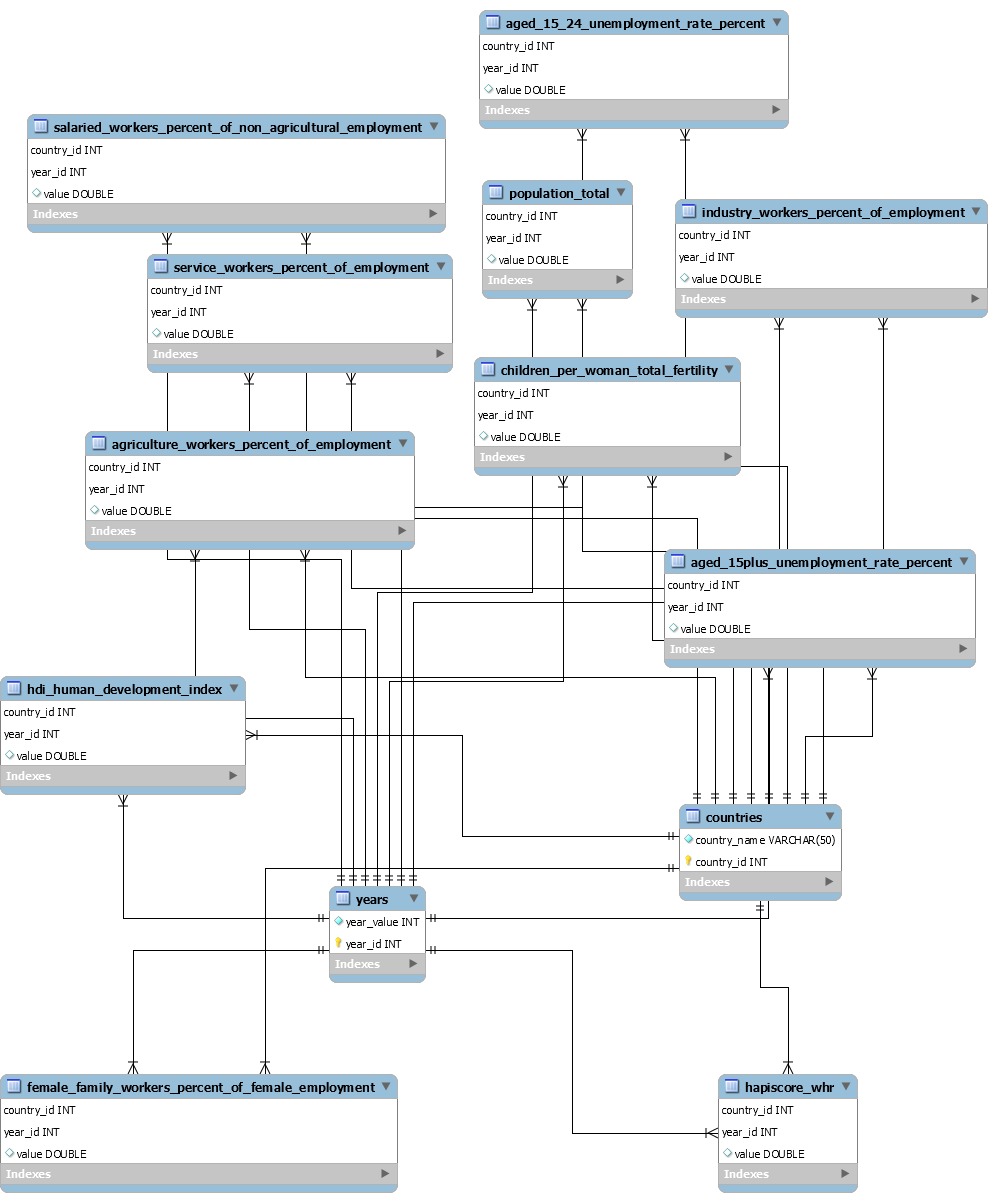
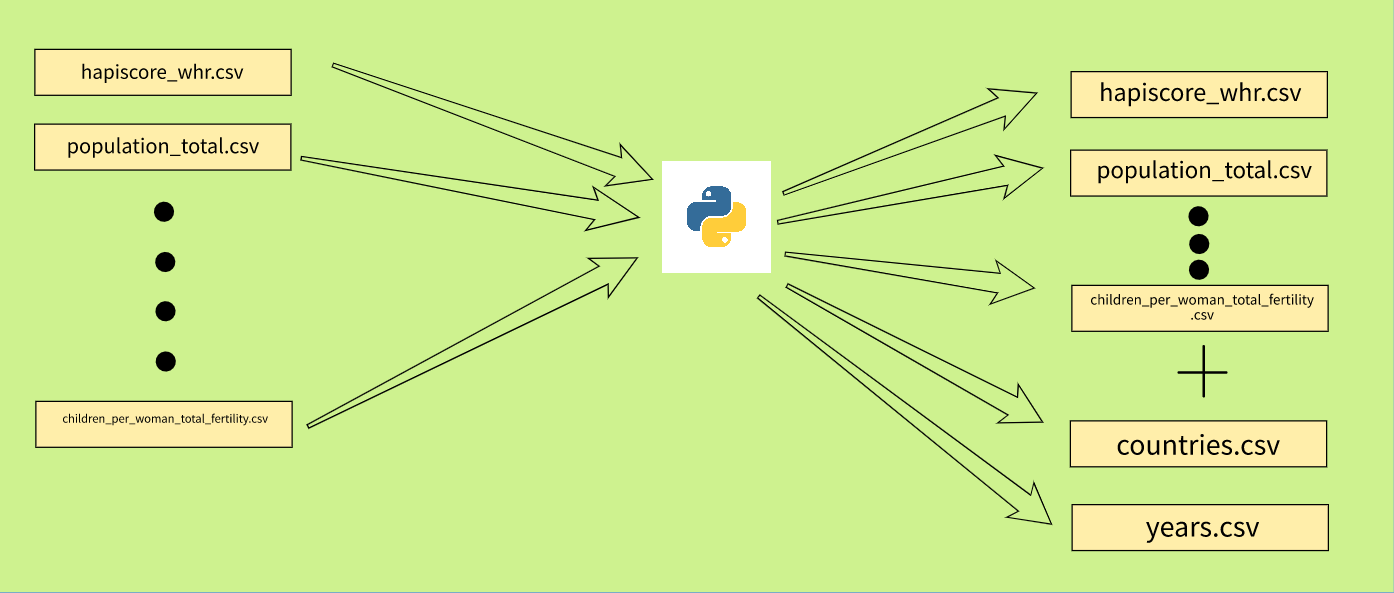
Λάσκας Πέτρος 3019 – Περικλής Μασούρας 3024 12/05/2021

**Προχωρημένα θέματα τεχνολογίας και εφαρμογών βάσεων δεδομένων**

**PROJECT**

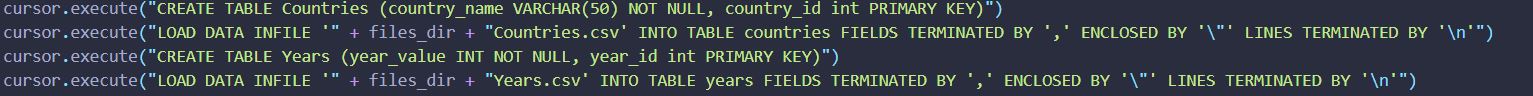
 Ξεκινώντας, το σχήμα της βάσης που επιλέξαμε να σχεδιάσουμε είναι το παρακάτω.

Για να δημιουργήσουμε το παραπάνω σχήμα επεξεργαστήκαμε τα δεδομένα μας με την βοήθεια Python scripts.

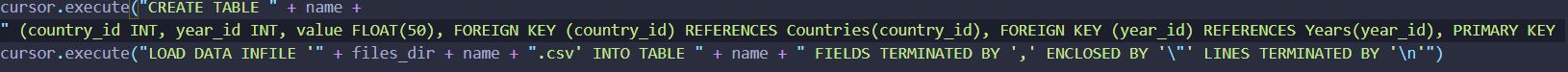


Ουσιαστικά τα αρχεία του gapminder τα μετατρέψαμε σε «πίνακες» όπου η κάθε γραμμή έχει τρείς τιμές country\_id, year\_id, value. Επιπλέον δημιουργήσαμε 2 ακόμα αρχεία Countries.csv και Years.csv με κάθε γραμμή να περιέχει country\_id, country και year\_id, year αντίστοιχα. Η παραπάνω διαδικασία βρίσκεται στο αρχείο fileprocess.py.

Στην συνέχεια για να μπορέσουμε να φορτώσουμε τα αρχεία στην βάση δεδομένων MySQL χρησιμοποιήσαμε πάλι Python, για να συνδεθούμε, να δημιουργήσουμε τους πίνακες και να φορτώσουμε τα δεδομένα σε αυτούς. Παρακάτω ακολουθεί ένα βασικό κομμάτι του κώδικα.

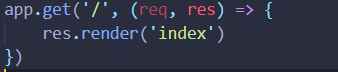


Εικόνα 1 Δημιουργία Πινάκων Countries & Years

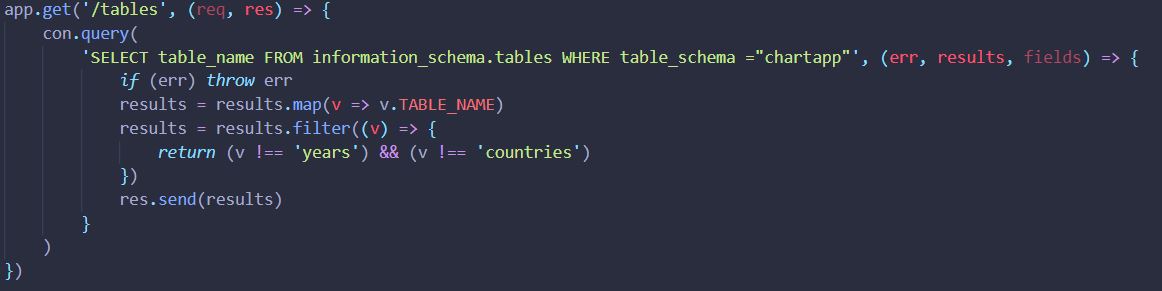


Εικόνα 2 Δημιουργία πινάκων για τα υπόλοιπα δεδομένα (ΔΕΝ ΦΑΙΝΕΤΑΙ ΟΛΟΚΛΗΡΟ).

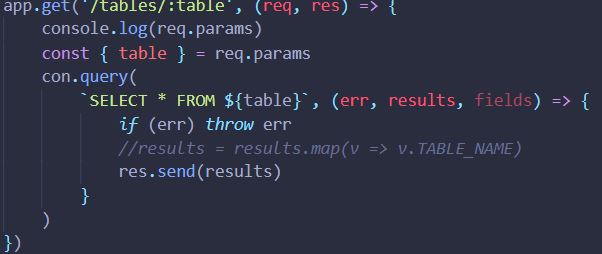
Γιατί επιλέξαμε αυτό το σχήμα; Απλά γιατί δεν είχαμε εμπειρία. Τι κερδίσαμε; Στο συγκεκριμένο πρότζεκτ πρακτικά τίποτα, ίσα ίσα έκανε την ζωή μας δυσκολότερη λόγο των πολύπλοκων queries με τα πολλαπλά JOINS. Γιατί δεν το αλλάξαμε; Γιατί αφού τελικά γράψαμε τα queries, δεν υπάρχει λόγος αλλαγής. Ούτε χάνουμε ούτε κερδίζουμε κάτι για το συγκεκριμένο πρότζεκτ από πλευρά απόδοσης. Αν ξανακάναμε κάτι αντίστοιχο σίγουρα θα σκεφτόμασταν κάποια άλλη «απλούστερη» σχεδίαση. Η σχεδίαση που επιλέξαμε εμείς θα βόλευε αν υπήρχαν περισσότερα χαρακτηριστικά, όπως πρωτεύουσες στο πίνακα countries ή αν θελήσουμε στο μέλλον να προσθέσουμε χαρακτηρίστηκα βάζοντας απλά ένα νέο πίνακα, φυσικά θα χρειαστεί η κατάλληλη προεπεξεργασία. Ολόκληρος ο κώδικας για την δημιουργία της βάσης βρίσκεται στο αρχείο dbmsload.py.

 Το stack που χρησιμοποιήσαμε για την δημιουργία του web app είναι Node.js + Express.js για το server και HTML + Javascript + Bulma CSS framework + EJS template για το client side, τέλος η d3.js είναι η βιβλιοθήκη που επιλέξαμε για να δημιουργήσουμε τα γραφήματα. Ξεκινώντας με το server, το σύνολο του κώδικά του βρίσκεται στο αρχείο index.js. Όταν ο server ξεκινάει αρχικά, συνδέεται στο database και στην συνέχεια ακούει στο localhost:3000. Έχουμε δημιουργήσει τέσσερα routes :

Βασικό route, κάνει render to αρχικό menu από το index.ejs και το παραδίδει στον client όταν αυτός ζητήσει το localhost:3000.



Ζητάει από την βάση όλους τους διαθέσιμους πίνακες για να τους εμφανίσει στο αρχικό menu.

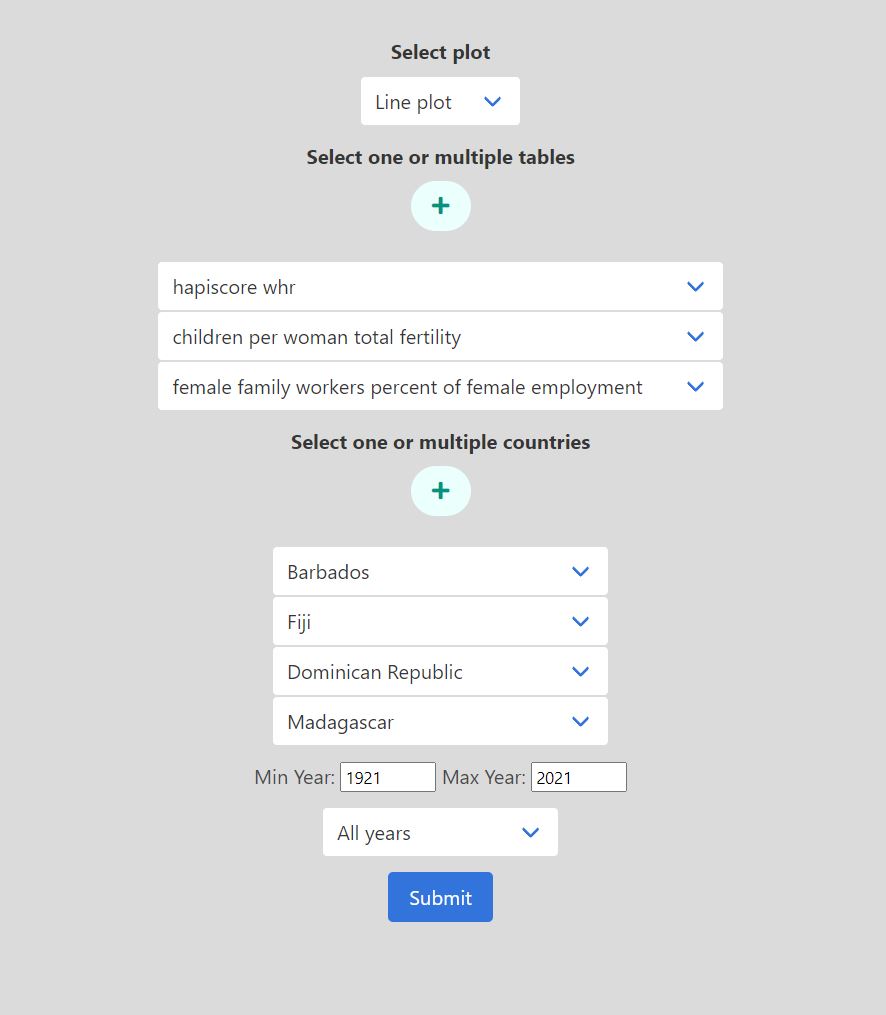


Ζητάει από την βάση ένα ολόκληρο table, το χρησιμοποιούμε για να βρούμε όλες τις διαθέσιμες χώρες και να τις εμφανίσουμε στο αρχικό menu.



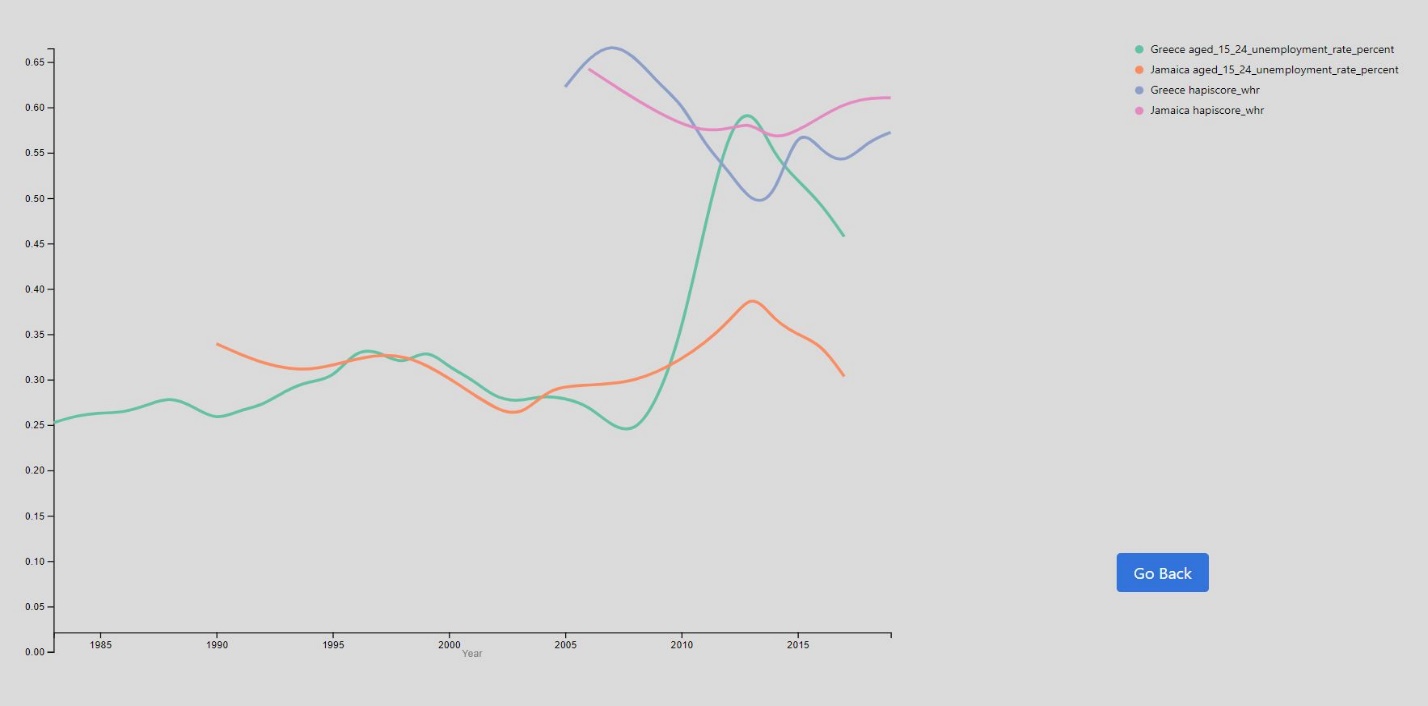
Το βασικό route το οποίο απαντάει στο get request του form από το βασικό menu. Χτίζει το κατάλληλο request με βάση το αίτημα του χρήστη, κάνει render το show.ejs και το επιστρέφει μαζί με κατάλληλες παραμέτρους και δεδομένα (για αυτό το λόγο κάναμε χρήση του ejs template, εύκολη εισαγωγή των δεδομένων στη d3) για την σωστή λειτουργία της βιβλιοθήκης d3.js. Επίσης στο server υλοποιείται η συνάρτηση groupby η οποία ανάλογα την επιλογή του χρήστη κάνει group τα δεδομένα ανά δεκαετία ή πενταετία. Για περισσότερες λεπτομέρειες σχετικά με την λειτουργία του server δείτε το αρχείο.

Για το Front-end, όλα τα αρχεία βρίσκονται στο φάκελο Views. Εκεί θα βρούμε τα index.ejs, show.ejs και app.js . To index.ejs περιέχει όλο τον HTML κώδικα για την δημιουργία του form. Επιπλέον κάναμε χρήση του CSS framework bulma για λόγους ευκολίας, κυρίως στην διάταξη των στοιχείων του form, για λόγους «ομορφίας» και τέλος γιατί είναι responsive για οποιοδήποτε μέγεθος οθόνης( υπολογιστή, τάμπλετ, κινητό…).

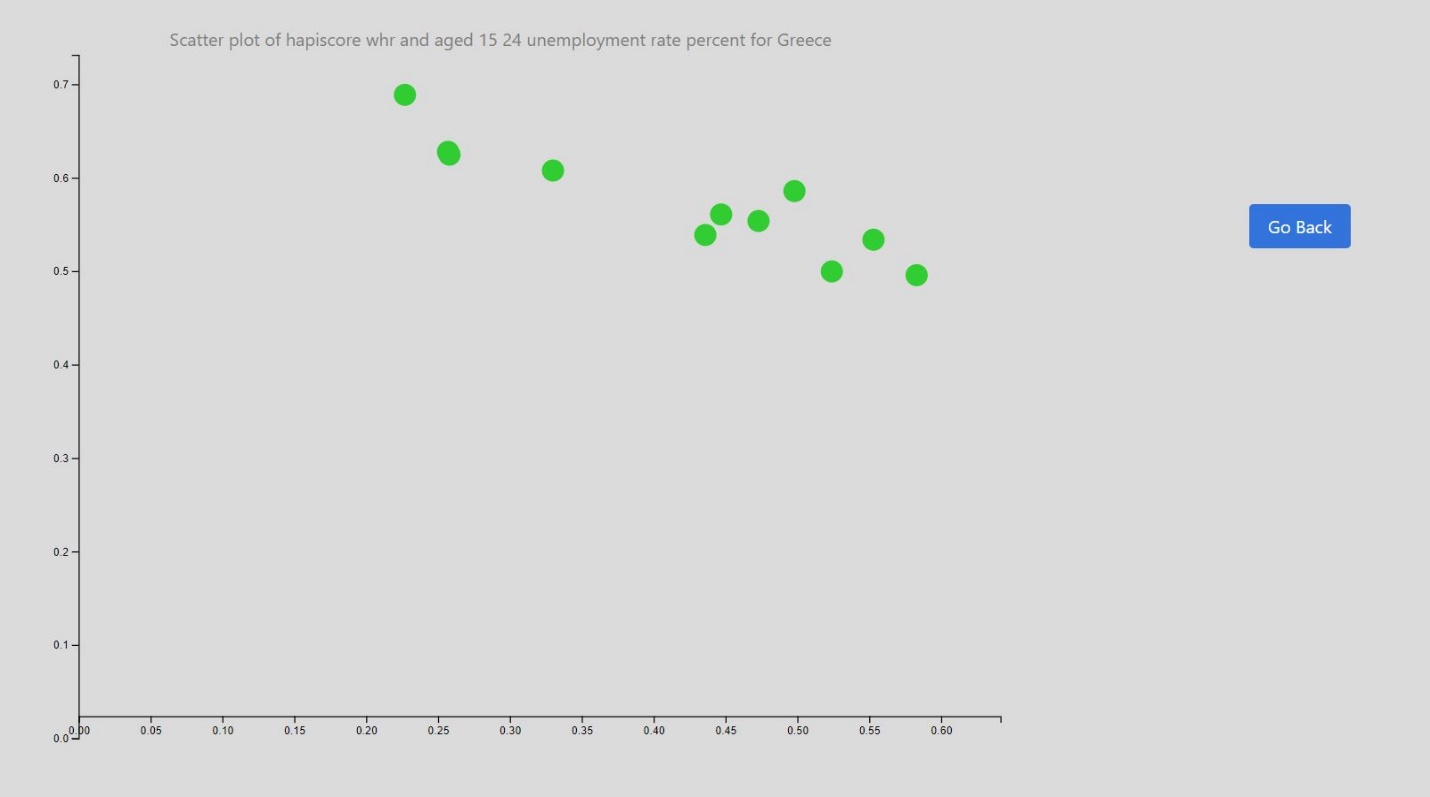


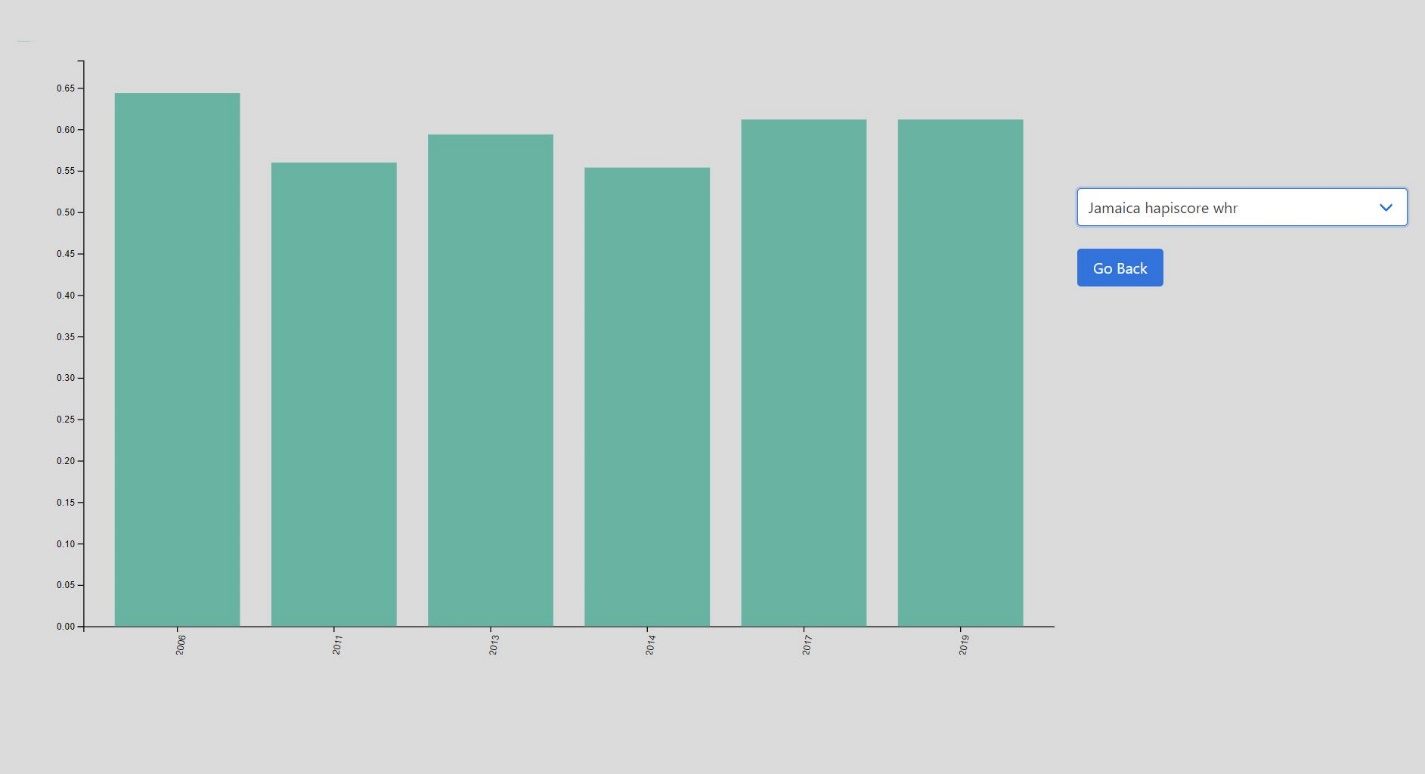
Παραπάνω βλέπουμε ένα screenshot από το menu. Για line και bar plot επιτρέπουμε έως 5 πίνακες και έως 5 χώρες, επίσης μπορούμε να επιλέξουμε ένα εύρος για τα χρόνια αλλά και αν θέλουμε να τα ομαδοποιήσουμε ανά δεκαετία και πενταετία (παίρνουμε μέσο όρο). Για το Scater plot αναγκαστικά έχουμε μόνο 2 πίνακες και μόνο μία χώρα. Περισσότερα για την χρήση του μενού στο βίντεο. Ο κώδικας javascript για την λειτουργικότητα του βρίσκεται στο αρχείο app.js.

Τέλος έχουμε το αρχείο show.ejs στο οποίο βρίσκεται ο κώδικας για την δημιουργία της σελίδας αλλά και για λόγους ευκολίας ο κώδικας της d3.js ο οποίος φτιάχνει το κατάλληλο σχήμα με βάση τα δεδομένα που έχει λάβει από το server. Ακολουθούν φωτογραφίες από τα τρία πλότ.



Line plot: Για κάθε συνδυασμό χώρας και πίνακα δημιουργεί μία γραμμή μα διαφορετικά χρώματα. Στα δεξιά υπάρχει Legend με τα χρώματα και κουμπί για επιστροφή στο form. Πέρα από το plot το legend και το κουμπί είναι επίσης responsive, δηλαδή σε κινητό θα αλλάξουν θέση και θα μπουν κάτω από το plot.





Bar plot: Δεν τα βάλαμε όλα σε ένα γιατί δεν μας φάνηκε όμορφο. Δεξιά υπάρχει ένα select το οποίο περιέχει όλες τις επιλογές του χρήστη και αλλάζει δυναμικά το πλοτ. Περισσότερα στο βίντεο.

Πως το τρέχουμε ;

1. Θέλουμε εγκαταστημένη python
2. Θέλουμε εγκατεστημένο Node.js και npm
3. Ετοιμάζουμε αρχεία τρέχοντας το fileprocess.py (το αποφευγουμε αυτό γιατί έχουμε έτοιμα τα αρχεία στο DBMSready folder.
4. Τρέχουμε το dbmsLoad.py για να φορτώσουμε τα δεδομένα στη βάση. Θα χρειαστεί να μεταφέρουμε τα αρχεία στο secure folder που διαβάζει η MySQL.
5. Ανοίγουμε terminal μέσα στο φάκελο του project τρέχουμε npm install και θα γίνει εγκατάσταση όλων των απαραίτητων αρχείων που χρειάζονται από το αρχείο package.json
6. Τέλος τρέχουμε το server με node index.js και ανοιγουμε browser σε localhost: 3000