέποντας τις προηγούμενες εικόνες βγάζουμε το συμπέρασμα ότι οι εγγραφές έγιναν σωστά. Να αναφέρουμε ότι δεν υπάρχει κάποιος θεατρικός ρόλος ο οποίος θα γέμιζε το πεδίο subRole , έτσι παραμένει κενό(βλέπε πίνακα contributions).Μπορεί κάποιος να συμπεράνει ότι τελικά δεν είναι τόσο δύσκολο να αντλήσεις δεδομένα απο έναν ιστότοπο.Αυτό που παίζει μεγάλο ρόλο βέβαια είναι και το τί δεδομένα είναι αυτά . Ανάλογα τα δεδομένα που τραβάμε απαιτούνται οι κατάλληλες ενέργειες και στην περίπτωση μας οι ενέργειες αυτές ήταν σε κανονικό επίπεδο.Εν τέλει , δεν πάυει να είναι μία αξιόπιστη τεχνική που θα σώσει αρκετό χρόνο δουλειάς και θα κάνει την ζωή των χρηστών ευκολότερη.

Κεφάλαιο 8ο: Συμπεράσματα και προτάσεις βελτίωσης

Το ScrapMeNow έχει ολοκληρωθεί με επιτυχία με τα αποτελέσματα του να είναι ικανοποιητικά.Κατά την διάρκεια υλοποίησης της πτυχιακής έμαθα πολλά πράγματα σχετικά για τις δυνατότητες της τεχνικής Web Scraping , ενώ παράλληλα βελτίωσα τις γνώσεις μου στην γλώσσα C# . Έμαθα να χειρίζομαι και να εξάγω τα δεδομένα που αποθηκεύτηκαν καλύπτοντας αν όχι όλες τις περισσότερες προδιαγραφές ενός Web Scraping εργαλείου.

Το πρόγραμμα αφού σχεδιάστηκε και υλοποιήθηκε , έχει προγραμματιστεί έτσι ώστε όλα να γίνονται αυτόματα , χωρίς την παρέμβαση του χρήστη.Ο χρήστης έχει άμεση εικόνα και ενημέρωση για τυχόν νέα δεδομένα που αποθηκεύονται βλέποντας τα απο την κεντρική οθόνη του προγράμματος και εναλλακτικά απο την MySQL βάση.Αυτό που γίνεται με χειροκίνητη επιλογή χρήστη είναι η ανανέωση του List View αλλά και η εξαγωγή δεδομένων απο την βάση σε αρχεία.

Καταλήγω στο οτί είναι ένα τέτοιο εργαλείο μπορεί να φανεί αρκετά χρήσιμο και διευκολύνει σε μεγάλο βαθμό την ζωή του χρήστη.Μία σκέψη βελτίωσης θα ήταν να επεκταθεί και σε άλλες σελίδες θεατρικών παραστάσεων ,προσφέροντας μεγαλύτερο όγκο δεδομένων.Επίσης για πιο ακριβή και σωστή εισαγωγή στοιχείων , θα μπορούσε να υπάρχει η και να φτιαχτεί , ένας αλγόριθμος αναγνώρισης ρόλων για να αποφευχθεί ένα σύνολο λάθος εγγραφών στην βάση , που μπορεί να δημιουργήσει ασάφειες.

Συνοπτικά,έχοντας βγάλει ένα συμπέρασμα για τον κλάδο , μπορεί κανείς να καταλήξει στο συμπέρασμα ότι η λίστα με τα πράγματα που μπορείς να κάνεις με το Web Scraping είναι αμέτρητα, έχει να κάνει με το τι μπορείς να κάνεις με τα δεδομένα που έχεις συλλέξει, και το πόσο χρήσιμα-πολύτιμα μπορείς να τα κάνειςΜπορεί κανείς να βρεί την παρουσίαση την πτυχιακής εργασίας μου στο link : https://www.youtube.com/watch?v=BU7AiKtL9pU&t=3s

Βιβλιογραφία & Πηγές

What is Web Scraping and What is it Used For? | Definition and Examples EXPLAINED
https://www.youtube.com/watch?v=Ct8Gxo8StBU&t=4s

Web Scraping vs Web Crawling

https://dzone.com/articles/web-scraping-vs-web-crawling-whats-the-difference

Top 7 Web Scraping Tips

https://www.scraping-bot.io/top-7-web-scraping-tips/

Τα πάντα για το HAP

https://html-agility-pack.net/online-examples

GitHub οδηγός

https://guides.github.com/introduction/git-handbook/

Screen Scraping

https://searchdatacenter.techtarget.com/definition/screen-scraping

https://prowebscraper.com/blog/screen-scraping/

Selenium

https://www.guru99.com/selenium-tutorial.html

https://www.selenium.dev/documentation/en/

SQL by Wikipedia

https://el.wikipedia.org/wiki/SQL

SQL Prepared statement

https://docs.microsoft.com/en-us/dotnet/api/system.data.sqlclient.sqlcommand.prepare?view=dotnet-plat-ext-5.0

C# generaly, arrays , data types

https://www.w3schools.com/cs/cs_arrays.asp

https://en.wikipedia.org/wiki/C_Sharp_(programming_language)

https://www.w3schools.com/cs/cs_data_types.asp

ListView vs GridView

https://www.c-sharpcorner.com/forums/difference-between-gridview-and-listview

Data Exporting

https://www.computerhope.com/jargon/e/export.htm

Data Tables

https://bettersolutions.com/csharp/databases/datatable.htm

Web Scraping Overview

https://www.scrapinghub.com/what-is-web-scraping/

Open Data

https://opendatahandbook.org/guide/el/what-is-open-data/