



University of  
Macedonia  
MSc in  
Applied  
Informatics

ΠΑΝΕΠΙΣΤΗΜΙΟ ΜΑΚΕΔΟΝΙΑΣ  
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ  
ΤΜΗΜΑΤΟΣ ΕΦΑΡΜΟΣΜΕΝΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΒΙΒΛΙΟΘΗΚΕΣ ΟΠΤΙΚΟΠΟΙΗΣΗΣ  
ΔΕΔΟΜΕΝΩΝ. ΑΝΑΛΥΣΗ ΚΑΙ ΣΥΓΚΡΙΣΗ ΤΩΝ  
ΒΙΒΛΙΟΘΗΚΩΝ: D3, MARBOX GL ΚΑΙ PYGAL

Διπλωματική Εργασία

του

Σπύρου Σταμπολίδη

Θεσσαλονίκη, 2022

ΒΙΒΛΙΟΘΗΚΕΣ ΟΠΤΙΚΟΠΟΙΗΣΗΣ  
ΔΕΔΟΜΕΝΩΝ.ΑΝΑΛΥΣΗ ΚΑΙ ΣΥΓΚΡΙΣΗ ΤΩΝ  
ΒΙΒΛΙΟΘΗΚΩΝ:D3, MAPBOX GL ΚΑΙ PYGAL

Σπύρος Σταμπολίδης

Πτυχίο Μηχανικός Τοπογραφίας και Γεωπληροφορικής, ΤΕΙ Κεντρικής  
Μακεδονίας 2018

Διπλωματική Εργασία

υποβαλλόμενη για την μερική εκπλήρωση των απαιτήσεων του  
ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΤΙΤΛΟΥ ΣΠΟΥΔΩΝ ΣΤΗΝ ΕΦΑΡΜΟΣΜΕΝΗ  
ΠΛΗΡΟΦΟΡΙΚΗ

Επιβλέπων καθηγητής

Κασκάλης Θεόδωρος

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την

Κασκάλης Θεόδωρος      Μαργαρίτης Κωνσταντίνος      Γεωργιάδης Χρήστος

.....

.....

.....

Σταμπολίδης Σπύρος

## Περίληψη

Η οπτικοποίηση των δεδομένων (data visualization) αποτελεί τη διαδικασία αναπαράστασης αφαιρετικών ή επιστημονικών δεδομένων με τη μορφή εικόνας, η οποία μπορεί να συμβάλει στην καλύτερη και βαθύτερη κατανόηση της σημασίας των δεδομένων και των μεταβλητών ή των μονάδων που συνιστούν τα δεδομένα αυτά. Λόγω των τεράστιων και συνεχώς αυξανόμενων ποσοτήτων και πηγών πληροφορίας, η ανάγκη για οπτικοποίηση έχει γίνει μεγάλη. Η παρούσα διπλωματική εργασία θα ασχοληθεί με την καταγραφή των βιβλιοθηκών d3.js mapbox gl.js και pygal.org που σχετίζονται με την οπτικοποίηση των δεδομένων. Στην αρχή της διπλωματικής εργασίας θα γίνει μια μεγάλη αναζήτηση βιβλιογραφικών αναφορών μερικές εκ των οποίων παρατίθενται και παρακάτω. Γίνεται επίσης αναλυτική περιγραφή των βιβλιοθηκών d3.js , mapbox gl.js , pygal.org. ενώ παρατίθενται για κάθε μια ξεχωριστά διάφορα παραδείγματα και εφαρμογές για την χρήση τους. Στην συνέχεια ακολουθεί η σύγκριση μεταξύ τους ενώ παράλληλα αναλύονται τα θετικά και τα αρνητικά της κάθε βιβλιοθήκης. Τέλος παρατίθενται ιδέες συγκερασμού και απόπειρες βελτιστοποίησής τους σε επίπεδο εκτέλεσης.

**Λέξεις-Κλειδιά:** οπτικοποίηση των δεδομένων, JavaScript, Python, d3.js , mapbox gl.js , pygal.org, βιβλιοθήκη δεδομένων

## **Abstract**

Data visualization is the process of representing abstract or scientific data in the form of an image, which can contribute to a better and deeper understanding of the importance of data and the variables or units that make up that data. Due to the huge and ever-increasing amounts and sources of information, the need for visualization has become great. This dissertation will deal with the recording of d3.js, mapbox gl.js and pygal.org libraries related to data visualization. At the beginning of the dissertation there will be a great search for bibliographic references, some of which are listed below. A detailed description of the d3.js, mapbox gl.js , pygal.org libraries is also given, and various examples and applications for their use are listed separately. Then follows the comparison between them while analyzing the positives and negatives of each library. Finally, ideas for concatenation and attempts to optimize them at the execution level are presented.

**Keywords:** data visualization, JavaScript, Python, d3.js, mapbox gl.js , pygal.org ,data library

## Περιεχόμενα

<b>Περίληψη</b> .....	<b>3</b>
<b>Abstract</b> .....	<b>4</b>
<b>Κατάλογος Εικόνων &amp; Πινάκων</b> .....	<b>7</b>
<b>Κεφαλαίο 1<sup>ο</sup>: Οπτικοποίηση των δεδομένων</b> .....	<b>9</b>
1.1 Περιγραφή του όρου.....	9
1.2 Η ιστορία της Οπτικοποίησης Δεδομένων.....	11
1.3 Τρέχουσες τάσεις στην Οπτικοποίηση δεδομένων .....	13
1.4 Εργαλεία οπτικοποίησης δεδομένων .....	17
<b>Κεφαλαίο 2: Εισαγωγή στην Python</b> .....	<b>18</b>
2.1 Ιστορική αναδρομή.....	18
2.2 Η δράση της Python.....	19
2.2 Πλεονεκτήματα και Μειονεκτήματα της Python .....	20
<b>Κεφάλαιο 3: Βιβλιοθήκη οπτικοποίησης Pygal</b> .....	<b>23</b>
3.1 Τι είναι η βιβλιοθήκη Pygal .....	23
3.2 Τομείς που χρησιμοποιείται η Pygal .....	23
3.3 Βασικά στοιχεία – χαρακτηριστικά της Pygal .....	24
3.4 Εγκατάσταση της Pygal.....	25
3.5 Εφαρμογές.....	29
<b>Κεφαλαίο 4: Εισαγωγή στην JavaScript</b> .....	<b>31</b>
4.1 Ιστορική αναδρομή.....	31
4.2 Η δράση της JavaScript .....	32
4.3 Πλεονεκτήματα και Μειονεκτήματα της JavaScript .....	33
<b>Κεφαλαίο 5: Βιβλιοθήκες οπτικοποίησης D3.js</b> .....	<b>35</b>
5.1 Τι είναι η βιβλιοθήκη D3.js.....	35

5.2 Τομείς που χρησιμοποιείται η d3.js.....	35
5.3 Χαρακτηριστικά της d3.js.....	40
5.4 Εγκατάσταση της d3.js .....	41
5.5 Εφαρμογές της d3.js .....	44
<b>Κεφάλαιο 6: Βιβλιοθήκη οπτικοποίησης Mapbox.....</b>	<b>47</b>
6.1 Τι είναι η βιβλιοθήκη Mapbox .....	47
6.2 Τομείς που χρησιμοποιείται η Mapbox .....	48
6.3 Βασικά στοιχεία και χαρακτηριστικά της βιβλιοθήκης Mapbox .....	50
6.4 Εγκατάσταση της Mapbox.....	52
6.5 Εφαρμογές της Mapbox.....	57
<b>Κεφάλαιο 7: Σύγκριση βιβλιοθηκών οπτικοποίησης.....</b>	<b>61</b>
7.1 Ποσότητα κώδικα.....	61
7.2 Ευκολία εγκατάστασης.....	63
7.3 Styling.....	64
7.4 Basemap.....	68
7.5 Projections – Οπτικοποίηση σε μορφή χάρτη.....	68
7.6 Centering the map.....	69
7.7 Σύνοψη σύγκρισης.....	70
<b>Κεφάλαιο 8: Συμπεράσματα .....</b>	<b>71</b>
<b>Βιβλιογραφικές αναφορές .....</b>	<b>74</b>
Βιβλία .....	76

## Κατάλογος Εικόνων & Πινάκων

<b>Πίνακας 1.</b> Σύγκριση βιβλιοθηκών οπτικοποίησης δεδομένων.....	70
<b>Εικόνα 1.</b> Ιστορικό Χρονοδιάγραμμα Οπτικοποίησης δεδομένων. Πηγή: Zhang, 2006.....	11
<b>Εικόνα 2.</b> Η σελίδα τεκμηρίωσης της Pygal. Πηγή: Adams, 2014.	25
<b>Εικόνα 3.</b> Οθόνη εγκατεστημένης βιβλιοθήκης Pygal. Πηγή: Adams, 2014.....	27
<b>Εικόνα 4.</b> Έναρξη εγκατάστασης lxml με τη χρήση του Python Tools. Πηγή: Adams, 2014. ....	27
<b>Εικόνα 5.</b> Παράθυρο Solution Explorer με εγκατεστημένη τη lxml. Πηγή: Adams, 2014. ....	28
<b>Εικόνα 6.</b> Εγκατάσταση βιβλιοθήκης Pygal. Πηγή: Adams, 2014. ....	28
<b>Εικόνα 7.</b> Βασικό διάγραμμα γραμμής της Pygal. Πηγή: Adams, 2014 .....	30
<b>Εικόνα 8.</b> Απλό γράφημα που υλοποιήθηκε με τη βιβλιοθήκη D3. Πηγή: Meeks, 2017. ....	36
<b>Εικόνα 9.</b> Διαδικτυακός χάρτης ο οποίος παρουσιάζει την εθνολογική σύνθεση των σημαντικότερων πόλεων των ΗΠΑ και υλοποιήθηκε με τη βιβλιοθήκη D3. Πηγή: Meeks, 2017. ....	36
<b>Εικόνα 10.</b> Διαδραστικός χάρτης υποθαλάσσιων καλωδίων τηλεπικοινωνιών που υλοποιήθηκε με τη D3. Πηγή: Meeks, 2017.....	37
<b>Εικόνα 11.</b> Δίκτυο συνεργασιών των αρχαιολόγων που εργάζονται στην ίδια ανασκαφή για περίπου 25 χρόνια το οποίο υλοποιήθηκε με τη D3. Πηγή: Meeks, 2017. ....	37
<b>Εικόνα 12.</b> Δέντρο αναπαράστασης δεδομένων που υλοποιήθηκε με τη D3. Πηγή: Meeks, 2017. ....	38
<b>Εικόνα 13.</b> Αναπαράσταση ηχητικών κυμάτων που υλοποιήθηκε με τη D3. Πηγή: Meeks, 2017. ....	38
<b>Εικόνα 14.</b> Εξατομικευμένο γράφημα μπάρας που υλοποιήθηκε με τη D3. Πηγή: Meeks, 2017. ....	39

<b>Εικόνα 15.</b> Αξιοποίηση των δυνατοτήτων της D3 για την ενημέρωση του περιεχομένου της οπτικοποίησης δεδομένων σύμφωνα με την αλληλεπίδραση με τον χρήστη. Πηγή: Meeks, 2017.....	46
<b>Εικόνα 16.</b> Mapbox JSON. Πηγή: Eriksson & Rydkvist, 2015.....	50



# Κεφαλαίο 1<sup>ο</sup>: Οπτικοποίηση των δεδομένων

## 1.1 Περιγραφή του όρου

Οπτικοποίηση δεδομένων είναι η πρακτική της μετάφρασης πληροφοριών σε ένα οπτικό πλαίσιο, όπως ένας χάρτης ή ένα γράφημα, για να διευκολύνει τον ανθρώπινο εγκέφαλο να κατανοήσει και να αντλήσει πληροφορίες από τα δεδομένα. Ο κύριος στόχος της οπτικοποίησης δεδομένων είναι να διευκολύνει τον εντοπισμό προτύπων, τάσεων και ακραίων στοιχείων σε μεγάλα σύνολα δεδομένων. Ο όρος χρησιμοποιείται συχνά εναλλακτικά με άλλους, συμπεριλαμβανομένων των γραφικών πληροφοριών, της οπτικοποίησης πληροφοριών και των στατιστικών γραφικών (Brush, 2017).

Η οπτικοποίηση δεδομένων είναι ένα από τα βήματα της διαδικασίας της επιστήμης δεδομένων, η οποία δηλώνει ότι μετά τη συλλογή, επεξεργασία και μοντελοποίηση των δεδομένων, πρέπει να οπτικοποιηθούν για να βγουν συμπεράσματα. Η οπτικοποίηση δεδομένων είναι επίσης ένα στοιχείο της ευρύτερης αρχιτεκτονικής παρουσίασης δεδομένων (DPA), η οποία στοχεύει στον εντοπισμό, τον χειρισμό, τη μορφοποίηση και την παράδοση δεδομένων με τον πιο αποτελεσματικό δυνατό τρόπο (Heer et al., 2013).

Η οπτικοποίηση δεδομένων είναι σημαντική για σχεδόν κάθε καριέρα. Μπορεί να χρησιμοποιηθεί από καθηγητές για την εμφάνιση των αποτελεσμάτων των μαθητών, από επιστήμονες υπολογιστών που εξερευνούν τις εξελίξεις στην τεχνητή νοημοσύνη (AI) ή από στελέχη που θέλουν να μοιραστούν πληροφορίες με τους ενδιαφερόμενους. Διαδραματίζει επιπλέον, σημαντικό ρόλο σε έργα μεγάλων δεδομένων. Καθώς οι επιχειρήσεις συσσωρεύουν τεράστιες συλλογές δεδομένων κατά τα πρώτα χρόνια της τάσης των μεγάλων δεδομένων, χρειάζονταν έναν τρόπο για να αποκτήσουν γρήγορα και εύκολα μια επισκόπηση των δεδομένων τους. Τα εργαλεία οπτικοποίησης ήταν μια φυσική εφαρμογή (Hyman, 2009).

Η οπτικοποίηση είναι κεντρική στα προηγμένα αναλυτικά στοιχεία για παρόμοιους λόγους. Όταν ένας επιστήμονας δεδομένων γράφει προηγμένους αλγόριθμους προγνωστικής ανάλυσης ή μηχανικής μάθησης (ML), είναι σημαντικό να απεικονίζει τα αποτελέσματα για να παρακολουθεί τα αποτελέσματα και να

διασφαλίζει ότι τα μοντέλα αποδίδουν όπως προβλέπεται. Αυτό συμβαίνει επειδή οι οπτικοποιήσεις πολύπλοκων αλγορίθμων είναι γενικά πιο εύκολο να ερμηνευτούν από τις αριθμητικές εξόδους.

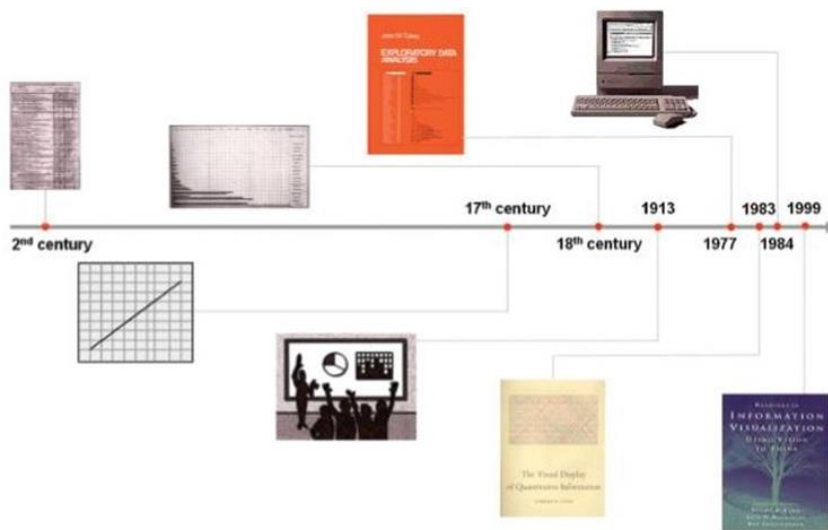
Η οπτικοποίηση δεδομένων παρέχει έναν γρήγορο και αποτελεσματικό τρόπο επικοινωνίας πληροφοριών με καθολικό τρόπο χρησιμοποιώντας οπτικές πληροφορίες. Η πρακτική μπορεί επίσης να βοηθήσει τις επιχειρήσεις να προσδιορίσουν ποιοι παράγοντες επηρεάζουν τη συμπεριφορά των πελατών (Zhang, 2006).

Άλλα οφέλη της οπτικοποίησης δεδομένων περιλαμβάνουν τα ακόλουθα (Antoniou et al., 2012):

- Την ικανότητα γρήγορης απορρόφησης πληροφοριών, βελτίωσης των γνώσεων και λήψης ταχύτερων αποφάσεων.
- Αυξημένη κατανόηση των επόμενων βημάτων που πρέπει να γίνουν για τη βελτίωση της οργάνωσης.
- Βελτιωμένη ικανότητα διατήρησης του ενδιαφέροντος του κοινού με πληροφορίες που μπορεί να κατανοήσει.
- Μια εύκολη διανομή πληροφοριών που αυξάνει την ευκαιρία ανταλλαγής πληροφοριών με όλους τους εμπλεκόμενους.
- Να εξαλείψει την ανάγκη για επιστήμονες δεδομένων, καθώς τα δεδομένα είναι πιο προσιτά και κατανοητά.
- Αυξημένη ικανότητα να ενεργεί γρήγορα στα ευρήματα και, ως εκ τούτου, να επιτυγχάνει με μεγαλύτερη ταχύτητα και λιγότερα λάθη.

## 1.2 Η ιστορία της Οπτικοποίησης Δεδομένων

Για να κατανοήσουμε τις τρέχουσες και τις μελλοντικές τάσεις του τομέα της οπτικοποίησης δεδομένων, θα πρέπει να εξετάσουμε ορισμένα ιστορικά στοιχεία. Παρά το γεγονός ότι οι προκάτοχοι της οπτικοποίησης δεδομένων χρονολογούνται από το 2<sup>ο</sup> αιώνα μ.Χ., οι περισσότερες βελτιώσεις έχουν πραγματοποιηθεί τους τελευταίους δύο περίπου αιώνες και κυρίως κατά τη διάρκεια των τελευταίων 30 χρόνων (Εικόνα 1) (Zhang, 2006).



*Εικόνα 1. Ιστορικό Χρονοδιάγραμμα Οπτικοποίησης δεδομένων. Πηγή: Zhang, 2006*

Ο παλαιότερος πίνακας που έχει διατηρηθεί δημιουργήθηκε το 2ο αιώνα στην Αίγυπτο και είχε ως σκοπό την οργάνωση αστρονομικών πληροφοριών. Ένας πίνακας αποτελεί μία αναπαράσταση δεδομένων με τη μορφή κειμένου, αλλά χρησιμοποιεί τα εικονικά χαρακτηριστικά της ευθυγράμμισης, του κενού και ορισμένες φορές οριζόντιες ή κάθετες γραμμές για το διαχωρισμό των δεδομένων σε γραμμές και στήλες. Οι πίνακες, όπως επίσης οι γράφηματα και τα διαγράμματα, ανήκουν στις ομάδες αναπαράστασης δεδομένων που καλούνται γραφικές παραστάσεις. Παρά το γεγονός ότι οι πίνακες περιλαμβάνουν κυρίως κείμενο, το εικονικό τους τμήμα, δηλαδή ο διαχωρισμός σε γραμμές και στήλες, αποτέλεσε ένα σημαντικό πρώτο βήμα για τις βελτιώσεις που ακολούθησαν (Heer et al., 2013).

Η εικονική αναπαράσταση ποσοτικών δεδομένων σε σχέση με τις κλίμακες δύο διαστάσεων, δηλαδή η πιο κοινή μορφή μίας γραφικής παράστασης, χρησιμοποιήθηκε αρκετά αργότερα, κατά τη διάρκεια του 17ου αιώνα. Ο Rene Descartes, ο Γάλλος φιλόσοφος και μαθηματικός, γνωστός για τη φράση “Cogito ergo sum” (“Σκέφτομαι άρα υπάρχω”), επινόησε αυτή τη μέθοδο αναπαράστασης ποσοτικών δεδομένων όχι για την παρουσίαση πληροφορίας αλλά για την πραγματοποίηση ορισμένων μαθηματικών πράξεων, βασισμένων σε ένα σύστημα συντεταγμένων. Αργότερα, όμως, η αναπαράσταση αυτή αναγνωρίστηκε ως ένα αποδοτικό μέσο παρουσίασης δεδομένων.

Ακολουθώντας την καινοτομία του Descartes, στα τέλη του 18ου αιώνα και στις αρχές του 19ου, ο Σκωτσέζος κοινωνικός επιστήμονας William Playfair επινόησε ή βελτίωσε εξαιρετικά πολλές γραφικές παραστάσεις που χρησιμοποιούνται στη σημερινή εποχή, όπως τα διαγράμματα ράβδων και οι πίτες.

Παρ’ όλα αυτά, πέρασε τουλάχιστον ένας αιώνας μέχρι να αναγνωριστεί η αξία αυτών των τεχνικών. Έτσι, το 1913, εισήχθησαν για πρώτη φορά ακαδημαϊκά μαθήματα οπτικοποίησης δεδομένων στο Κρατικό Πανεπιστήμιο της Iowa.

Ο John Tukey, καθηγητής Στατιστικής στο Princeton, ανέδειξε, το 1977, τη δύναμη της οπτικοποίησης δεδομένων σαν ένα μέσο εξερεύνησης και κατανόησης των δεδομένων, παρουσιάζοντας μία εικονική προσέγγιση εξερεύνησης και ανάλυσης δεδομένων, η οποία καλείται Εξερευνητική Ανάλυση Δεδομένων (Tukey, 1977).

Το 1983, ο λάτρης της οπτικοποίησης δεδομένων, Edward Tufte δημοσίευσε το πρωτοποριακό του βιβλίο με τίτλο «*Η Εικονική Παρουσίαση της Ποσοτικής Πληροφορίας*», στο οποίο περιέγραφε αποδοτικούς τρόπους οπτικής παρουσίασης δεδομένων και τους συνέκρινε με τους – μη αποδοτικούς – τρόπους που χρησιμοποιούνταν στην καθημερινή ζωή. Έναν χρόνο αργότερα, το 1984, η Apple Computer παρουσίασε τον πρώτο δημοφιλή και προσιτό υπολογιστή, ο οποίος επικεντρωνόταν στα γραφικά, από πλευράς αλληλεπίδρασης και παρουσίασης.

### 1.3 Τρέχουσες τάσεις στην Οπτικοποίηση δεδομένων

Η οπτικοποίηση δεδομένων είναι ένας εξαιρετικά ισχυρός τρόπος για τη δημιουργία επιχειρηματικών πληροφοριών και τη λήψη καλύτερων αποφάσεων που βασίζονται σε δεδομένα. Ο τρόπος με τον οποίο δημιουργούνται οι οπτικοποιήσεις έχει εξελιχθεί με την πάροδο του χρόνου από τα απλά και στατικά γραφήματα του παρελθόντος σε διαδραστικά, ελκυστικά γραφικά σήμερα. Παρακάτω αναλύονται οι επτά κορυφαίες τάσεις στην οπτικοποίηση δεδομένων που θα γίνουν πιο διαδεδομένες στο μέλλον (Meridou et al., 2014).

- **Διατήρηση του χρήστη στο κέντρο της σχεδίασης της οπτικοποίησης δεδομένων.** Η παραδοσιακή διεπαφή χρήστη και οι επιλογές οπτικοποίησης δεδομένων που διατίθενται από πλατφόρμες αναλυτικών στοιχείων, όπως αναφορές, γραφήματα και πίνακες εργαλείων λειτουργούν καλά για ορισμένες εφαρμογές, αλλά περιμένουν από τον χρήστη να προσκολληθεί στην πλατφόρμα, αντί να κάνουν την πλατφόρμα να λειτουργεί για τον χρήστη. Στις παραδοσιακές διεπαφές χρήστη, η προσέγγιση ήταν συχνά "περισσότερα δεδομένα ίσον μεγαλύτερη αξία", που σημαίνει ότι εάν δεν είναι σίγουρο ποια δεδομένα θα χρειαστούν οι χρήστες για να επιτύχουν τους στόχους τους, τότε υπάρχει κίνδυνος να τους δοθούν πάρα πολλές πληροφορίες για την κάλυψη όλων των σεναρίων. Αυτό έχει ως αποτέλεσμα πάρα πολλούς στατικούς πίνακες εργαλείων, φίλτρα, γραφήματα και αριθμούς που δεν προσθέτουν αξία και μπορούν να μπερδέψουν τους χρήστες. Αντίθετα, αυτό που χρειάζεται είναι οπτικοποιήσεις που είναι εξατομικευμένες στις συγκεκριμένες ανάγκες κάθε χρήστη.
- **Η οπτικοποίηση των δεδομένων γίνεται πιο κοινωνική.** Η οπτικοποίηση δεδομένων γίνεται ολοένα και πιο κοινωνική και μοιράζεται συχνά σε διάφορες πλατφόρμες μέσω κοινωνικής δικτύωσης αυτές τις μέρες, όπου προσελκύει πολλούς οπαδούς και θεατές. Οι επιστήμονες δεδομένων έχουν επίσης αναγνωρίσει αυτήν την τάση και πλέον επικεντρώνονται περισσότερο στο να κάνουν τις

οπτικοποιήσεις φιλικές στα μέσα κοινωνικής δικτύωσης, σχεδιασμένες να προσελκύουν καλύτερα τους οπαδούς στα μέσα κοινωνικής δικτύωσης. Καθώς το εύρος προσοχής των χρηστών των μέσων κοινωνικής δικτύωσης είναι αρκετά σύντομο, τα δεδομένα πρέπει να είναι εμφανή, οπτικά ελκυστικά, συνοπτικά και εύστοχα. Η οπτικοποίηση δεδομένων για την κοινή χρήση μέσων κοινωνικής δικτύωσης ακολουθεί τη φιλοσοφία «λιγότερο είναι περισσότερο». Μερικά παραδείγματα οπτικοποιήσεων μέσων κοινωνικής δικτύωσης περιλαμβάνουν τρισδιάστατα κινούμενα σχέδια, GIF και οπτικοποιήσεις περιεχομένου και δεδομένων με τη μορφή βίντεο για δημοφιλείς εφαρμογές κοινής χρήσης βίντεο.

- **Ο εκδημοκρατισμός της ανάλυσης δεδομένων.** Μια άλλη ενδιαφέρουσα τάση είναι ο εκδημοκρατισμός δεδομένων που σημαίνει ότι οποιοσδήποτε μπορεί να χρησιμοποιήσει δεδομένα ανά πάσα στιγμή για να λάβει αποφάσεις, χωρίς όρια πρόσβασης. Τα εργαλεία οπτικοποίησης δεδομένων διαδραματίζουν σημαντικό ρόλο στον εκδημοκρατισμό των δεδομένων και των αναλυτικών στοιχείων και καθιστούν τις πληροφορίες που βασίζονται σε δεδομένα προσβάσιμες σε όλους τους χρήστες σε έναν οργανισμό. Δίνει τη δυνατότητα στους χρήστες να έχουν πρόσβαση και να οπτικοποιούν δεδομένα και να δημιουργούν διαδραστικούς εξατομικευμένους πίνακες εργαλείων με μερικά κλικ. Αυτό βοηθά στον εντοπισμό νέων επιχειρηματικών ευκαιριών και λύσεων σε προκλήσεις και βασίζεται σε ιδέες και καινοτομίες σε ολόκληρο τον οργανισμό.
- **Αναζήτηση ιστοριών που αποκαλύπτουν τα δεδομένα.** Όταν οι άνθρωποι ακούν ιστορίες, συχνά αισθάνονται συναισθηματικά δεσμευμένοι και τείνουν να τις θυμούνται καλύτερα από ό,τι μπορούν να ανακαλέσουν ξεκάθαρα γεγονότα. Τα δεδομένα και οι αριθμοί έχουν σημαντικές ιστορίες να πουν, αλλά βασίζονται σε εσάς για να τους δώσετε μια καθαρή και πειστική φωνή. Η ιδέα της οπτικοποίησης δεδομένων για την αφήγηση και για να νοηματοδοτήσει τους αριθμούς και τα δεδομένα γίνεται ολοένα και πιο δημοφιλής. Ωστόσο, δεν αρκεί μόνο η οπτικοποίηση των δεδομένων. Οι οπτικοποιητές δεδομένων

πρέπει να προχωρήσουν προς το να γίνουν αφηγητές. ανακαλύπτοντας πρώτα το νόημα των ίδιων των δεδομένων και, στη συνέχεια, δημιουργώντας μια αφήγηση που βοηθά το κοινό να αποκαλύψει το νόημά τους, ενώ το κρατά αφοσιωμένο στα δεδομένα. Για να επιτευχθεί αυτό, ένας αριθμός οπτικοποιήσεων συγκεντρώνεται σε μια σειρά που οδηγεί το κοινό στα δεδομένα και το βοηθά να καταλήξει σε ουσιαστικές ερμηνείες.

- **Η οπτικοποίηση δεδομένων δεν περιορίζεται πλέον σε επιστήμονες και αναλυτές δεδομένων.** Η IBM, μέσω μιας πρόσφατης μελέτης, προβλέπει αύξηση σχεδόν 40% στη ζήτηση για επιστήμονες δεδομένων και μηχανικούς δεδομένων τα επόμενα δύο χρόνια. Ταυτόχρονα, οι εργοδότες σήμερα αρχίζουν να περιμένουν ένα ορισμένο επίπεδο εξοικείωσης και άνεσης με το χειρισμό και την οπτικοποίηση δεδομένων σε όλες τις εταιρείες τους και όχι μόνο από τους επιστήμονες και τους μηχανικούς τους. Ως αποτέλεσμα αυτής της τάσης, αναμένεται η συνεχής ανάπτυξη εργαλείων και πόρων που στοχεύουν στο να καταστήσουν το πεδίο οπτικοποίησης δεδομένων και τα οφέλη του πιο προσιτά σε όλους και όχι μόνο στους επιστήμονες δεδομένων.
- **Η Τεχνητή Νοημοσύνη και η Μηχανική Μάθηση θα κάνουν τη δημιουργία οπτικοποιήσεων δεδομένων πιο έξυπνη, όχι δυσκολότερη.** Η μηχανική μάθηση (ML) και η τεχνητή νοημοσύνη (AI) είναι τομείς ταχείας τεχνολογικής προόδου, που εφαρμόζονται συνεχώς με νέους τρόπους για επιχειρηματικούς στόχους. Αυτές οι αναδυόμενες τεχνολογίες διαδραματίζουν επίσης ολοένα και πιο σημαντικό ρόλο στην οπτικοποίηση δεδομένων. Υπάρχει πληθώρα δεδομένων που λαμβάνονται από μεγάλο αριθμό διαφορετικών πηγών, συμπεριλαμβανομένων εσωτερικών συστημάτων όπως ο προγραμματισμός πόρων της επιχείρησης (ERP) και οι εφαρμογές διαχείρισης σχέσεων με τους πελάτες (CRM), ή εξωτερικές πηγές όπως ο ιστός και τα δεδομένα ροής. Τέτοια δεδομένα διατίθενται σε μορφές με μεγάλες παραλλαγές. Ωστόσο, πρέπει να ενσωματωθούν, να μορφοποιηθούν και να προσαρμοστούν ώστε να είναι σχετικά για έναν συγκεκριμένο χρήστη, πράγμα που απαιτεί την εύρεση κρίσιμων

πληροφοριών που βοηθούν στην απόδοση καλύτερων αποτελεσμάτων. Η μηχανική εκμάθηση και η επεξεργασία φυσικής γλώσσας (NLP) έχουν ισχυρές δυνατότητες να ερμηνεύουν μη δομημένα δεδομένα, συμπεριλαμβανομένων κειμένου, ήχου και βίντεο, και να ανακαλύπτουν σημαντικές πληροφορίες. Αυτό είναι το πρώτο βήμα προς την ανάπτυξη αποτελεσματικών οπτικοποιήσεων δεδομένων. Η ML και η τεχνητή νοημοσύνη μπορούν να βοηθήσουν στη μείωση του φόρτου εργασίας της οπτικοποίησης μέσω της αυτοματοποίησης.

- **Πρώτες οπτικοποιήσεις δεδομένων φιλικές προς κινητά.** Οι φορητές συσκευές έχουν γίνει απαραίτητα εργαλεία εργασίας τόσο για επαγγελματικούς όσο και για προσωπικούς σκοπούς. Καθώς τα χρησιμοποιούμε για πρόσβαση σε εφαρμογές, ιστότοπους, αναφορές και πολλά άλλα, η εμφάνιση για κινητά είναι συχνά η πρώτη εντύπωση. Η ποιότητα της εμπειρίας χρήστη κατά την πρόσβαση σε οπτικοποιήσεις δεδομένων σε κινητές συσκευές είναι ζωτικής σημασίας και θα είναι ακόμη μεγαλύτερη στο μέλλον. Οι οπτικοποιήσεις δεδομένων που έχουν αναπτυχθεί για εταιρικές εφαρμογές ή ιστότοπους πρέπει να δοκιμάζονται σε διάφορες κινητές συσκευές και διαφορετικά προγράμματα περιήγησης. Είναι πολύ σημαντικό οι απεικονίσεις να είναι σαφείς, απλές, συμπαγείς και συνοπτικές.



## 1.4 Εργαλεία οπτικοποίησης δεδομένων

Τα εργαλεία οπτικοποίησης δεδομένων μπορούν να χρησιμοποιηθούν με διάφορους τρόπους. Η πιο κοινή χρήση σήμερα είναι ως εργαλείο αναφοράς επιχειρηματικής ευφυΐας (BI). Οι χρήστες μπορούν να ρυθμίσουν εργαλεία οπτικοποίησης για τη δημιουργία αυτόματων πινάκων εργαλείων που παρακολουθούν την απόδοση της εταιρείας σε βασικούς δείκτες απόδοσης ( KPI ) και ερμηνεύουν οπτικά τα αποτελέσματα (Brush, 2017).

Οι παραγόμενες εικόνες μπορεί επίσης να περιλαμβάνουν διαδραστικές δυνατότητες, που επιτρέπουν στους χρήστες να τις χειριστούν ή να εξετάσουν πιο προσεκτικά τα δεδομένα για ερωτήσεις και ανάλυση. Μπορούν επίσης να ενσωματωθούν δείκτες που έχουν σχεδιαστεί για να ειδοποιούν τους χρήστες όταν τα δεδομένα έχουν ενημερωθεί ή όταν εμφανίζονται προκαθορισμένες συνθήκες.

Πολλά επιχειρηματικά τμήματα εφαρμόζουν λογισμικό οπτικοποίησης δεδομένων για να παρακολουθούν τις δικές τους πρωτοβουλίες. Για παράδειγμα, μια ομάδα μάρκετινγκ μπορεί να εφαρμόσει το λογισμικό για την παρακολούθηση της απόδοσης μιας καμπάνιας ηλεκτρονικού ταχυδρομείου, παρακολουθώντας μετρήσεις όπως το ανοιχτό ποσοστό, την αναλογία κλικ προς αριθμό εμφανίσεων και το ποσοστό μετατροπών (Brush, 2017).

Τα μεγαλύτερα ονόματα στην αγορά εργαλείων μεγάλων δεδομένων περιλαμβάνουν Microsoft, IBM, SAP και SAS. Ορισμένοι άλλοι προμηθευτές προσφέρουν εξειδικευμένο λογισμικό απεικόνισης μεγάλων δεδομένων. Δημοφιλή ονόματα σε αυτήν την αγορά περιλαμβάνουν τα Tableau, Qlik και Tibco.

## Κεφαλαίο 2: Εισαγωγή στην Python

### 2.1 Ιστορική αναδρομή

Η Python δημιουργήθηκε το 1989 από τον Ολλανδό Guido van Rossum στο ερευνητικό κέντρο Centrum Wiskunde & Informatica (CWI), του οποίου βασική έρευνα θεωρείται το πεδίο των Μαθηματικών και της Επιστήμης Υπολογιστών. Βασική πηγή έμπνευσης της δημιουργίας της Python είναι η γλώσσα προγραμματισμού ABC. Στην αρχή την χρησιμοποίησαν ως γλώσσα σεναρίων για το καταναμημένο λειτουργικό σύστημα Amoeba. Όπως αναφέρει ο ίδιος ο Guido van Rossum, η πιο καινοτόμα συνεισφορά του στην επιτυχία της Python ήταν το ότι αυτή είχε τη δυνατότητα να επεκτείνεται εύκολα. Το μοντέλο της Python σχεδιάστηκε με τέτοιο τρόπο ώστε να είναι ευέλικτα επεκτάσιμο, να παρέχει ενσωματωμένα στοιχεία (εντολές, τύπους αντικειμένων, κ.λπ.) αλλά και να δίνει τη δυνατότητα στους προγραμματιστές να προσθέτουν τα δικά τους στοιχεία ανάλογα με τις ανάγκες τους και το σύστημα που χρησιμοποιούν. Η πρώτη έκδοση κυκλοφόρησε το 1991. Η ονομασία της προέρχεται από μια κωμική σειρά της δεκαετίας του 70 το «Monty Python's Flying Circus» του BBC της Μεγάλης Βρετανίας. Ο Guido van Rossum το όνομα που έψαχνε για τη νέα γλώσσα προγραμματισμού ήθελε να είναι μικρό, μοναδικό και ελαφρώς μυστήριο. Από αυτό συμπεραίνουμε ότι δεν υπάρχει καμία απολύτως συσχέτιση με ερπετά ή το γνωστό φίδι πύθωνας (Harrison, 2003).

Η Python έχει ραγδαία εξέλιξη με τις νέες εκδόσεις και με βασικό εργαλείο τα Python Enhancement Proposals ("PEPs"), τα οποία είναι απαραίτητα στο να δίνουν γενικές πληροφορίες, οδηγίες, προτάσεις και περιγραφές για τυχόν νέα χαρακτηριστικά της γλώσσας. Μέχρι στιγμής υπάρχουν δύο σειρές εκδόσεων της Python, η σειρά 2.χ με πιο πρόσφατη έκδοση παραγωγής την 2.7.9 και η νεότερη σειρά 3.χ με πιο πρόσφατη έκδοση παραγωγής την 3.4.3. Και στις δυο σειρές εκδόσεων, η Python είναι στο το μεγαλύτερο μέρος της η ίδια, αλλά με διαφορές σε πολλές λεπτομέρειες, κυρίως στο πως λειτουργούν τα ενσωματωμένα δεκαπέντε αντικείμενα όπως τα λεξικά και οι συμβολοσειρές (Harrison, 2003).

## 2.2 Η δράση της Python

«Η Python (εικόνα 2) είναι από τις πιο διαδεδομένες γλώσσες προγραμματισμού σήμερα. Η Python αποτελεί έναν συμβιβασμό μεταξύ της R, η οποία ασχολείται σε μεγάλο βαθμό με την ανάλυση και την απεικόνιση δεδομένων, και της Java, η οποία αποτελεί την βάση πολλών εφαρμογών μεγάλης κλίμακας. Μεταξύ άλλων, υποστηρίζει αντικειμενοστραφή προγραμματισμό, δομημένο προγραμματισμό και λειτουργικά πρότυπα προγραμματισμού. Η Python μπορεί να πραγματοποιήσει κάθε εργασία από να βγάλει δεδομένα έως το να κατασκευάσει ιστότοπους που να μπορούν να λειτουργούν ενσωματωμένα συστήματα, όπου όλα είναι σε μια ενοποιημένη γλώσσα. Για παράδειγμα, στο *Forecast Watch*, η Python βοήθησε έναν αναλυτή να μαζέψει τις προβλέψεις από άλλους ιστότοπους και να δημιουργήσει έναν μηχανισμό συσσωμάτωσης για την κατάρτιση των δεδομένων και έναν κώδικα ιστότοπου για την εμφάνιση των αποτελεσμάτων. Αρχικά χρησιμοποίησαν την PHP για να φτιάξει τον ιστότοπο μέχρι που η εταιρεία κατάλαβε ότι μπορούσε πιο εύκολα να ασχοληθεί μόνο με μια ενιαία γλώσσα» (Jackson, 2011, pp: 80-83).



*Εικόνα 2. Η σελίδα τεκμηρίωσης της Pygal.*

Το Facebook, σύμφωνα με άρθρο του περιοδικού *Fast Company* για το 2014, επέλεξε την Python για την ανάλυση δεδομένων, επειδή ήδη χρησιμοποιούνταν σε άλλα μέρη της εταιρείας. Λόγω της πολυχρησιμότητάς της, η Python είναι πολύ αγαπητή από τους προγραμματιστές.

Επιπρόσθετα, η Python θεωρείται σχετικά εύκολη γλώσσα προγραμματισμού και οι πολλές αποκλειστικές αναλυτικές βιβλιοθήκες που διαθέτει σήμερα έχει ως αποτέλεσμα, να μπορούν να βρύνουν οι επιστήμονες πακέτα δεδομένων σε κάθε τομέα σχεδόν, τα οποία είναι ήδη προσαρμοσμένα στις ανάγκες τους και διαθέσιμα για λήψη (Jackson, 2011, pp: 80-83).

## 2.2 Πλεονεκτήματα και Μειονεκτήματα της Python

Για την εκμάθηση μιας νέας γλώσσας προγραμματισμού είναι σημαντικό ο χρήστης να γνωρίζει τα οφέλη και τα μειονεκτήματα αυτής της γλώσσας. Αυτό θα τον βοηθήσει να αποκτήσει καλύτερη γνώση για το πώς μπορεί να εκμεταλλευτεί πλήρως τη γλώσσα προγραμματισμού.

Γνωρίζοντας τα δυνατά και τα αδύνατα σημεία της Python, ο χρήστης μπορεί να δημιουργήσει ισχυρές εφαρμογές. Παρακάτω παρατίθενται τα πλεονεκτήματα και τα μειονεκτήματα της Python (McKinney, 2008).

### Πλεονεκτήματα

- **Εύκολη στην ανάγνωση, την μάθηση και την γραφή.** Η Python είναι μια γλώσσα προγραμματισμού υψηλού επιπέδου που έχει σύνταξη όπως η αγγλική. Αυτό διευκολύνει την ανάγνωση και την κατανόηση του κώδικα. Η Python είναι πραγματικά εύκολη στην εκμάθηση και την εκμάθηση, γι' αυτό πολλοί άνθρωποι συνιστούν την Python σε αρχάριους. Χρειάζεται λιγότερες γραμμές κώδικα για να εκτελέσει την ίδια εργασία σε σύγκριση με άλλες κύριες γλώσσες όπως η C/C++ και η Java.
- **Βελτιωμένη παραγωγικότητα.** Η Python είναι μια πολύ παραγωγική γλώσσα. Λόγω της απλότητας της Python, οι προγραμματιστές μπορούν να επικεντρωθούν στην επίλυση του προβλήματος. Δεν χρειάζεται να αφιερώνουν πολύ χρόνο στην κατανόηση της σύνταξης ή της συμπεριφοράς της γλώσσας προγραμματισμού. Γράφουν λιγότερο κώδικα και κάνουν περισσότερα πράγματα.
- **Διερμηνευμένη γλώσσα.** Η Python είναι μια διερμηνευμένη γλώσσα (interpreted που σημαίνει ότι η Python εκτελεί απευθείας τον κώδικα γραμμή προς γραμμή. Σε περίπτωση οποιουδήποτε σφάλματος, σταματά την περαιτέρω εκτέλεση και αναφέρει το σφάλμα που έχει προκύψει. Η Python εμφανίζει μόνο ένα σφάλμα ακόμα κι αν το πρόγραμμα έχει πολλά σφάλματα. Αυτό διευκολύνει τον εντοπισμό σφαλμάτων.

- **Δυναμική γλώσσα.** Η Python δεν γνωρίζει τον τύπο της μεταβλητής μέχρι να εκτελέσουμε τον κώδικα. Εκχωρεί αυτόματα τον τύπο δεδομένων κατά την εκτέλεση . Ο προγραμματιστής δεν χρειάζεται να ανησυχεί για τη δήλωση μεταβλητών και των τύπων δεδομένων τους.
- **Ανοιχτού κώδικα.** Η Python καλύπτεται από την άδεια ανοιχτού κώδικα εγκεκριμένη από το OSI . Αυτό καθιστά δωρεάν τη χρήση και τη διανομή .Ο χρήστης μπορεί να κατεβάσει τον πηγαίο κώδικα, να τον τροποποιήσει και ακόμη και να διανεμίει την έκδοση της Python.
- **Υποστήριξη Vast Libraries.** Η τυπική βιβλιοθήκη της Python είναι τεράστια, δίνει τη δυνατότητα στο χρήστη να βρει σχεδόν όλες τις λειτουργίες που χρειάζονται για την εργασία του. Επομένως, δεν χρειάζεται να βασίζεται σε εξωτερικές βιβλιοθήκες. Αλλά ακόμα κι αν το κάνει, ένας διαχειριστής πακέτων Python (pip) διευκολύνει την εισαγωγή άλλων σπουδαίων πακέτων από το ευρετήριο πακέτων Python (PyPi) που αποτελείται από πάνω από 200.000 πακέτα.
- **Φορητότητα.** Σε πολλές γλώσσες όπως η C/C++, είναι απαραίτητη η αλλαγή του κωδικού σας για την εκτέλεση του πρόγραμματος σε διαφορετικές πλατφόρμες. Αυτό δεν είναι το ίδιο με την Python. Μπορεί κανείς να γράψει μόνο μία φορά και το τρέξει οπουδήποτε. Ωστόσο, θα πρέπει να προσέχει να μην συμπεριλάβει λειτουργίες που εξαρτώνται από το σύστημα.

### **Μειονεκτήματα της Python**

- **Αργή ταχύτητα.** Συζητήσαμε παραπάνω ότι η Python είναι μια ερμηνευμένη γλώσσα και μια δυναμικά πληκτρολογημένη γλώσσα. Η εκτέλεση του κώδικα γραμμή προς γραμμή συχνά οδηγεί σε αργή εκτέλεση. Η δυναμική φύση της Python είναι επίσης υπεύθυνη για την αργή ταχύτητα της Python επειδή πρέπει να κάνει την επιπλέον εργασία κατά την εκτέλεση του κώδικα. Έτσι, η Python δεν χρησιμοποιείται για σκοπούς όπου η ταχύτητα είναι μια σημαντική πτυχή του έργου.
- **Μη αποδοτική μνήμη.** Για να προσφέρει απλότητα στον προγραμματιστή, η Python πρέπει να κάνει μια μικρή ανταλλαγή. Η γλώσσα προγραμματισμού Python χρησιμοποιεί μεγάλη ποσότητα μνήμης . Αυτό μπορεί να είναι ένα

μειονέκτημα κατά τη δημιουργία εφαρμογών όταν προτιμάται η βελτιστοποίηση της μνήμης.

- **Αδυναμία στο Mobile Computing.** Η Python χρησιμοποιείται γενικά στον προγραμματισμό από την πλευρά του διακομιστή. Δεν μπορεί να δει κανείς την Python στις εφαρμογές πελάτη ή στις κινητές συσκευές. Αυτό συμβαίνει επειδή η Python δεν είναι αποδοτική στη μνήμη και έχει αργή ισχύ επεξεργασίας σε σύγκριση με άλλες γλώσσες.
- **Πρόσβαση στη βάση των δεδομένων.** Ο προγραμματισμός σε Python είναι εύκολος και χωρίς άγχος. Αλλά όταν αλληλοεπιδρά με τη βάση δεδομένων, υστερεί. Το επίπεδο πρόσβασης στη βάση δεδομένων της Python είναι πρωτόγονο και υπανάπτυκτο σε σύγκριση με τις δημοφιλείς τεχνολογίες όπως το JDBC και το ODBC. Οι τεράστιες επιχειρήσεις χρειάζονται ομαλή αλληλεπίδραση σύνθετων δεδομένων παλαιού τύπου και επομένως η Python σπάνια χρησιμοποιείται σε επιχειρήσεις.
- **Σφάλματα χρόνου εκτέλεσης.** Είναι γνωστό ότι η Python είναι μια γλώσσα, όπου ο τύπος δεδομένων μιας μεταβλητής μπορεί να αλλάξει ανά πάσα στιγμή. Μια μεταβλητή που περιέχει ακέραιο αριθμό μπορεί να περιέχει μια συμβολοσειρά στο μέλλον, η οποία μπορεί να οδηγήσει σε σφάλματα χρόνου εκτέλεσης. Ως εκ τούτου, οι προγραμματιστές Python πρέπει να εκτελέσουν ενδελεχή δοκιμή των εφαρμογών.

## Κεφάλαιο 3: Βιβλιοθήκη οπτικοποίησης Pygal

### 3.1 Τι είναι η βιβλιοθήκη Pygal

Η Pygal είναι μία βιβλιοθήκη οπτικοποίησης δεδομένων ανοιχτού κώδικα σε περιβάλλον Python. Είναι μία από τις βέλτιστες βιβλιοθήκες Python για τη δημιουργία εξαιρετικά διαδραστικών γραφημάτων και γραφημάτων για διαφορετικά σύνολα δεδομένων. Επίσης, καθιστά δυνατή τη μεταφόρτιση των απεικονίσεων σε μορφή SVG (Scalable Vector Graphics) ή PNG (Portable Graphics Format) για πολλαπλές εφαρμογές και την ανάλογη προσαρμογή τους (Walia, 2022).

Μπορεί να γίνει λήψη για τις εκδόσεις της Python 3.6, 3.7, 3.8, 3.9 και pygy και τα πακέτα διανομής Linux για τις ακόλουθες εκδόσεις του Linux:

- Fedora
- Gentoo
- Ubuntu
- Debian
- Arch Linux

Το κύριο πακέτο της βιβλιοθήκης περιέχει όλα τα διαθέσιμα γραφήματα, την κλάση διαμόρφωσης και τα πρόσθετα ονομασιών επεκτάσεων απεικονίσεων (Walia, 2022).

### 3.2 Τομείς που χρησιμοποιείται η Pygal

Η Pygal είναι μία βιβλιοθήκη οπτικοποίησης δεδομένων σε περιβάλλον Python που επιτρέπει την υλοποίηση κομψών και διαδραστικών γραφικών παραστάσεων με ελάχιστες προσπάθειες κωδικοποίησης. Εάν ο χρήστης επιθυμεί τη βέλτιστη ανάλυση για την εκτύπωση των γραφικών στον εξοπλισμό του, τότε η Pygal επιτρέπει τη μεταφόρτωση των απεικονίσεων σε μορφή SVG, PNG, Etree, Base 64 data URI, Browser και PyQuery. Επιπλέον, μπορεί να χρησιμοποιηθεί για να εμφανίσει τις παραστάσεις με τη χρήση των εφαρμογών Flask App ή Django ή

απευθείας στο HTML του χρήστη, παραπέμποντας τις ενσωματώσεις του στα έγγραφα των ιστοσελίδων (Walia, 2022).

Η Pygal αποτελεί μία κατάλληλη επιλογή κατά την εργασία με σύνθετα σύνολα δεδομένων, καθώς είναι εξαιρετικά προσαρμόσιμη και παρέχει διαφορετικά γραφήματα (γραμμή, μπάρα, Ιστόγραμμα, πίτα, πλαίσιο, τελεία, κλπ.), στυλ (ενσωματωμένο στυλ, παραμετρικά στυλ, κλπ.), διαμορφώσεις γραφημάτων, διαμορφώσεις σειρών, διαμορφώσεις τιμών (ετικέτες, στυλ, χαρακτηριστικά κόμβου, σύνδεσμοι), sparklines (σπινθηρωτό κείμενο, κλπ.), πίνακες (στυλ, σύνολο, ανάστροφο) και μορφές εξόδου (Walia, 2022).

### **3.3 Βασικά στοιχεία – χαρακτηριστικά της Pygal**

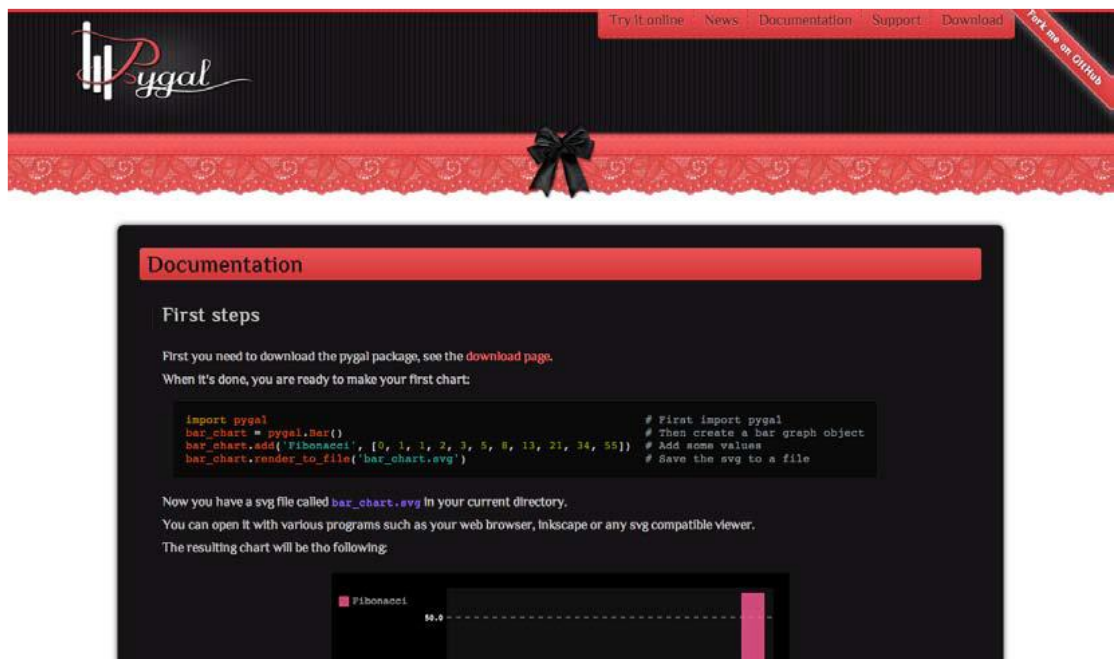
Η Pygal διατίθεται στον ιστότοπο <http://pygal.org/>. Είναι ένα εργαλείο δημιουργίας γραφημάτων SVG που βασίζεται στη γλώσσα προγραμματισμού Python. Αναπτύχθηκε από την Κοινότητα Kozea (<http://community.kozea.fr/>), όπως φαίνεται στο παρακάτω στιγμιότυπο οθόνης. Η κοινότητα αυτή είναι μία ομάδα η οποία δραστηριοποιείται στην οικοδόμηση ποιοτικών βιβλιοθηκών ανοιχτού κώδικα (κυρίως βασισμένες στην Python, αλλά και για έργα HTML5) (Adams, 2014).

Η βιβλιοθήκη Pygal προσφέρει πολλαπλές επιλογές χαρτογράφησης πέρα από εκείνες που θεωρούνται τυπικά διαγράμματα: γραφήματα ράβδων, γραμμικά γραφήματα και γραφήματα πίτας. Περιλαμβάνει έναν παγκόσμιο χάρτη, γραφήματα διοχέτευσης, διαγράμματα ραντάρ και γραφικά κουτιών, μεταξύ άλλων (Adams, 2014).

Επίσης, περιλαμβάνει προκατασκευασμένα θέματα και στυλ, τα οποία δεν χρειάζεται να προσαρμοστούν εάν κάτι τέτοιο δεν είναι επιθυμητό από τον προγραμματιστή. Ακόμη, δεδομένου ότι η έξοδος της βιβλιοθήκης γραφημάτων είναι σε μορφή SVG, αυτό την καθιστά έναν εξαιρετικά ευέλικτο τύπο εξόδου για HTML5 ή ακόμα και για μέσα εκτύπωσης. Ένα ζήτημα με ορισμένες βιβλιοθήκες δημιουργίας γραφημάτων στην Python είναι ότι η έξοδος είναι προεπιλεγμένη στη μορφή PNG με καθορισμένο μέγεθος εικόνας. Καθώς, το SVG είναι ένα διανυσματικό γραφικό (ένας τύπος γραφικού που είναι επεκτάσιμος χωρίς απώλεια ποιότητας εικόνας), μπορεί να κλιμακωθεί και να αλλάξει το μέγεθός της για οποιαδήποτε εφαρμογή χωρίς απώλεια της ποιότητας (Adams, 2014).



Ο ιστότοπος της Pygal περιλαμβάνει επιπλέον, μία αρκετά καλή και ευανάγνωστη τεκμηρίωση. Ένα στοιχείο που είναι αρκετά κοινό με τις βιβλιοθήκες Python που αναπτύσσονται από τρίτους είναι ότι η τεκμηρίωση μπορεί να κυμαίνεται από ένα καλά τεκμηριωμένο wiki με δυνατότητα αναζήτησης στο διαδίκτυο έως ένα απλό αρχείο readme.txt που δείχνει μόνο τον τρόπο εγκατάστασης της βιβλιοθήκης. Η βιβλιοθήκη Pygal επίσης δεν απαιτεί πολλές εξαρτήσεις, κάτι που είναι κρίσιμο για ένα εισαγωγικό βιβλίο, καθώς μία πολύ εξαρτημένη βιβλιοθήκη μπορεί να προκαλέσει προβλήματα σε νέους προγραμματιστές (Adams, 2014).



Εικόνα 2. Η σελίδα τεκμηρίωσης της Pygal. Πηγή: Adams, 2014.

Πολλά frameworks έχουν ορισμένες πολύ επιλεκτικές εξαρτήσεις που μπορεί να χρειαστούν για τη διεκπεραίωση ορισμένων έργων, αλλά ενδέχεται να μη λειτουργήσουν με το σύστημα του προγραμματιστή (Adams, 2014).

Η βιβλιοθήκη lxml είναι η μόνη βιβλιοθήκη που απαιτείται για την Pygal, αλλά έχει ορισμένα προβλήματα ανάλογα με το λειτουργικό σύστημα στο οποίο εκτελείται ο κώδικας Python του εκάστοτε χρήστη (Adams, 2014).

### 3.4 Εγκατάσταση της Pygal

Αρχικά, για την περίπτωση κατά την οποία δεν έχει εγκατασταθεί το lxml, εάν ο προγραμματιστής εργάζεται σε περιβάλλον Windows, θα πρέπει να εγκαταστήσει το πρόγραμμα εγκατάστασης lxml. Διαφορετικά, οι ακόλουθες εντολές εγκαθιστούν το lxml. Στη συνέχεια, χρησιμοποιείται το pip και εγκαθίσταται το Pygal

χρησιμοποιώντας τις ακόλουθες εντολές για συστήματα Windows και Mac/Linux (πρέπει να σημειωθεί ότι το sudo χρησιμοποιείται στο Mac και την εγκατάσταση Ubuntu) (Adams, 2014).

Σε περιβάλλον Windows πρέπει να πληκτρολογηθεί η παρακάτω εντολή:

```
pip install pygal
```

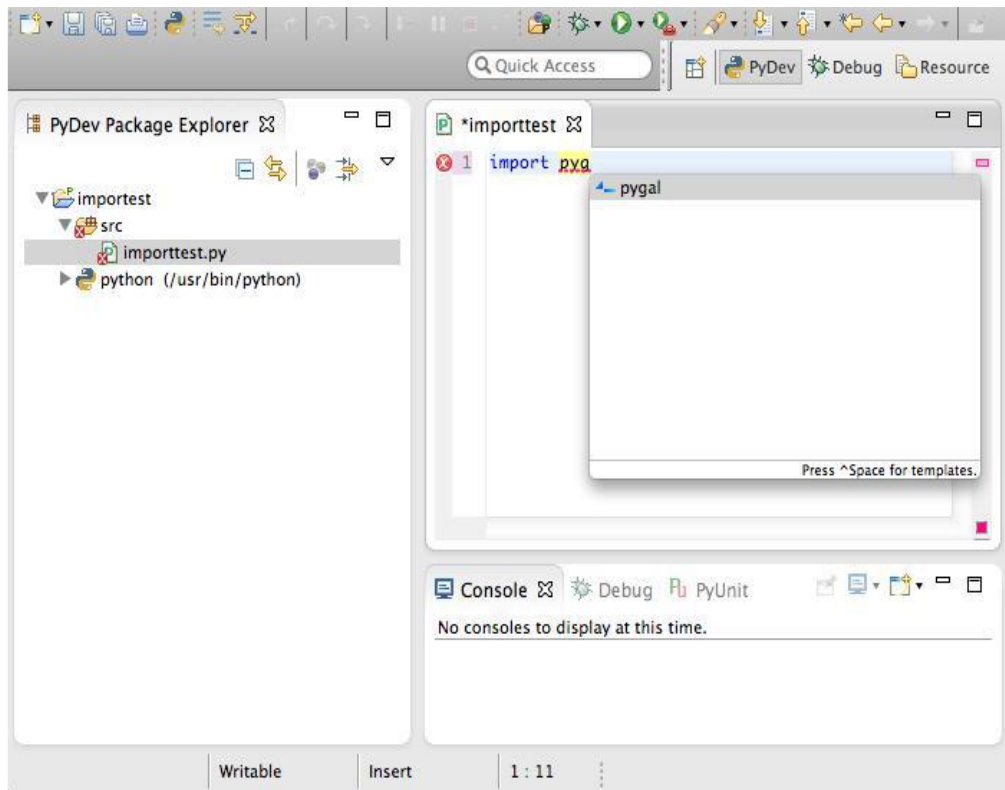
Σε περιβάλλον Mac ή Ubuntu πρέπει να πληκτρολογηθεί η παρακάτω εντολή:

```
sudo pip install pygal
```

Στη συνέχεια, εκκινείται το Eclipse μέσω του PyDev και εκεί πρέπει να δημιουργηθεί ένα νέο project Python, ακολουθούμενο από ένα νέο αρχείο. Μόλις το project δημιουργηθεί, πρέπει να δημιουργηθεί και το νέο αρχείο, το οποίο ονομάζεται importtest.py και πληκτρολογούνται τα παρακάτω:

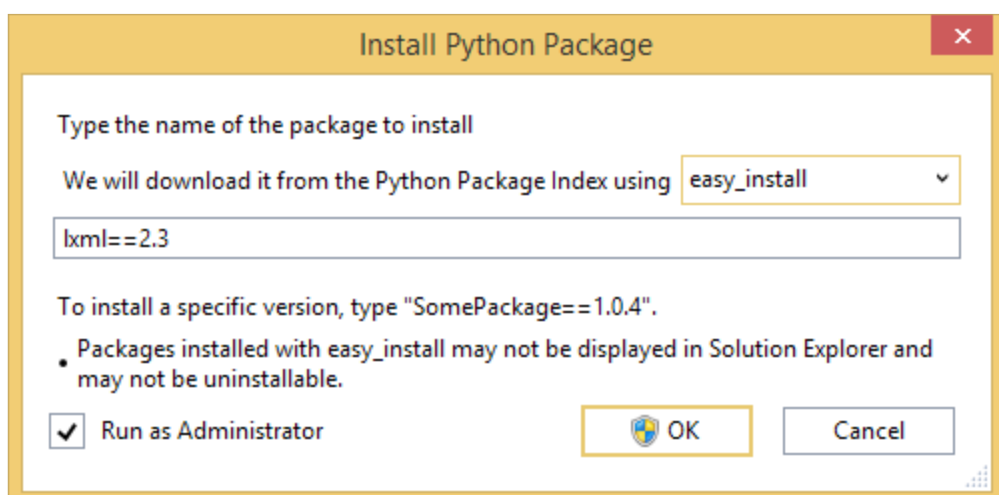
```
import pygal
```

Εάν η εγκατάσταση είναι επιτυχής, θα πρέπει να μπορεί ο χρήστης να πληκτρολογήσει Ctrl + Space bar και να δει τον κώδικα του PyDev να υποδεικνύει την άντληση όλων των βιβλιοθηκών που είναι εγκατεστημένες στο σύστημα. Στο παρακάτω στιγμιότυπο οθόνης, παρουσιάζεται η Pygal να αναγνωρίζεται από το σύστημα.



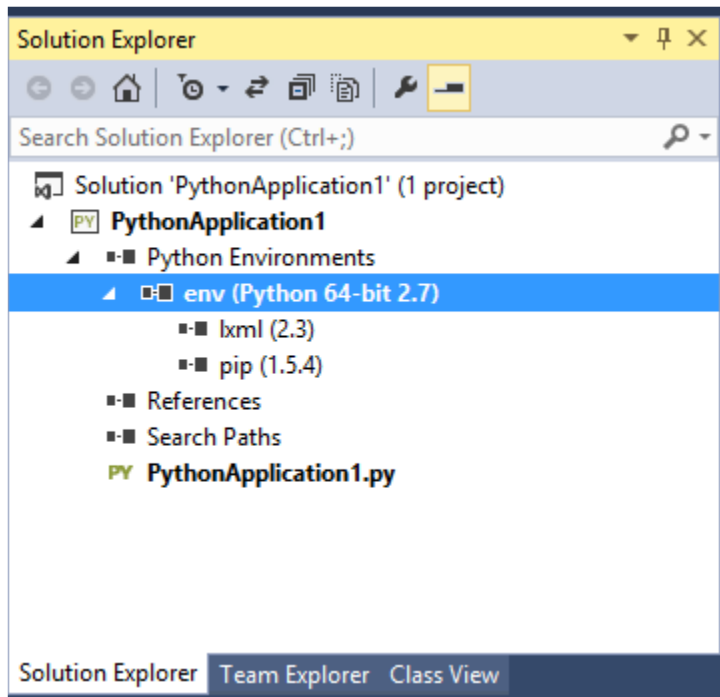
**Εικόνα 3.** Οθόνη εγκατεστημένης βιβλιοθήκης Pygal. Πηγή: Adams, 2014.

Στην περίπτωση κατά την οποία ο χρήστης σκοπεύει να εργαστεί στο Visual Studio, πρέπει να λάβει υπόψη του τα ακόλουθα σχετικά με την εγκατάσταση: πρώτον, εάν δεν έχει ήδη εγκαταστήσει το lxml, τότε πρέπει να εκτελέσει την εντολή `easy_install` με το δικό του περιβάλλον Python στο παράθυρο Install Python Package, όπως φαίνεται στο ακόλουθο στιγμιότυπο οθόνης:



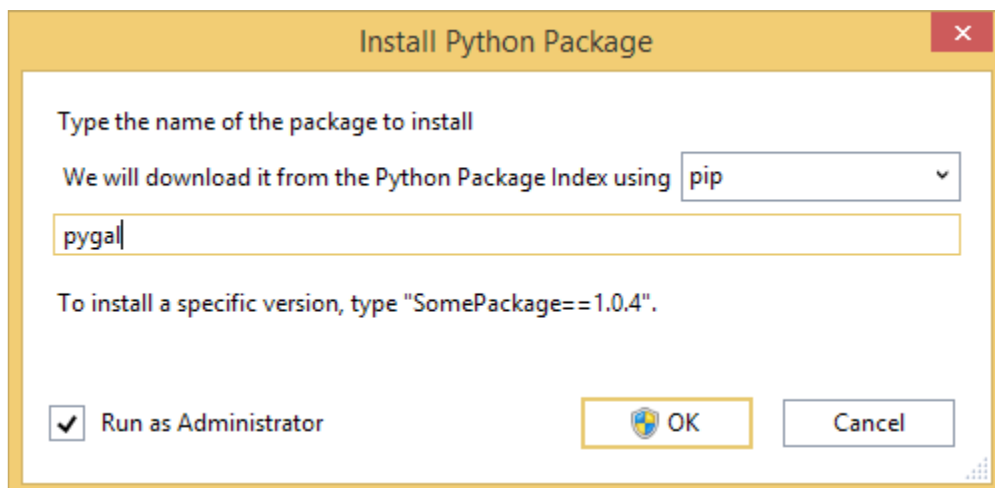
**Εικόνα 4.** Έναρξη εγκατάστασης lxml με τη χρήση του Python Tools. Πηγή: Adams, 2014.

Εάν η εγκατάσταση είναι επιτυχής, το παράθυρο του Solution Explorer θα πρέπει να μοιάζει με αυτό που εμφανίζεται στο παρακάτω στιγμιότυπο οθόνης, το οποίο συμπεριλαμβάνει το lxml:



*Εικόνα 5. Παράθυρο Solution Explorer με εγκατεστημένη τη lxml. Πηγή: Adams, 2014.*

Τελευταίο βήμα της διαδικασίας είναι η εγκατάσταση της βιβλιοθήκης Pygal. Γίνεται δεξί κλικ στο περιβάλλον και επιλέγεται Install Python Package, αυτή τη φορά με την Pygal, όπως φαίνεται στο παρακάτω στιγμιότυπο οθόνης:



*Εικόνα 6. Εγκατάσταση βιβλιοθήκης Pygal. Πηγή: Adams, 2014.*

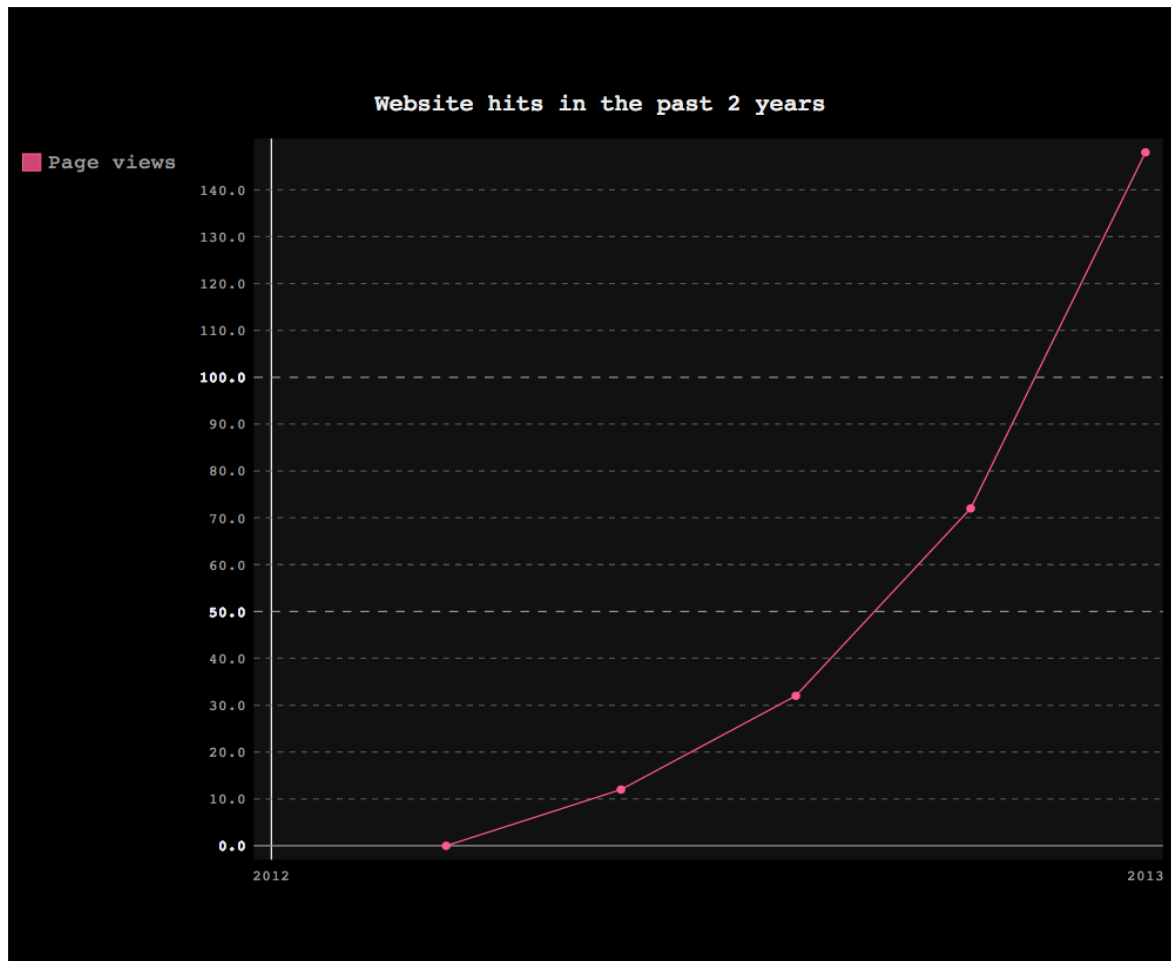
## 3.5 Εφαρμογές

### 3.5.1 Υλοποίηση ενός διαγράμματος γραμμής

Τα γραφήματα γραμμών δείχνουν συνήθως πως μεταβάλλονται συγκεκριμένα δεδομένα σε διαφορετικά χρονικά διαστήματα. Αποτελούν τα απλούστερα γραφήματα που μπορούν να υλοποιηθούν, συνήθως με τους x και y άξονες και κάθε άξονας στο γράφημα υποδεικνύει χρόνο, τιμή ή άλλη παράμετρο (Adams, 2014).

Στο επόμενο παράδειγμα υλοποιείται ένα απλό γράφημα, για το πόσες επισκέψεις έχει λάβει ένας ιστότοπος τα δύο τελευταία χρόνια (2012–2014). Η πρώτη γραμμή στον παρακάτω κώδικα είναι μία δηλωτική γραμμή από τον διερμηνέα Python για να καθορίσει τον τύπο της κωδικοποίησης συμβολοσειράς στο αρχείο. Επίσης, στο `line.x_labels` χρησιμοποιείται μία ενσωματωμένη συνάρτηση που ονομάζεται `range ()`. Αυτό επιτρέπει τη δημιουργία ενός πίνακα αριθμών, ξεκινώντας από τον χαμηλότερο αριθμό έως τον υψηλότερο, στην ακόλουθη περίπτωση, το 2012 και το 2014 θα εκτυπωθούν ως 2012, 2013, 2014 σε συστοιχία. Η αντιγραφή του ακόλουθου κώδικα στο κύριο αρχείο Python του έργου δημιουργεί το γράφημα της Εικόνας 7:

```
# -*- coding: utf-8 -*-  
  
import pygal  
  
#create a new line chart.  
  
line = pygal.Line()  
  
line.title = 'Website hits in the past 2 years' #set chart title  
  
line.x_labels = map(str, range(2012, 2014)) #set the x-axis labels.  
  
line.add('Page views', [None, 0, 12, 32, 72, 148]) #set values.  
  
line.render_to_file('linechart.svg') #set filename.
```



*Εικόνα 7. Βασικό διάγραμμα γραμμής της Pygal. Πηγή: Adams, 2014*

Στο κύριο αρχείο του έργου όπου εκτελέστηκε το σενάριο, υπάρχει το `linechart.svg` αρχείο που δημιουργήθηκε. Ο δείκτης του ποντικιού μπορεί να τοποθετηθεί πάνω από τις τελείες και να ληφθούν οι τιμές κάθε δείκτη στο γράφημα. Αυτές είναι ορισμένες από τις λειτουργίες που είναι προεγκατεστημένες με τη βιβλιοθήκη Pygal.

Επιπρόσθετα, μπορεί να παρατηρηθεί ότι το χρονοδιάγραμμα του γραφήματος ξεκινά από το 0,0 το 2013. Στη δήλωση `line.add()`, η πρώτη παράμετρος είναι `None`. Αυτό προσθέτει ένα διαχωριστικό στο διάγραμμα για να ωθήσει λίγο τα δεδομένα του γραφήματος αντί να αναγκάσει το γράφημα να ξεκινήσει το 2012. Αποτελεί μία κοινή πρακτική για τη ρύθμιση των διατάξεων γραφημάτων.

## Κεφαλαίο 4: Εισαγωγή στην JavaScript

### 4.1 Ιστορική αναδρομή

Η JavaScript είναι η πιο δημοφιλής scripting γλώσσα στον κόσμο. Είναι μια απλή, εύχρηστη, δυναμική γλώσσα προγραμματισμού που έχει αντικειμενοστραφείς ικανότητες. Ακόμη, είναι μια από τις πιο κοινές γλώσσες προγραμματισμού που χρησιμοποιείται σήμερα και η ζήτησή της έχει εκτοξευθεί. Παράλληλα με την HTML και τη CSS, η JavaScript είναι μία από τις βασικές τεχνολογίες του World Wide Web. Οι περισσότεροι ιστότοποι σήμερα έχουν δημιουργηθεί με τη χρήση κάποιος μορφής της JavaScript. (Flanagan,2021)

Ο Brendan Eich μέσα σε 10 ημέρες εφηύρε τη JavaScript το 1995 ως έναν τρόπο προγραμματισμού σε ένα απ' τα πρώτα δημοφιλή προγράμματα περιήγησης ιστού, στο Netscape Navigator. Στην αρχή είχε την επωνυμία Mocha και αργότερα μετονομάστηκε σε LiveScript, και τέλος σε JavaScript, κυρίως επειδή η ανάπτυξή της επηρεάστηκε περισσότερο από τη γλώσσα προγραμματισμού Java (Wirfs-Brock & Eich, 2020).

Το Σεπτέμβριο του 1995, το LiveScript ήταν το επίσημο όνομα της γλώσσας όταν για πρώτη φορά κυκλοφόρησε στην αγορά σε βήτα (beta) εκδόσεις με το πρόγραμμα περιήγησης στο Web, Netscape Navigator εκδοχή 2.0 (Wirfs-Brock & Eich, 2020).

Στις 4 Δεκεμβρίου, 1995 η Netscape σε κοινή απόφαση με την εταιρεία Sun Microsystems ανακοίνωσαν την μετονομασία της LiveScript σε JavaScript, όταν επεκτάθηκε στην έκδοση του προγράμματος περιήγησης στο Web, Netscape εκδοχή 2 (Wirfs-Brock & Eich, 2020).

Αργότερα, το Νοέμβριο του 1996, η Netscape ανακοίνωσε ότι είχε υποβάλει τη γλώσσα JavaScript στο Ecma International (μια οργάνωση της τυποποίησης των γλωσσών προγραμματισμού) για εξέταση ως βιομηχανικό πρότυπο, και στη συνέχεια το έργο είχε ως αποτέλεσμα την τυποποιημένη μορφή που ονομάζεται ECMAScript.

Η γλώσσα προγραμματισμού JavaScript συνεχίζει να εξελίσσεται με κάθε έκδοση του ECMAScript και ενισχύθηκε για να διατηρήσει την ισχύ της στο παρόν

καθώς και στο μέλλον της ανάπτυξης του διαδικτύου( Ranjan, Sinha, & Battewad, 2020).

## 4.2 Η δράση της JavaScript

Με την πάροδο του χρόνου και άλλα προγράμματα περιήγησης έχουν υιοθετήσει την JavaScript ως κύρια γλώσσα για τη δημιουργία ιστοσελίδων. Η JavaScript είναι μία διαδομένη δυναμική, scripting γλώσσα προγραμματισμού για τη δημιουργία εφαρμογών και διαδραστικών (interactive) ιστοσελίδων. Είναι μια ερμηνευόμενη γλώσσα που σημαίνει πως τα scripts εκτελούνται χωρίς να έχει προηγηθεί μεταγλώττιση του κώδικα και υποστηρίζεται απο όλους τους δημοφιλής φυλλομετρητές (browsers). Υποστηρίζεται από όλους τους δημοφιλείς περιηγητές, όπως οι Microsoft Internet Explorer (ξεκινώντας με την έκδοση 3.0), Firefox, Safari, Opera, Google Chrome και άλλους (Flanagan,2021).

Το συντακτικό της JavaScript είναι παρόμοιο με αυτό της C. Όπως και η C έτσι και η JavaScript δεν έχει δομές εισόδου ή εξόδου (input, output) από μόνη της. Η μηχανή της JavaScript στηρίζεται σε ένα ενσωματωμένο στο φυλλομετρητή (Wirfs-Brock & Eich, 2020).



Μοιράζεται μερικές ιδιότητες των γλωσσών Java και C , αλλά από μόνη της αποτελεί μια διαφορετική, αυτούσια γλώσσα προγραμματισμού. Βασίζεται στη χρήση συναρτήσεων πρώτης τάξης (συναρτήσεις μπορούν να δίνονται ως ορίσματα σε άλλες συναρτήσεις για την επιστροφή ή όχι τιμών), γεγονός που την κάνει να ακολουθεί τις αρχές τόσο του συναρτησιακού, όσο και του αντικειμενοστραφούς προγραμματισμού.



Αρχικά χρησιμοποιήθηκε για προγραμματισμό από την πλευρά του πελάτη (client) ως client-side γλώσσα προγραμματισμού που σημαίνει πως η σύνταξη του κώδικα και η παραγωγή του HTML περιεχομένου δεν πραγματοποιείται στον διακομιστή (server) αλλά στο πρόγραμμα περιήγησης των επισκεπτών (Flanagan,2021).

### 4.3 Πλεονεκτήματα και Μειονεκτήματα της JavaScript

Για την καλύτερη χρήση της Javascript και για τη δημιουργία εφαρμογών είναι απαραίτητο η γνώση των πλεονεκτημάτων και τα μειονεκτημάτων της Javascript,. Παρακάτω παρατίθενται κάποια πλεονεκτήματα και μειονεκτήματα της Javascript (Καλλινικος Γεωργιος, 2018).

#### Πλεονεκτήματα

- **Επεξεργασία από τον πελάτη (client-side):** Αυτό σημαίνει ότι ο κώδικας εκτελείται στον επεξεργαστή του χρήστη αντί του διακομιστή ιστού, εξοικονομώντας έτσι εύρος ζώνης και μειώνοντας το επιπλέον φορτίο του διακομιστή.
- **Απλή να τη μάθουν:** Η σύνταξη αυτής της γλώσσας είναι παρόμοια με απλά αγγλικά που διευκολύνουν τους προγραμματιστές να μάθουν.
- **Απλή στην εφαρμογή της:** Η δυνατότητα χρήσης της ίδιας γλώσσας front-end και back-end διευκολύνει τους προγραμματιστές.
- **Δεν χρειάζεται μεταγλώττιση:** Το πρόγραμμα περιήγησης ερμηνεύει τη JavaScript ως ετικέτες HTML.
- **Γρήγορη για τον τελικό χρήστη:** Με τη JavaScript, κάθε πεδίο μπορεί να επαληθεύεται καθώς συμπληρώνεται από τους χρήστες, γεγονός που παρέχει άμεση ανατροφοδότηση, όταν αυτοί κάνουν κάποιο λάθος(Καλλινικος Γεωργιος, 2018).

## Μειονεκτήματα

- **Ασφάλεια από την πλευρά του πελάτη :** Εφόσον ο κώδικας JavaScript είναι ορατός στον χρήστη, άλλοι ενδέχεται να τον χρησιμοποιήσουν για κακόβουλους σκοπούς. Αυτές οι πρακτικές μπορεί να περιλαμβάνουν τη χρήση του πηγαίου κώδικα χωρίς έλεγχο ταυτότητας. Επίσης, είναι πολύ εύκολο να τοποθετήσετε κάποιο κώδικα στον ιστότοπο που θέτει σε κίνδυνο την ασφάλεια των δεδομένων του ιστότοπου.
- **Υποστήριξη προγράμματος περιήγησης:** Το πρόγραμμα περιήγησης ερμηνεύει διαφορετικά τη JavaScript σε διαφορετικά προγράμματα περιήγησης. Επομένως, ο κώδικας πρέπει να εκτελείται σε διάφορες πλατφόρμες πριν από τη δημοσίευση. Τα παλαιότερα προγράμματα περιήγησης δεν υποστηρίζουν ορισμένες νέες λειτουργίες και πρέπει να τις ελέγξουμε επίσης
- **Έλλειψη δυνατότητας εντοπισμού σφαλμάτων:** Αν και ορισμένοι επεξεργαστές HTML υποστηρίζουν τον εντοπισμό σφαλμάτων, δεν είναι τόσο αποτελεσματικοί όσο άλλοι επεξεργαστές όπως οι επεξεργαστές C/C++. Επίσης, καθώς το πρόγραμμα περιήγησης δεν εμφανίζει κανένα σφάλμα, είναι δύσκολο για τον προγραμματιστή να εντοπίσει το πρόβλημα.
- **Ενιαία κληρονομιά:** Η JavaScript υποστηρίζει μόνο μεμονωμένη κληρονομικότητα και όχι πολλαπλή κληρονομικότητα. Ορισμένα προγράμματα ενδέχεται να απαιτούν αυτό το χαρακτηριστικό αντικειμενοστραφής γλώσσας.
- **Σταμάτημα απόδοσης :** Ένα μόνο σφάλμα κώδικα μπορεί να σταματήσει την απόδοση ολόκληρου του κώδικα JavaScript στον ιστότοπο. Για τον χρήστη, φαίνεται σαν να μην υπήρχε JavaScript. Ωστόσο, τα προγράμματα περιήγησης είναι εξαιρετικά ανεκτικά σε αυτά τα σφάλματα.

## Κεφαλαίο 5: Βιβλιοθήκες οπτικοποίησης D3.js

### 5.1 Τι είναι η βιβλιοθήκη D3.js

Η d3.js είναι μία βιβλιοθήκη της JavaScript για τη διαδικτυακή οπτικοποίηση δεδομένων. Αναπτύχθηκε από τον Mike Bostock για τη δημιουργία διαδραστικών οπτικοποιήσεων στον φυλλομετρητή χρησιμοποιώντας SVG, HTML και CSS. Η D3 είναι ανοιχτού κώδικα και δίνει την δυνατότητα στους προγραμματιστές να δημιουργήσουν ένα διαδραστικό και κινούμενο περιεχόμενο βασισμένο σε δεδομένα και να το συνδέουν με την ιστοσελίδα (Meeks, 2017).

Με τον τεράστιο όγκο δεδομένων που παράγονται σήμερα, η επικοινωνία αυτών των πληροφοριών γίνεται δύσκολη. Οι οπτικές αναπαραστάσεις δεδομένων είναι το πλέον αποτελεσματικό μέσο για τη μετάδοση ουσιαστικών πληροφοριών και η βιβλιοθήκη d3.js παρέχει μεγάλη ευκολία και ευελιξία για τη δημιουργία των συγκεκριμένων απεικονίσεων δεδομένων. Είναι δυναμική, διαισθητική και με μικρό κόστος μπορεί να υποστηρίξει αρκετά γρήγορα μεγάλα σύνολα δεδομένων καθώς και δυναμικές τακτικές για αλληλεπίδραση και animation (Meeks, 2017).

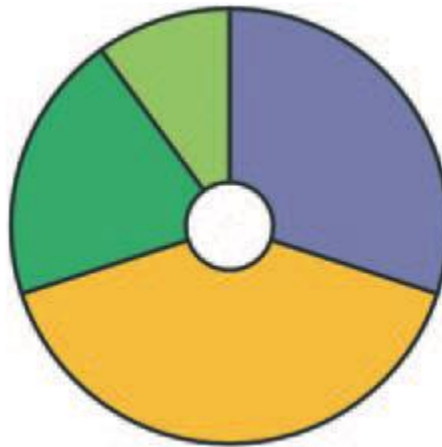
Η d3.js δεν παρέχει κάθε πιθανή λειτουργία οπτικοποίησης και ούτε αντιμετωπίζει το οποιοδήποτε πρόβλημα. Αντίθετα όμως, η d3.js επεξεργάζεται κείμενα με βάση τα δεδομένα. Αυτό την καθιστά περισσότερο ελαστική, καθώς μπορεί να παρουσιάσει πολλές από τις δυνατότητες των προτύπων του Διαδικτύου, όπως η HTML και η SVG. Τέλος η d3.js υποστηρίζει όλους τους σύγχρονους φυλλομετρητές, καθώς και τα λειτουργικά συστήματα Android και IOS (Meeks, 2017).

### 5.2 Τομείς που χρησιμοποιείται η d3.js

Ένα από τα βασικά πλεονεκτήματα της d3.js είναι ότι επιτρέπει τη δημιουργία διανυσματικών γραφικών για τη δημιουργία παραδοσιακών γραφημάτων, αλλά και τη δημιουργία γεωχωρικών και δικτυακών οπτικοποιήσεων, ενώ διαθέτει και σημαντικές δυνατότητες αναφορικά με τη δημιουργία κινούμενων εικόνων και τη διαδραστικότητα. Αυτή η ευρεία προσέγγιση για την οπτικοποίηση δεδομένων, όπου ένας χάρτης ή ένα γράφημα δικτύου ή ένας πίνακας αποτελούν διάφορα είδη

αναπαράστασης δεδομένων, είναι ο πυρήνας της απήχησης της βιβλιοθήκης d3.js αναφορικά με την ανάπτυξη εφαρμογών (Meeks, 2017).

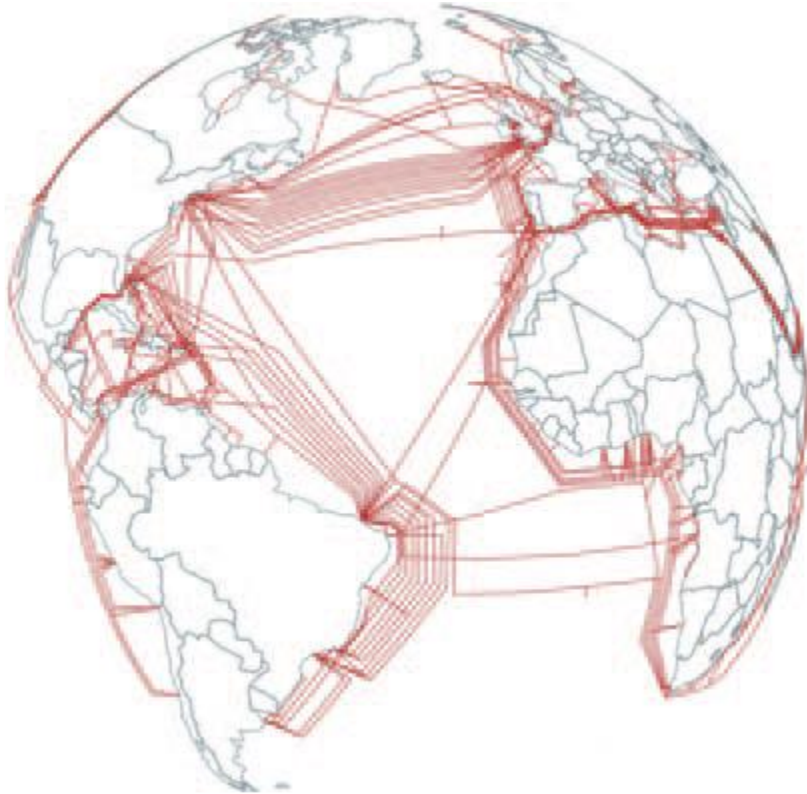
Στις Εικόνες 8-14 παρουσιάζονται διάφορα παραδείγματα οπτικοποίησης δεδομένων που δημιουργήθηκαν με τη βιβλιοθήκη d3.js. Περιλαμβάνουν χάρτες και δίκτυα, μαζί με περισσότερο παραδοσιακά γραφήματα πίτας και διατάξεις εντελώς προσαρμοσμένης οπτικοποίησης δεδομένων με βάση τις συγκεκριμένες ανάγκες των χρηστών.



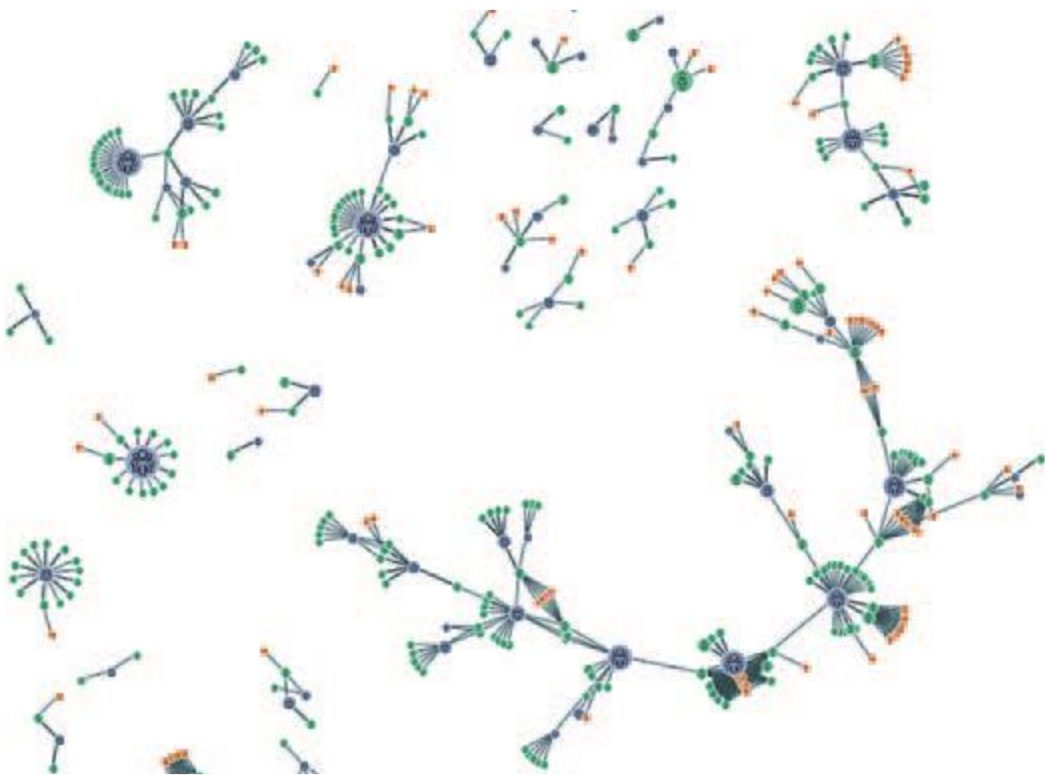
**Εικόνα 8.** Απλό γράφημα που υλοποιήθηκε με τη βιβλιοθήκη D3. Πηγή: Meeks, 2017.



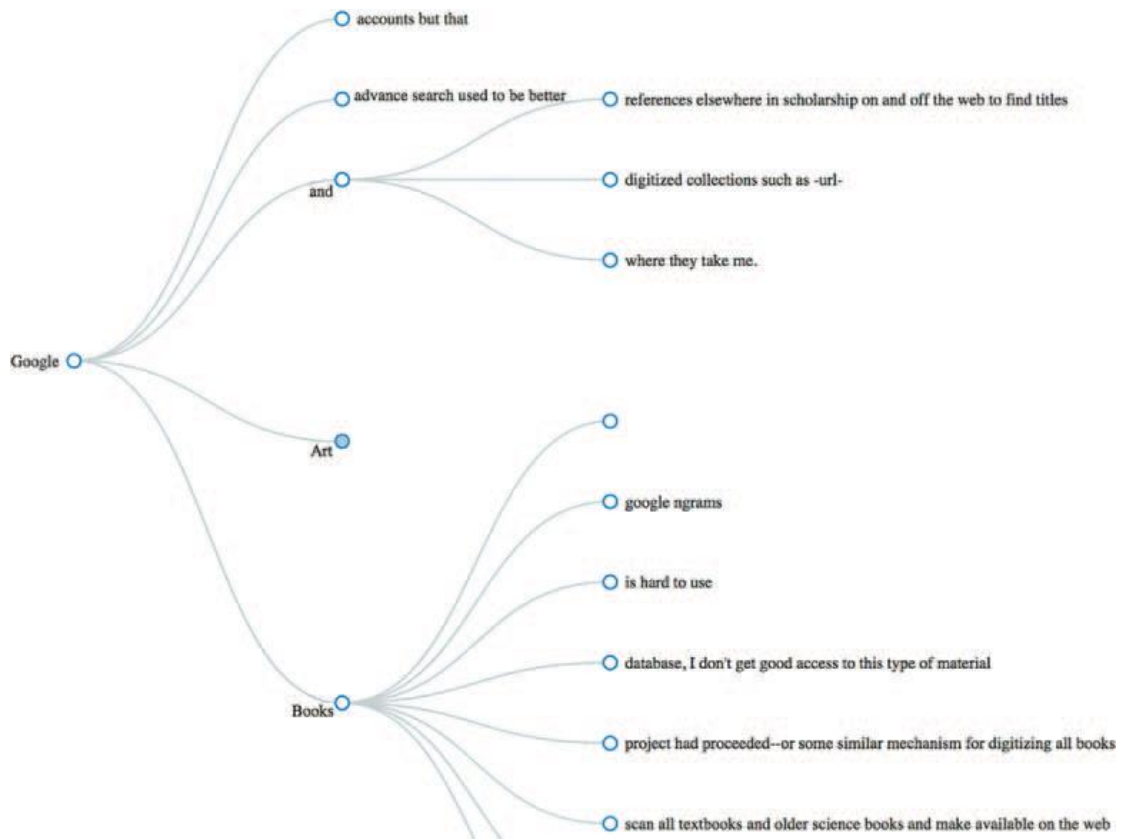
**Εικόνα 9.** Διαδικτυακός χάρτης ο οποίος παρουσιάζει την εθνολογική σύνθεση των σημαντικότερων πόλεων των ΗΠΑ και υλοποιήθηκε με τη βιβλιοθήκη D3. Πηγή: Meeks, 2017.



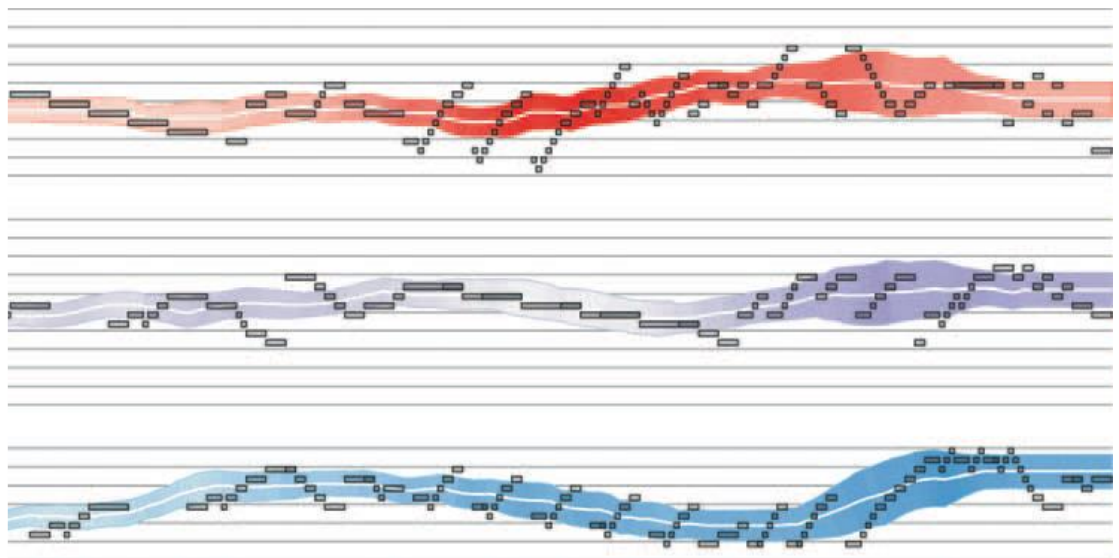
*Εικόνα 10. Διαδραστικός χάρτης υποθαλάσσιων καλωδίων τηλεπικοινωνιών που υλοποιήθηκε με τη D3. Πηγή: Meeks, 2017.*



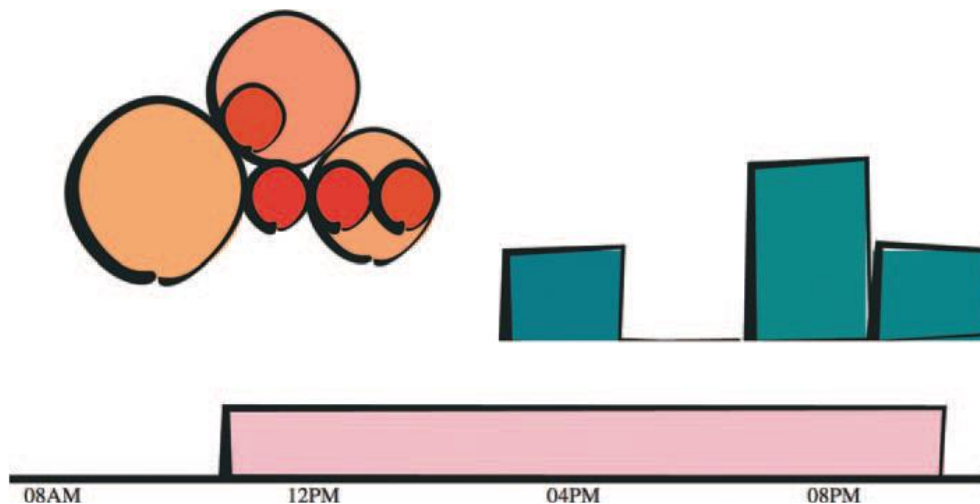
*Εικόνα 11. Δίκτυο συνεργασιών των αρχαιολόγων που εργάζονται στην ίδια ανασκαφή για περίπου 25 χρόνια το οποίο υλοποιήθηκε με τη D3. Πηγή: Meeks, 2017.*



*Εικόνα 12. Δέντρο αναπαράστασης δεδομένων που υλοποιήθηκε με τη D3. Πηγή: Meeks, 2017.*



*Εικόνα 13. Αναπαράσταση ηχητικών κυμάτων που υλοποιήθηκε με τη D3. Πηγή: Meeks, 2017.*



*Εικόνα 14. Εξατομικευμένο γράφημα μπάρας που υλοποιήθηκε με τη D3. Πηγή: Meeks, 2017.*

Αν και η ικανότητα δημιουργίας πλούσιων και ποικίλων γραφικών είναι ένα από τα πλεονεκτήματα της d3.js, το πλέον σημαντικό χαρακτηριστικό για τις σύγχρονες εφαρμογές οπτικοποίησης δεδομένων είναι η δυνατότητα ενσωμάτωσης της διαδραστικότητας υψηλού επιπέδου που περιμένουν οι χρήστες. Με την d3.js , κάθε στοιχείο κάθε γραφήματος, από μία περιστρεφόμενη υδρόγειο έως ένα ενιαίο, λεπτό κομμάτι ενός γραφήματος πίτας, γίνεται διαδραστικό κατά τον ίδιο τρόπο. Επιπρόσθετα, η βιβλιοθήκη d3.js περιλαμβάνει διαδραστικά στοιχεία και συμπεριφορές που είναι τυπικές σε εφαρμογές που σχετίζονται με την οπτικοποίηση δεδομένων και την ανάπτυξη ιστότοπων (Meeks, 2017).

Η βιβλιοθήκη D3 απαιτεί σημαντικό χρόνο εκμάθησης. Ως εκ τούτου, δεν είναι η πλέον κατάλληλη λύση για τη δημιουργία απλών γραφημάτων, τα οποία μπορούν να υλοποιηθούν με απλούστερες βιβλιοθήκες. Η D3 δίνει τη δυνατότητα εφαρμογής σχεδόν κάθε σημαντικής – σύνθετης τεχνικής οπτικοποίησης δεδομένων. Επίσης δίνει τη δυνατότητα δημιουργίας εξατομικευμένων τεχνικών οπτικοποίησης δεδομένων, κάτι το οποίο δεν είναι δυνατό να γίνει με μία γενική βιβλιοθήκη (Meeks, 2017).

Η βιβλιοθήκη d3.js παρέχει στους προγραμματιστές τη δυνατότητα να υλοποιήσουν όχι μόνο πλούσιες διαδραστικές εφαρμογές αλλά και εφαρμογές που έχουν τη μορφή και παρουσιάζονται όπως το παραδοσιακό περιεχόμενο ιστού. Αυτό τις καθιστά περισσότερο φορητές, περισσότερο προσαρμόσιμες στον αναπτυσσόμενο, συνδεδεμένο ιστό δεδομένων, ενώ συντηρούνται πιο εύκολα από μεγάλες ομάδες όπου τα άλλα μέλη της ομάδας δεν γνωρίζουν τη συγκεκριμένη

σύνταξη της d3, αλλά, για παράδειγμα, μπορεί να χρησιμοποιήσουν CSS για το στυλ των στοιχείων της οπτικοποίησης δεδομένων (Meeks, 2017).

### 5.3 Χαρακτηριστικά της d3.js

Η βιβλιοθήκη d3.js δεν αποτελεί ένα παραδοσιακό πλαίσιο που επιδιώκει να παράσχει κάθε δυνατή λειτουργία. Αντιθέτως, επιλύει την ουσία του προβλήματος, εφόσον χειρίζεται αποτελεσματικά έγγραφα που βασίζονται σε δεδομένα. Έτσι, παρέχει εξαιρετική ευελιξία και πλήρης αξιοποίηση των δυνατοτήτων προτύπων Ιστού όπως HTML, SVG και CSS. Με ελάχιστη επιβάρυνση, η d3.js μπορεί να γίνει εξαιρετικά γρήγορη, υποστηρίζοντας μεγάλα σύνολα δεδομένων και δυναμικές συμπεριφορές για αλληλεπίδραση και κίνηση. Το λειτουργικό στυλ των αρχείων d3.js επιτρέπει την επαναχρησιμοποίηση κώδικα μέσω μίας διαφοροποιημένης συλλογής από στοιχεία και συνδέσεις (plugins). Καθώς το μοντέλο εγγράφου ορίζει άμεσα γραφικά θεμελιακά στοιχεία, η d3.js επιφέρει επίσης μία ομοιότητα με βιβλιοθήκες γραφικών χαμηλού επιπέδου όπως οι Processing και Raphael. Για τη δυνατότητα υψηλού επιπέδου, η d3.js περιλαμβάνει μία συλλογή από βοηθητικά στοιχεία που βρίσκονται στην κορυφή του πυρήνα. Τα στοιχεία αυτά επηρεάζονται σε μεγάλο βαθμό από προγενέστερα συστήματα οπτικοποίησης, συμπεριλαμβανομένου του Protovis (Bostock et al., 2011).

Τα κυριότερα πλεονεκτήματα της d3.js μπορούν να συνοψιστούν ως εξής:

- Η d3.js είναι μία βιβλιοθήκη Javascript. Έτσι, μπορεί να χρησιμοποιηθεί με οποιοδήποτε πλαίσιο JS, όπως Angular.js, React.js ή Ember.js.
- Εστιάζει στα δεδομένα, επομένως είναι το πλέον κατάλληλο και εξειδικευμένο εργαλείο για οπτικοποιήσεις δεδομένων.
- Είναι ανοιχτού κώδικα. Έτσι, είναι δυνατή η εργασία με τον πηγαίο κώδικα και η προσθήκη των επιθυμητών δυνατοτήτων από τον προγραμματιστή.
- Λειτουργεί με πρότυπα ιστού, επομένως δεν χρειάζεται άλλη τεχνολογία ή πρόσθετο εκτός από ένα πρόγραμμα περιήγησης για τη χρησιμοποίηση της d3.js.
- Λειτουργεί με πρότυπα ιστού όπως HTML, CSS και SVG, δεν απαιτείται νέο εργαλείο εκμάθησης ή εντοπισμού σφαλμάτων για την εργασία στη d3.js.
- Δεν παρέχει καμία συγκεκριμένη δυνατότητα, επομένως δίνει τον πλήρη έλεγχο της οπτικοποίησής στον προγραμματιστή προκειμένου εκείνος να την προσαρμόσει όπως



θέλει. Με τον τρόπο αυτό αποκτά ένα πλεονέκτημα σε σχέση με άλλα δημοφιλή εργαλεία όπως το Tableau ή το QlikView.

- Δεδομένου ότι η d3.js λειτουργεί άμεσα με πρότυπα web, είναι εξαιρετικά γρήγορη και λειτουργεί καλά με μεγάλα σύνολα δεδομένων.

## 5.4 Εγκατάσταση της d3.js

Για την εγκατάσταση της d3.js απαιτούνται τα παρακάτω στοιχεία:

- Βιβλιοθήκη d3.js
- Συντάκτης (Editor)
- Φυλλομετρητής διαδικτύου (web browser)
- Εξυπηρετητής (web server)

Αρχικά, πρέπει να συμπεριληφθεί η βιβλιοθήκη D3.js στην ιστοσελίδα HTML προκειμένου να χρησιμοποιηθεί η D3.js για τη δημιουργία εφαρμογών οπτικοποίησης δεδομένων. Η διαδικασία μπορεί να λάβει χώρα με τους εξής δύο τρόπους:

- Επισύναψη της βιβλιοθήκης d3.js στον φάκελο του έργου.
- Επισύναψη της βιβλιοθήκης d3.js στο CDN (Δίκτυο παράδοσης περιεχομένου – Content Delivery Network).

Η d3.js είναι μία βιβλιοθήκη ανοιχτού κώδικα και ο πηγαίος κώδικας της βιβλιοθήκης είναι ελεύθερα διαθέσιμος στον ιστότοπο <https://d3js.org/>. Μέσα από τον συγκεκριμένο ιστότοπο μπορεί να πραγματοποιηθεί λήψη της πιο πρόσφατης έκδοσης του d3.js (d3.zip). Προς το παρόν, η τελευταία έκδοση είναι η 4.6.0.

Αφού ολοκληρωθεί η λήψη, το αρχείο αποσυμπίεζεται και αναζητείται το αρχείο d3.min.js. Αυτή είναι η ελαχιστοποιημένη έκδοση του πηγαίου κώδικα d3.js. Αντιγράφεται το αρχείο d3.min.js και επικολλείται στον κύριο φάκελο του έργου ή σε οποιονδήποτε άλλο φάκελο, όπου ο χρήστης επιθυμεί να διατηρήσει όλα τα αρχεία της βιβλιοθήκης. Το αρχείο d3.min.js επισυνάπτεται στη σελίδα HTML με τις παρακάτω εντολές:

```
<!DOCTYPE html>
<html lang = "en">
  <head>
    <script src = "/path/to/d3.min.js"></script>
  </head>
  <body>
    <script>
      // write your d3 code here..
    </script>
  </body>
</html>
```

Η d3.js είναι μια βιβλιοθήκη JavaScript, επομένως πρέπει να γραφεί όλος ο κώδικας d3 μέσα στην ετικέτα "script". Μπορεί να χρειαστεί να επεξεργαστούν τα υπάρχοντα στοιχεία DOM, επομένως συνιστάται να γραφεί ο κώδικας d3 λίγο πριν το τέλος της ετικέτας "body".

Μπορεί να χρησιμοποιηθεί η βιβλιοθήκη d3.js συνδέοντάς την απευθείας στη σελίδα HTML από το Δίκτυο Παράδοσης Περιεχομένου (CDN). Το CDN είναι ένα δίκτυο διακομιστών όπου φιλοξενούνται αρχεία και παραδίδονται σε έναν χρήστη με βάση τη γεωγραφική του θέση. Εάν χρησιμοποιηθεί το CDN, δεν χρειάζεται να μεταφορτωθεί ο πηγαίος κώδικας.

Η βιβλιοθήκη d3.js επισυνάπτεται χρησιμοποιώντας τη διεύθυνση URL του CDN <https://d3js.org/d3.v4.min.js> όπως φαίνεται παρακάτω:

```
<!DOCTYPE html>

<html lang = "en">

  <head>

    <script src = "https://d3js.org/d3.v4.min.js"></script>

  </head>

  <body>

    <script>

      // write your d3 code here..

    </script>

  </body>

</html>
```

Θα χρειαστεί ένας επεξεργαστής για να ξεκινήσει να γράφεται ο κώδικας . Υπάρχουν διάφορα IDE (Integrated Development Environment) με υποστήριξη για JavaScript όπως για παράδειγμα τα παρακάτω:

- Visual Studio Code
- WebStorm
- Eclipse
- Sublime Text

Αυτά τα IDE παρέχουν τη δυνατότητα έξυπνης συμπλήρωσης κώδικα καθώς και υποστηρίζουν ορισμένα από τα σύγχρονα frameworks. Εναλλακτικά, μπορεί πάντα να χρησιμοποιηθεί ένα βασικό πρόγραμμα επεξεργασίας όπως το Σημειωματάριο, το VI, κ.λπ.

Η D3.js λειτουργεί σε όλα τα προγράμματα περιήγησης εκτός από το IE8 και παλαιότερες εκδόσεις.

Τα περισσότερα προγράμματα περιήγησης εξυπηρετούν τοπικά αρχεία HTML απευθείας από το τοπικό σύστημα αρχείων. Ωστόσο, υπάρχουν ορισμένοι περιορισμοί όσον αφορά τη φόρτωση εξωτερικών αρχείων δεδομένων. Μπορεί να χρησιμοποιηθεί οποιοσδήποτε διακομιστής ιστού με τον οποίο αισθάνεται ο χρήστης άνετα – π.χ. IIS, Apache, κ.λπ.

Στις περισσότερες περιπτώσεις, ο χρήστης απλά ανοίγει το αρχείο HTML σε ένα πρόγραμμα περιήγησης ιστού για να το προβάλει. Ωστόσο, κατά τη φόρτωση εξωτερικών πηγών δεδομένων, είναι πιο αξιόπιστη η εκτέλεση ενός τοπικού διακομιστή web και η προβολή της σελίδας από τον διακομιστή (<http://localhost:8080>).

## 5.5 Εφαρμογές της d3.js

Έστω ένα σύνολο δεδομένων, όπως η τιμή και το μέγεθος μερικών σπιτιών, και ένα σύνολο στοιχείων ιστοσελίδας, είτε γραφικά είτε στοιχεία <div>, και στόχος είναι η αναπαράσταση του συνόλου δεδομένων, είτε με κείμενο είτε μέσω μεγέθους και χρώματος. Μία επιλογή είναι η ομάδα των δεδομένων και των στοιχείων μαζί. Στα στοιχεία της ομάδας μπορούν να εκτελεστούν ενέργειες όπως η μετακίνησή τους ή η αλλαγή του χρώματός τους. Μπορούν επίσης να ενημερωθούν οι τιμές των δεδομένων. Αν και είναι δυνατή η εργασία με τα δεδομένα και τα στοιχεία της ιστοσελίδας ξεχωριστά, η πραγματική δύναμη της d3.js προέρχεται από τη χρήση επιλογών για τον συνδυασμό δεδομένων και στοιχείων ιστοσελίδας.

Παρακάτω παρουσιάζεται μία επιλογή χωρίς δεδομένα:

```
d3.selectAll("circle.a").style("fill", "red").attr("cx", 100);
```

Με τον τρόπο αυτό ο κάθε κύκλος της σελίδας με την κλάση CSS του a επεξεργάζεται, γίνεται κόκκινος και μετακινείται έτσι ώστε το κέντρο του να είναι 100 pixel στα δεξιά της αριστερής πλευράς του καμβά <svg>. Επίσης, αυτός ο κώδικας κάνει κάθε div στην ιστοσελίδα κόκκινο και αλλάζει την κλάση του σε b:

```
d3.selectAll("div").style("background", "red").attr("class", "b");
```

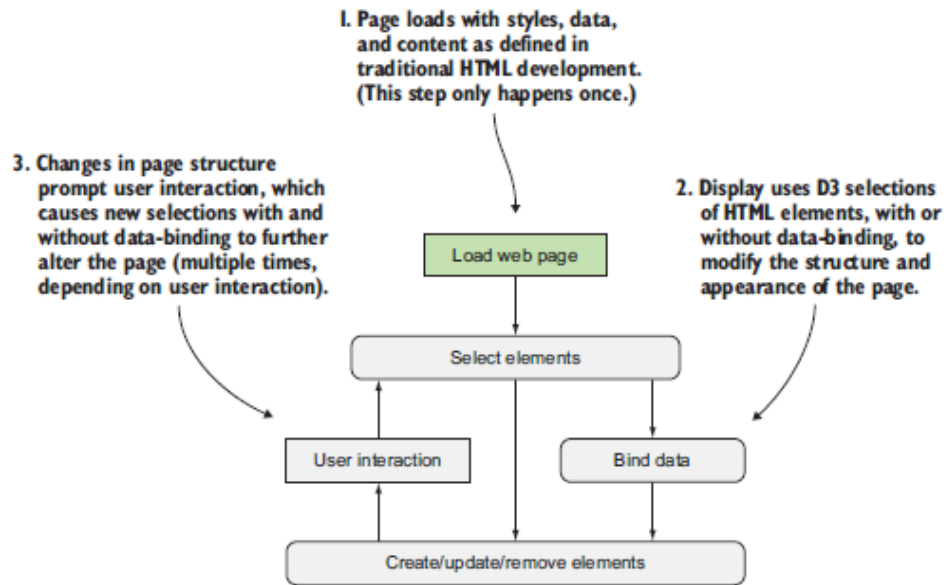
Το πρώτο μέρος αυτής της γραμμής κώδικα, d3.selectAll(), είναι μέρος της απαραίτητης βασικής λειτουργικότητας για την κατανόηση των επιλογών της D3. Οι επιλογές μπορούν να γίνουν με την εντολή d3.select(), που επιλέγει το πρώτο μεμονωμένο στοιχείο που βρέθηκε, αλλά πιο συχνά χρησιμοποιείται η d3.selectAll(),

που μπορεί να χρησιμοποιηθεί για την επιλογή πολλών στοιχείων. Οι επιλογές (selections) είναι ομάδες ενός ή περισσότερων στοιχείων ιστοσελίδας που μπορεί να συσχετίζονται με ένα σύνολο δεδομένων, όπως ο ακόλουθος κώδικας, που δεσμεύει τα στοιχεία του πίνακα [1,5,11,3] σε στοιχεία <div> με την κλάση της αγοράς:

### **d3.selectAll("div.market").data([1,5,11,3])**

Αυτή η συσχέτιση είναι γνωστή ως δέσμευση δεδομένων ενώ η επιλογή μπορεί να θεωρηθεί ως σύνολο στοιχείων ιστοσελίδας και ένα αντίστοιχο, συσχετισμένο σύνολο δεδομένων. Μερικές φορές υπάρχουν περισσότερα στοιχεία δεδομένων από τα στοιχεία DOM ή το αντίστροφο, οπότε η d3.js έχει συναρτήσεις που έχουν σχεδιαστεί για να δημιουργούν ή να αφαιρούν στοιχεία που μπορούν να χρησιμοποιηθούν για τη δημιουργία περιεχομένου. Μπορεί να χρησιμοποιηθεί μία επιλογή σε οποιαδήποτε στοιχεία σε μία ιστοσελίδα, συμπεριλαμβανομένων στοιχείων σε μία λίστα, κύκλων ή ακόμα και περιοχών σε έναν χάρτη της Αφρικής. Με τον ίδιο τρόπο που τα στοιχεία μπορούν να πάρουν μία σειρά από σχήματα, τα δεδομένα που σχετίζονται με αυτά τα στοιχεία (όπου ισχύει) μπορούν να πάρουν πολλές μορφές.

Μετά τη δημιουργία μίας επιλογής, μπορεί να χρησιμοποιηθεί η d3.js για την τροποποίηση της εμφάνισης στοιχείων της ιστοσελίδας που αντικατοπτρίζουν τις διαφορές στα δεδομένα. Μπορεί το μήκος μίας γραμμής να γίνει ίσο με την τιμή των δεδομένων ή να αλλάξει το χρώμα σε αυτό που αντιστοιχεί σε μία κλάση των δεδομένων. Μπορούν να κρυφτούν ή να εμφανιστούν στοιχεία καθώς αντιστοιχούν στην πλοήγηση ενός χρήστη ενός συνόλου δεδομένων. Όπως φαίνεται στην Εικόνα 15, μετά τη φόρτωση της σελίδας, χρησιμοποιείται το d3.js για την επιλογή στοιχείων και τη δέσμευση δεδομένων για δημιουργία, αφαίρεση ή αλλαγή στοιχείων DOM.



**Εικόνα 15.** Αξιοποίηση των δυνατοτήτων της D3 για την ενημέρωση του περιεχομένου της οπτικοποίησης δεδομένων σύμφωνα με την αλληλεπίδραση με τον χρήστη. Πηγή: Meeks, 2017.

Η εμφάνιση των στοιχείων μπορεί να τροποποιηθεί χρησιμοποιώντας επιλογές που σχετίζονται με την αναφορά στα δεδομένα που δεσμεύονται σε ένα στοιχείο σε μία επιλογή. Η d3.js επαναλαμβάνεται μέσω των στοιχείων της επιλογής και εκτελεί την ίδια ενέργεια χρησιμοποιώντας τα δεσμευμένα δεδομένα, με αποτέλεσμα τη δημιουργία διαφορετικών γραφικών αναπαραστάσεων. Αν και η ενέργεια που εκτελείται είναι η ίδια, το αποτέλεσμα είναι διαφορετικό επειδή βασίζεται στη διακύμανση των δεδομένων.

Στα πλαίσια της κλασικής προσέγγισης οι ιστοσελίδες αποτελούνται από στοιχεία κειμένου με πλαίσια για φωτογραφίες, βίντεο ή ενσωματωμένες εφαρμογές. Στα πλαίσια της d3.js, κάθε στοιχείο στη σελίδα μπορεί να αντιμετωπιστεί με τις ίδιες αφαιρέσεις υψηλού επιπέδου. Το πλέον βασικό στοιχείο σε μία ιστοσελίδα, το <div> αντιπροσωπεύει ένα ορθογώνιο στο οποίο ο σχεδιαστής μπορεί να τοποθετήσει παραγράφους, λίστες και πίνακες, μπορεί να επιλεγεί και να τροποποιηθεί με τον ίδιο τρόπο που μπορεί να επιλεγεί και να τροποποιηθεί μία χώρα σε έναν διαδικτυακό χάρτη ή μεμονωμένοι κύκλοι και γραμμές που συνθέτουν ένα σύνθετο στοιχείο οπτικοποίησης (Meeks, 2017).

## Κεφάλαιο 6: Βιβλιοθήκη οπτικοποίησης Mapbox

### 6.1 Τι είναι η βιβλιοθήκη Mapbox

Παραδοσιακά, οι ψηφιακοί χάρτες αποδίδονται στον διακομιστή και αποστέλλονται ως εικόνες στο πελάτη (client), ωστόσο η νέα τεχνολογία web ανοίγει δυνατότητες για δυναμική απόδοση των χαρτών σε ένα πρόγραμμα περιήγησης ιστού. Η Mapbox είναι μία πλατφόρμα ανοιχτού κώδικα η οποία παρέχει εργαλεία ψηφιακής χαρτογράφησης που επιτρέπουν τη γρήγορη και εύκολη προσθήκη προσαρμοσμένων χαρτών σε εφαρμογές (Eriksson & Rydkvist, 2015). Είναι μία πλατφόρμα που χρησιμοποιείται από πολλές εταιρίες όπως οι Foursquare, Lonely Planet, Facebook, the Financial Times, The Weather Channel, Snapchat για την παροχή διαδικτυακών χαρτών και χαρτογραφικών εφαρμογών. Μέσα από αυτές τις εφαρμογές επιλύει προβλήματα που σχετίζονται με χάρτες, δεδομένα και χωρική ανάλυση (Eriksson & Rydkvist, 2015).

Η Mapbox GL JS είναι μία client-side (απόδοση από την πλευρά του χρήστη) βιβλιοθήκη Javascript για τη δημιουργία χαρτών web και εφαρμογών web χρησιμοποιώντας την τεχνολογία χαρτογράφησης της Mapbox. Όσον αφορά τη μορφή των δεδομένων που ο χρήστης μπορεί να εισάγει στη Mapbox, οι προσεγγίσεις είναι είτε μεταφορτώνοντας tilesets ή δημιουργώντας datasets. Αρχικά, τα tilesets είναι διανυσματικά ή κανονικοποιημένα δεδομένα που ο χρήστης μπορεί να επεξεργαστεί ως προς το στυλ τους, δηλαδή τα χρώματα, τη διαφάνεια, την ονοματολογία, τον τρόπο συμβολισμού. Δεν επιδέχονται άλλου είδους επεξεργασία όπως η αλλαγή γεωμετρίας και αυτή είναι η βασική τους διαφορά με τα datasets. Όταν τα δεδομένα που θέλει να μεταφορτώσει ο χρήστης εισέρχονται στη βάση δεδομένων υπολογιστικού νέφους της Mapbox μετατρέπονται αυτόματα σε tilesets με την κωδικοποίηση pbf (Protocolbuffer Binary Format), που είναι μία μορφή αρχείου προερχόμενη από τη γεωβάση SQLite. Η μεταφόρτωση tilesets υποστηρίζει διανυσματικά και κανονικοποιημένα δεδομένα δημιουργώντας raster και vector tilesets. Τα raster tilesets προέρχονται από raster τύπους αρχείων όπως GeoTIFF και φορτώνονται ως ψηφιακές εικόνες περιορίζοντας έτσι την επεξεργασία τους. Τα vector tilesets προέρχονται από πολλούς τύπους αρχείων, όπως GeoJSON, Shapefiles,

MBTILES, και δίνουν τη δυνατότητα στον χρήστη να χρησιμοποιήσει μεγάλο εύρος λειτουργιών για να τα φέρει στην επιθυμητή μορφή απόδοσης (Netek et al., 2020).

Το παρακάτω παράδειγμα παρουσιάζει τον ελάχιστο κώδικα που χρειάζεται προκειμένου να προστεθεί ένας χάρτης σε μία ιστοσελίδα:

```
mapboxgl.accessToken = ''; const map = new mapboxgl.Map({ container:
'map', // container ID style: 'mapbox://styles/mapbox/streets-v11', // style URL
center: [-74.5, 40], // starting position [lng, lat] zoom: 9 // starting zoom });
```

Οι χάρτες της Mapbox GL JS μπορούν να αποτελούνται από πολλά επίπεδα που παρέχουν οπτικά στοιχεία και δεδομένα χάρτη. Κάθε επίπεδο παρέχει κανόνες σχετικά με τον τρόπο με τον οποίο ο renderer πρέπει να σχεδιάσει ορισμένα δεδομένα στο πρόγραμμα περιήγησης και ο renderer χρησιμοποιεί αυτά τα επίπεδα για να σχεδιάσει τον χάρτη στην οθόνη.

## 6.2 Τομείς που χρησιμοποιείται η Mapbox

Η Mapbox GL JSON είναι μία καθαρή δομή JSON που χρησιμοποιείται για τη διαμόρφωση χαρτών στη Mapbox GL JS. Στην Εικόνα 16 περιγράφεται ένα στυλ χάρτη σε μορφή Mapbox GL JSON. Η ιδιότητα τύπου περιγράφει τον τύπο της γεωμετρίας που αποδίδεται για το συγκεκριμένο επίπεδο. Παρακάτω παρατίθενται οι τρεις σημαντικότεροι τύποι επιπέδων της συγκεκριμένης βιβλιοθήκης (Eriksson & Rydkvist, 2015):

- **Γέμισμα (Fill):** Κτίρια, πάρκα, λίμνες και άλλα πολυγωνικά αντικείμενα εντάσσονται στο συγκεκριμένο επίπεδο. Είναι περιοχές που μπορούν να γεμιστούν με ένα μεμονωμένο χρώμα ή μοτίβο χρωμάτων. Είναι επίσης δυνατό να δοθεί ένα περίγραμμα στα αντικείμενα γεμίσματος.
- **Γραμμή (Line):** Οι δρόμοι, τα σύνορα και λοιπά παρόμοια αντικείμενα εντάσσονται στο συγκεκριμένο επίπεδο. Ωστόσο, λόγω του περιορισμού επαναλήψεων των OpenGL's GL Lines καθώς και του γεγονότος του ότι η δυνατότητα antialiasing (αντιαλλοίωσης) του OpenGL δεν παρέχεται αξιόπιστα σε όλες τις συσκευές, το Mapbox GL αποδίδει τις γραμμές ως πολύγωνα.



- Σύμβολο (Symbol): Το συγκεκριμένο επίπεδο χρησιμοποιείται κατά την απόδοση κειμένων και εικονιδίων, όπως ονόματα πόλεων ή σημεία ενδιαφέροντος.

Εκτός από την ιδιότητα τύπου, τα επίπεδα έχουν δύο άλλες ιδιότητες που καθορίζουν το πως αποδίδονται τα δεδομένα από το κάθε επίπεδο: τη διάταξη (layout) και τη βαφή (paint).

Η διάταξη (layout) καθορίζει για παράδειγμα το που και πότε τοποθετούνται οι γεωμετρικές στον χάρτη καθώς και το πότε είναι ορατές ή όχι. Η αξία μίας ιδιότητας διάταξης υπολογίζεται και παρεμβάλλεται μία φορά ανά πλακίδιο (tile), σε κάθε ακέραιο επίπεδο ζουμ. Οι διατάξεις εφαρμόζονται νωρίς κατά τη διαδικασία απόδοσης. Καθορίζουν το πως μεταβιβάζονται τα δεδομένα στη GPU για το τρέχον επίπεδο (Eriksson & Rydkvist, 2015).

Με την κοινή χρήση των ιδιοτήτων διάταξης μεταξύ των επιπέδων, ο συνολικός χρόνος επεξεργασίας μπορεί να μειωθεί. Αυτό το γεγονός επιτρέπει επίσης σε πολλά επίπεδα να μοιράζονται τη μνήμη GPU και τους υπόλοιπους πόρους που σχετίζονται με τις διατάξεις (Eriksson & Rydkvist, 2015).

Οι ιδιότητες της βαφής (paint) χρησιμοποιούνται για να αλλάξουν την οπτική εμφάνιση του χάρτη. Χαρακτηριστικά παραδείγματα αποτελούν τα πλάτη γραμμών, τα χρώματα και τα μεγέθη κειμένου. Η αξία μίας ιδιότητας βαφής υπολογίζεται και παρεμβάλλεται σε κάθε πλαίσιο κατά την αλληλεπίδραση με τον χάρτη. Η τιμή της ιδιότητας βαφής μπορεί να είναι ένα αντικείμενο stop, το οποίο περιέχει πολλαπλά ζεύγη επιπέδου ζουμ και τιμών. Περιγράφει πως η συγκεκριμένη ιδιότητα βαφής αποδίδεται σε διαφορετικά επίπεδα ζουμ. Για παράδειγμα, το πλάτος μίας γραμμής μπορεί να αλλάξει δυναμικά κατά τη μεγέθυνση. Οι βαφές εφαρμόζονται σε μεταγενέστερο στάδιο της διαδικασίας απόδοσης. Επίπεδα που μοιράζονται ιδιότητες διατάξεων μπορούν να διατηρούν ανεξάρτητες ιδιότητες βαφής. Τα επίπεδα μπορεί να έχουν συγκεκριμένη κατηγορία ιδιοτήτων βαφής, καθιστώντας δυνατή την αλλαγή της εμφάνισης του χάρτη κατά έναν εύκολο τρόπο.

Τα επίπεδα ενδέχεται να περιέχουν τις προαιρετικές ιδιότητες του ελαχίστου και του μέγιστου ζουμ. Αυτές οι ιδιότητες περιγράφουν ανάμεσα σε ποια επίπεδα ζουμ θα πρέπει να αποδίδεται το επίπεδο. Αν το αντίστοιχο επίπεδο δεδομένων περιλαμβάνει δεδομένα σε περιττά επίπεδα ζουμ για το τρέχον στυλ, τότε πρέπει να

χρησιμοποιούνται ελάχιστα και μέγιστα ζουμ. Αυτός είναι ένας τρόπος για να ελαχιστοποιηθεί ο αριθμός των επιπέδων υπό επεξεργασία κατά τη φόρτωση ενός πλακιδίου.

```
1  "layers": [  
2    {  
3      "id": "school",  
4      "type": "fill",  
5      "source-layer": "building",  
6      "paint": {  
7        "fill-color": "#f2eff9"  
8      }  
9    },  
10   {  
11     "id": "school",  
12     "type": "line",  
13     "source-layer": "building",  
14     "paint": {  
15       "line-color": "#ff0000",  
16       "line-width": 0.1  
17     }  
18   }  
19 ]
```

Εικόνα 16. Mapbox JSON. Πηγή: Eriksson & Rydkvist, 2015.

### 6.3 Βασικά στοιχεία και χαρακτηριστικά της βιβλιοθήκης Mapbox

Τα χαρτογραφικά δεδομένα της Mapbox αποτελούνται από έτοιμα και μοναδικά σύνολα χαρτογραφικών δεδομένων σε μορφή tilesets (χαρτογραφικών πινακίδων) που προέρχονται από διάφορες πηγές και ανήκουν στην πλατφόρμα. Περιλαμβάνουν επίπεδα δρόμων, αναγλύφου, κυκλοφορίας, ακόμα και δορυφορικές εικόνες (Mapbox Streets, Mapbox Terrain, Mapbox Traffic, Mapbox Satellite).

Η χαρτογραφική σχεδίαση παρέχει τη δυνατότητα στον χρήστη να σχεδιάζει τον χάρτη μέσω της λειτουργίας του Mapbox Studio. Η εφαρμογή του Mapbox Studio style editor είναι το εργαλείο της σχεδιαστικής έναρξης δίνοντας την αρχική επιλογή από μία μεγάλη ποικιλία προκαθορισμένων στυλ που στη συνέχεια ο χρήστης μπορεί

να εμπλουτίσει και να τροποποιήσει. Ο σχεδιασμός γίνεται μέσα από την ιστοσελίδα της Mapbox και συγκεκριμένα στο Mapbox Studio style editor μέσω κάποιου web browser και περιλαμβάνει δυνατότητες αλλαγής χρωμάτων, ονοματολογίας, φιλτραρίσματα των προκαθορισμένων επιπέδων πληροφορίας κ.α.

Η προσθήκη δεδομένων από τον χρήστη αποτελεί ακόμα ένα χαρακτηριστικό της βιβλιοθήκης. Εκτός των προκαθορισμένων Mapbox δεδομένων, ο χρήστης μπορεί να εισάγει τα δικά του χαρτογραφικά δεδομένα με δύο τρόπους, με τη μεταφόρτωση tilesets και τη δημιουργία datasets. Η μορφή που παίρνουν τα δεδομένα μπορεί να φορτωθεί στο Mapbox Studio style editor και να δημιουργηθεί ένας χάρτης σε αυτόν.

Η δημιουργία εφαρμογής αποτελεί ένα επιπρόσθετο χαρακτηριστικό της βιβλιοθήκης. Μετά το πέρας των διαδικασιών μεταφόρτωσης, επεξεργασίας και styling των δεδομένων και του χάρτη παρέχεται η δυνατότητα χρήσης κάποιας JavaScript βιβλιοθήκης ώστε να δημοσιοποιηθεί ο χάρτης σε κάποια εφαρμογή. Οι εφαρμογές που υποστηρίζει τη Mapbox είναι διαδικτυακές εφαρμογές (web applications), εφαρμογές κινητών τηλεφώνων (mobile applications) αλλά και εφαρμογές unity (unity applications).

Η επέκταση της εφαρμογής συνοψίζεται στη δυνατότητα που δίδεται μετά και τη δημοσιοποίηση του χάρτη να εμπλουτισθούν οι λειτουργίες του με διαδραστικότητα και δυναμικότητα μέσω εργαλείων (web services APIs) που παρέχουν αλληλεπίδραση με τα δεδομένα, εύρεση διευθύνσεων, υλοποίηση χωρικής ανάλυσης και εύρεση διαδρομής.

Τελευταία δυνατότητα είναι η εξαγωγή στατικού χάρτη μετά και την επέκταση της εφαρμογής πάλι με τη βοήθεια εργαλείων της Mapbox (Mapbox Static Images API) (Mapbox, 2022<sup>a</sup>).

## 6.4 Εγκατάσταση της Mapbox

Πριν την έναρξη της εγκατάστασης της βιβλιοθήκης Mapbox GL JS, πρέπει να υπάρχει ένα διακριτικό πρόσβασης στη Mapbox και να προστεθεί η Mapbox GL JS στο έργο προς υλοποίηση χρησιμοποιώντας είτε το πακέτο CDN είτε το πακέτο `mapbox-gl npm` (Mapbox, 2022<sup>b</sup>).

Η χρησιμοποίηση της βιβλιοθήκης Mapbox GL JS απαιτεί την κατοχή από τον χρήστη ενός διακριτικού πρόσβασης (token) στη Mapbox, το οποίο συνδέεται με τη δημιουργία ενός λογαριασμού στην εκδότρια εταιρεία. Αυτό το διακριτικό πρόσβασης συσχετίζει τον χάρτη του χρήστη με έναν λογαριασμό Mapbox.

Προκειμένου να χρησιμοποιηθεί η βιβλιοθήκη Mapbox GL JS σε ένα έργο, πρέπει να εισαχθεί με τη χρήση της Mapbox GL JS CDN ή να εγκατασταθεί το πακέτο `mapbox-gl npm`.

Τα αρχεία JavaScript και CSS πρέπει να συμπεριληφθούν στο `<head>` του αρχείου HTML του χρήστη. Το αρχείο CSS απαιτείται για να εμφανιστεί ο χάρτης και να λειτουργήσουν στοιχεία όπως τα αναδυόμενα παράθυρα και οι δείκτες. Η εργασία αυτή λαμβάνει χώρα με τον παρακάτω κώδικα (Mapbox, 2022<sup>b</sup>):

```
<script src='https://api.mapbox.com/mapbox-gl-js/v2.9.2/mapbox-gl.js'>
</script>
<link href='https://api.mapbox.com/mapbox-gl-js/v2.9.2/mapbox-gl.css'
rel='stylesheet' />
```

Ο παρακάτω κώδικας πρέπει να συμπεριληφθεί στο <body> του κώδικα HTML του χρήστη (Mapbox, 2022<sup>b</sup>) :

```
<div id='map' style='width: 400px; height: 300px;'></div>  
<script>  
// TO MAKE THE MAP APPEAR YOU MUST  
// ADD YOUR ACCESS TOKEN FROM  
// https://account.mapbox.com  
mapboxgl.accessToken = '<your access token here>';  
const map = new mapboxgl.Map({  
  container: 'map', // container ID  
  style: 'mapbox://styles/mapbox/streets-v11', // style URL  
  center: [-74.5, 40], // starting position [lng, lat]  
  zoom: 9, // starting zoom  
  projection: 'globe' // display the map as a 3D globe  
});  
map.on('style.load', () => {  
  map.setFog({}); // Set the default atmosphere style  
});  
</script>
```

Η βιβλιοθήκη Mapbox GL JS v2 διανέμεται ως πακέτο JavaScript συμβατό με ES6 και είναι συμβατό με όλα τα κύρια σύγχρονα προγράμματα περιήγησης.

Το πακέτο JavaScript δεν είναι συμβατό με ορισμένους μετασχηματισμούς Babel λόγω του τρόπου με τον οποίο μοιράζεται τον κώδικα μεταξύ του κύριου νήματος και του Web Worker. Αυτό το χαρακτηριστικό λαμβάνει χώρα με γνώμονα τη μείωση του μεγέθους του πακέτου και τη βελτίωση της απόδοσης. Εάν ο

σχεδιαστής χρησιμοποιεί το v2 με ένα bundler λειτουργιών όπως το Webpack ή το Rollup μαζί με ένα transpiler όπως το Babel, υπάρχουν τρεις τρόποι προκειμένου να επιτευχθεί η συμβατότητα:

- Χρήση της εντολής browserlist για τη στόχευση της μετάβασης σε ένα σύνολο συμβατών μετασχηματισμών.
- Ολική απενεργοποίηση της μετάβασης του πακέτου Marbox GL JS.
- Φόρτωση και μεταφορά του κώδικα Web Worker ξεχωριστά με κόστος την αύξηση του μεγέθους του πακέτου και τη μείωση της απόδοσης.

Για την περίπτωση κατά την οποία ο χρήστης χρησιμοποιεί το @babel/preset-env σε συνδυασμό με το πρόγραμμα περιήγησης για τον ορισμό περιβαλλόντων προγράμματος περιήγησης, μπορεί να εξετάσει το ενδεχόμενο να χρησιμοποιήσει τα ακόλουθα ερωτήματα λίστας προγραμμάτων περιήγησης για να επιλέξει ένα σύνολο συμβατών μετασχηματισμών (Marbox, 2022<sup>b</sup>):

**>0.2%, not dead, not ie 11, not chrome < 51, not safari < 10, not android < 51**

ή

**defaults, not ie 11**

Τα παραπάνω μπορούν να καθοριστούν στο πακέτο.json του έργου ή σε ένα αρχείο .browserslistrc.

Για την περίπτωση κατά την οποία άλλα μέρη της αίτησής χρειάζονται τη μεταφορά ES5, τότε πρέπει να εξεταστεί το ενδεχόμενο ρητής εξαίρεσης του GL JS από τη μετάδοση. Εάν χρησιμοποιείται το Webpack, μπορεί να χρησιμοποιηθεί το πρόθεμα ! στη δήλωση εισαγωγής για τον αποκλεισμό της μετατροπής της marbox-gl από τους υπάρχοντες φορτωτές. Η διαδικασία λαμβάνει χώρα με τους παρακάτω κώδικες (Marbox, 2022<sup>b</sup>):

```
import mapboxgl from '!mapbox-gl';  
ή  
use: {  
  loader: 'babel-loader',  
  options: {  
    presets: ['my-custom-babel-preset'],  
    ..,  
    ..,  
    ignore: [ './node_modules/mapbox-gl/dist/mapbox-gl.js' ]  
  }  
}}
```

Εάν η εφαρμογή που υλοποιείται απαιτεί συμβατότητα ES5, τότε η δέσμη μονάδων πρέπει να διαμορφωθεί ώστε να φορτώνει και να μεταφέρει ξεχωριστά το Web Worker της Mapbox GL JS. Αυτό επιφέρει το κόστος της σημαντικής αύξησης του μεγέθους του πακέτου και του αρνητικού αντίκτυπου στην απόδοση και θα πρέπει να πραγματοποιείται μόνο εάν υπάρχει έντονη ανάγκη υποστήριξης προγραμμάτων περιήγησης παλαιού τύπου. Η Mapbox GL JS μπορεί να διαμορφωθεί με πρόσθετα worker-loader ειδικά για το bundler.

Εάν χρησιμοποιείται το Webpack, μπορεί να διαμορφωθεί το worker-loader ώστε να χρησιμοποιείται inline κατά την εισαγωγή της mapbox-gl (Mapbox, 2022<sup>b</sup>):

```
import mapboxgl from 'mapbox-gl/dist/mapbox-gl-csp';

import MapboxWorker from 'worker-loader!mapbox-gl/dist/mapbox-gl-csp-
worker'; // Load worker code separately with worker-loader

mapboxgl.workerClass = MapboxWorker; // Wire up loaded worker to be used
instead of the default

const map = new mapboxgl.Map({

  container: 'map', // container ID

  style: 'mapbox://styles/mapbox/streets-v11', // style URL

  center: [-74.5, 40], // starting position [lng, lat]

  zoom: 9, // starting zoom

  projection: 'globe' // display the map as a 3D globe

});

map.on('style.load', () => {

  map.setFog({}); // Set the default atmosphere style

});
```

Εναλλακτικά, μπορεί επίσης να ρυθμιστεί το worker-loader κεντρικά στο webpack.config.js ως εξής (Mapbox, 2022<sup>b</sup>):



```
module.exports = {  
  module: {  
    rules: [  
      {  
        test: /\.bmapbox-gl-csp-worker.js\b/i,  
        use: { loader: 'worker-loader' }  
      }  
    ]  
  }  
};
```

και, στη συνέχεια, να ενσωματωθεί το πρόγραμμα εργασίας που έχει φορτωθεί στο Webpack με τη Mapbox GL JS (Mapbox, 2022<sup>b</sup>):

```
import mapboxgl from 'mapbox-gl/dist/mapbox-gl';  
import MapboxWorker from 'mapbox-gl/dist/mapbox-gl-csp-worker';  
mapboxgl.workerClass = MapboxWorker;  
const map = new mapboxgl.Map({  
  container: 'map', // container ID  
  style: 'mapbox://styles/mapbox/streets-v11', // style URL  
  center: [-74.5, 40], // starting position [lng, lat]  
  zoom: 9, // starting zoom  
  projection: 'globe' // display the map as a 3D globe});  
map.on('style.load', () => {  
  map.setFog({}); // Set the default atmosphere style});
```

## 6.5 Εφαρμογές της Mapbox

Η αλληλεπίδραση με έναν χάρτη που αποδίδεται με τη χρήση της βιβλιοθήκης Mapbox GL JS, δίνει την εντύπωση μετατόπισης σε μια μεγάλη, συνεχή εικόνα. Καθώς μία εικόνα όλου του κόσμου σε επίπεδο δρόμων και οδών θα ήταν

πολύ μεγάλη για λήψη ή διατήρηση στη μνήμη, ο χάρτης αποτελείται από πολλαπλές μικρές τετράγωνα εικόνες που ονομάζονται πλακίδια (tiles). Τοποθετώντας τα πλακίδια σε διπλανές θέσεις, δημιουργείται μία ψευδαίσθηση μίας μεγάλης συνεχούς εικόνας. Είναι δυνατή η προβολή γεωγραφικών στοιχείων από διαφορετικές περιοχές στον ίδιο συνεχόμενο χώρο μετατοπίζοντας τον χάρτη. Αντί ο χρήστης να απαιτείται να πλοηγείτε μεταξύ χαρτών με διαφορετική λεπτομέρεια και επίπεδο ζουμ, μπορεί να πλοηγηθεί με ένα ενιαίο συνεχές σύστημα. Με τη χρήση της μεγέθυνσης και της σμίκρυνσης σε αυτό το σύστημα, το μέγεθος των λεπτομερειών στον κάθε χάρτη ποικίλλει. Τα υψηλότερα επίπεδα ζουμ γενικά οδηγούν σε περισσότερο λεπτομερείς χάρτες (Eriksson & Rydkvist, 2015).

Μία θέση πλακιδίων στον πλανήτη περιγράφεται από ένα τρισδιάστατο σύστημα συντεταγμένων. Κάθε πλακίδιο έχει μία συντεταγμένη  $z$ ,  $x$  και  $y$ . Η συντεταγμένη  $z$  περιγράφει το επίπεδο ζουμ των πλακιδίων, ενώ οι συντεταγμένες  $x$  και  $y$  περιγράφουν τη θέση τους μέσα σε ένα τετράγωνο πλέγμα για το επίπεδο ζουμ  $z$ . Το επίπεδο ζουμ 0 αποτελείται από μόνο ένα πλακίδιο που καλύπτει ολόκληρη την υδρόγειο. Αυτό το πλακίδιο έχει το χαμηλότερο επίπεδο λεπτομέρειας. Όταν αυξάνεται το επίπεδο ζουμ κατά ένα, το καθένα πλακίδιο χωρίζεται σε τέσσερα νέα πλακίδια.

Η εκθετική σχέση μεταξύ της συντεταγμένης  $z$  και του αριθμού των πλακιδίων στην πυραμίδα του συστήματος συντεταγμένων, καθώς και η ποσότητα των πλακιδίων που απαιτούνται για τη δημιουργία του ίδιου χάρτη μίας γεωγραφικής περιοχής γίνεται μεγάλη σε υψηλότερα επίπεδα ζουμ. Εξυπηρετώντας μόνο τα πλακίδια που απαιτούνται για την εμφάνιση του χάρτη της επιλεγμένης γεωγραφικής τοποθεσίας η ποσότητα των πλακιδίων γίνεται διαχειρίσιμη από την πλευρά της αποθήκευσης και του εύρους ζώνης. Μόλις γίνει λήψη ενός πλακιδίου, το πρόγραμμα περιήγησης μπορεί να το αποθηκεύσει στη μνήμη cache του και να το χρησιμοποιήσει ξανά όταν το πλακίδιο αυτό χρειαστεί. Με τη φόρτωση των πλακιδίων σταδιακά, ξεκινώντας από το κέντρο της οθόνης, ο χρήστης μπορεί να μετατοπίσει ή να μεγεθύνει σε μία συγκεκριμένη τοποθεσία, ακόμα και αν τα πλακίδια στα άκρα της οθόνης δεν έχουν φορτωθεί.

Τα πρωτόκολλα προσωρινής μνήμης (buffer) είναι ένας τρόπος σειριοποίησης δομημένων δεδομένων και συντηρούνται από τη Google. Είναι μία ουδέτερη γλώσσα και πλατφόρμα, πράγμα που σημαίνει ότι το πρωτόκολλο αποθηκεύει προσωρινά,

μπορεί να χρησιμοποιηθεί σε οποιαδήποτε πλατφόρμα και σε οποιαδήποτε γλώσσα. Ωστόσο, για να σειριοποιηθούν και να αποσειριοποιηθούν τα δεδομένα πρέπει να υπάρχει συγκεκριμένη εφαρμογή για αυτό το περιβάλλον, αλλά και τα δεδομένα που αποστέλλονται να είναι συνεπή. Ένα από τα κύρια χαρακτηριστικά των πρωτοκόλλων αυτών είναι ότι η δομή των δεδομένων πρέπει να καθοριστεί σε ένα σχήμα εκ των προτέρων, και τόσο ο αποστολέας όσο και ο δέκτης πρέπει να έχουν το συγκεκριμένο σχήμα. Ωστόσο, είναι δυνατή η αλλαγή του σχήματος και των δεδομένων από την πλευρά του αποστολέα, ενώ αργότερα όταν ο παραλήπτης αναλύει τα εισερχόμενα δεδομένα, παραλείπονται εκείνα τα χαρακτηριστικά που δεν γνωρίζει.

Τα πρωτοκόλλα buffer συγκρίνονται συχνά με τη JSON ή την XML. Και το JSON και το XML είναι αναγνώσιμα και αναλυτικά από τον άνθρωπο, ενώ τα buffer πρωτοκόλλου είναι μη αναγνώσιμα από τον άνθρωπο, σε δυαδική μορφή και καλύτερα σχεδιασμένα για μηχανική ανάγνωση. Αυτό έχει ως αποτέλεσμα τα αρχεία του πρωτοκόλλου buffer να είναι 3-10 φορές μικρότερα και να έχουν 20-100 φορές ταχύτερη ανάλυση.

Χαρακτηριστικό παράδειγμα αποτελεί ο παρακάτω κώδικας:

```
<person>
<name>John Doe</name>
<email>jdoe@example.com</email>
</person>
```

Αυτό το μικρό απόσπασμα XML είναι τουλάχιστον 69 byte, εάν αφαιρεθούν τα κενά και χρειάζονται 5.000-10.000 ns για να ολοκληρωθεί η ανάλυση. Το ισοδύναμο αρχείο buffer πρωτοκόλλου με τις ίδιες πληροφορίες θα ήταν περίπου 28 byte σε μέγεθος και θα χρειάζονταν 100-200 ns προκειμένου να αναλυθεί.

Το σχήμα που χρησιμοποιείται για τη σειριοποίηση και την ανάλυση αρχείων buffer πρωτοκόλλου ορίζεται στα αρχεία επέκτασης .proto και ονομάζεται μήνυμα. Κάθε μήνυμα πρωτοκόλλου buffer περιέχει μία σειρά από ζεύγη ονόματος-τιμής που περιγράφουν το σύνολο δεδομένων.

Έστω το παρακάτω υπόδειγμα κώδικα:

```
message Person {  
  required string name = 1 ;  
  optional boolean alive = 2 ;  
  repeated PhoneNumber phone = 3 ;  
  message PhoneNumber {  
    required string number = 1 ;  
  }  
}
```

Αποτελεί είναι ένα ελάχιστο παράδειγμα ενός τύπου μηνύματος Person, όπου κάθε πεδίο έχει ένα όνομα, έναν τύπο, έναν δείκτη σχετικά με το εάν είναι απαραίτητο, όχι απαραίτητο, επαναλαμβανόμενο και μοναδικό αναγνωριστικό. Το αναγνωριστικό χρησιμοποιείται για την αναγνώριση πεδίων όταν είναι σε δυαδική μορφή και δεν πρέπει να αλλάζει αφού χρησιμοποιηθεί ο τύπος μηνύματος. Η μορφή μηνύματος υποστηρίζει μία ιεραρχική δομή όπως φαίνεται στο παράδειγμα όπου πληκτρολογείται το μήνυμα PhoneNumber το οποίο ορίζεται μέσα στο πεδίο Person και ένα άτομο μπορεί να έχει μία λίστα με τους αριθμούς τηλεφώνου χάρη στη λέξη-κλειδί “repeated”.

## Κεφάλαιο 7: Σύγκριση βιβλιοθηκών οπτικοποίησης

### 7.1 Ποσότητα κώδικα

Παρακάτω παρουσιάζεται ένα παράδειγμα που αφορά τη σχεδίαση ενός χάρτη με τη χρήση της βιβλιοθήκης Mapbox (Frerichs, 2019):

```
var url =
  "https://gist.githubusercontent.com/milafreerichs/78ef5702db2dc514fc2
  bed465d58406b/raw/f1366ee2a83a9afb1dd2427e9cbd4cd3db8d87ca/bundeslae
  nder_simplif
  y200.geojson";
mapboxgl.accessToken = 'YOUR MAPBOX TOKEN NEEDS TO BE
  PLACED HERE';
var map = new mapboxgl.Map({
  container: 'map',
  style: 'mapbox://styles/mapbox/streets-v11',
  center: [13.79,53.545],
  zoom: 5
});
map.on('load', function () {
  map.addSource('bb', { type: 'geojson', data: url });
  map.addLayer({
    'id': 'berlin',
    'type': 'fill',
    'source': 'bb',
    'paint': {
      'fill-color': '#088',
      'fill-opacity': 0.8
    }
  });
  map.addLayer({
    'id': 'berlin-stroke',
    'type': 'line',
    'source': 'bb',
    'paint': {
      'line-width': 1,
      'line-color': '#000',
    }
  });
});
```

Ο κώδικας για την υλοποίηση του ίδιου χάρτη με τη χρήση της βιβλιοθήκης d3 παρατίθεται παρακάτω (Frerichs, 2019):

```
var url =
  "https://gist.githubusercontent.com/milafreerichs/78ef5702db2dc514fc2
  bed465d58406b/raw/f1366ee2a83a9afb1dd2427e9cbd4cd3db8d87ca
  /bundeslaender_simplify200
  .geojson";
d3.json(url).then(function(bb) {
  var bbox = d3.select("#d3-map").node().getBoundingClientRect();
  var width = bbox.width;
  var height = bbox.height;
  var projection = d3.geoEqualEarth();
  projection.fitExtent([[20, 20], [width, height]], bb);
  var geoGenerator = d3.geoPath().projection(projection);
  var svg = d3.select("#d3-map").append('svg')
  .style("width", "100%")
  .style("height", "100%");
  svg.append('g').selectAll('path')
  .data(bb.features)
  .enter()
  .append('path')
  .attr('d', geoGenerator)
  .attr('fill', '#088')
  .attr('stroke', '#000');
```

Η πρώτη διαφορά που εντοπίζεται είναι ότι ο κώδικας για την υλοποίηση του χάρτη με τη χρήση της βιβλιοθήκης Mapbox είναι σχεδόν διπλάσιος από τον κωδικά που απαιτεί η βιβλιοθήκη d3. Το γεγονός αυτό οφείλεται στο ότι το στυλ της Mapbox απαιτεί περισσότερη προσπάθεια και είναι διαμορφωμένο κατά έναν ευανάγνωστο τρόπο. Από τα παραπάνω μπορεί να εξαχθεί το συμπέρασμα του ότι η βιβλιοθήκη d3 υπερτερεί όταν λαμβάνεται ως κριτήριο σύγκρισης η ποσότητα του κώδικα που απαιτείται για την υλοποίηση μίας εφαρμογής.

Η βιβλιοθήκη οπτικοποίησης δεδομένων Pygal απαιτεί μικρή ποσότητα κώδικα για την υλοποίηση των διαφόρων εφαρμογών, χάρη στα διάφορα modules που διαθέτει. Τα γραφικά SVG που δημιουργούνται μέσω της Pygal βασίζονται σε διανύσματα XML τα οποία μπορούν να επεξεργαστούν σε οποιονδήποτε editor. Χαρακτηριστικό τους είναι ότι μπορούν να δημιουργηθούν με ελάχιστες γραμμές κώδικα οι οποίες είναι εύκολα κατανοητές και συντάξιμες. Για παράδειγμα, η υλοποίηση ενός χάρτη (μέσω του module pygal.maps) λαμβάνει χώρα με τον εξής κώδικα (AnyCodings, 2022):

```

import pygal

from pygal.maps.world import World

worldmap_chart = pygal.maps.world.World()

worldmap_chart.title = 'Some countries'

worldmap_chart.add('F countries', ['fr', 'fi'])

worldmap_chart.add('M countries', ['ma', 'mc', 'md', 'me', 'mg',
                                     'mk', 'ml', 'mm', 'mn', 'mo',
                                     'mr', 'mt', 'mu', 'mv', 'mw',
                                     'mx', 'my', 'mz'])

worldmap_chart.add('U countries', ['ua', 'ug', 'us', 'uy', 'uz'])

worldmap_chart.render()

```

Συγκρίνοντας τον κώδικα αυτόν με τους αντίστοιχους των βιβλιοθηκών που βασίζονται σε Java, είναι εμφανές ότι η βιβλιοθήκη Pygal απαιτεί τη λιγότερη ποσότητα κώδικα από τις τρεις εξεταζόμενες.

## 7.2 Ευκολία εγκατάστασης

Και οι δύο βιβλιοθήκες (Mapbox, d3) είναι σχετικά εύκολο να εγκατασταθούν. Υπάρχουν δύο επιλογές. Στα πλαίσια της πρώτης επιλογής, ο χρήστης έχει τη δυνατότητα να μεταφορτώσει τον κώδικα τους και να τον ενσωματώσει. Στα πλαίσια της δεύτερης επιλογής, μπορεί να χρησιμοποιήσει ένα CDN, κάτι το οποίο έλαβε χώρα και στα παραδείγματα που παρατέθηκαν στην παράγραφο 6.1.

Προκειμένου να εγκατασταθεί η βιβλιοθήκη οπτικοποίησης δεδομένων Mapbox πρέπει να πληκτρολογηθεί ο εξής κώδικας (Frerichs, 2019):

```

<script src="https://unpkg.com/mapbox-gl@0.49.0"
type="text/javascript"></script>
<link href="https://unpkg.com/mapbox-gl@0.49.0/dist/mapbox-gl.css"
rel="stylesheet" type="text/css">

```

Ο χρήστης πρέπει να συμπεριλάβει τον κώδικα της βιβλιοθήκης JavaScript και το CSS για το Marbox. Το φύλλο των στυλ είναι ιδιαίτερος σημαντικό προκειμένου να βεβαιωθεί ότι τα στοιχεία πλοήγησης είναι ορατά. Επιπλέον, ο χρήστης πρέπει να εγγραφεί στον ιστότοπο `marbox.com` προκειμένου να λάβει ένα διακριτικό API για να χρησιμοποιήσει τους χάρτες βάσης. Η συγκεκριμένη εγγραφή είναι δωρεάν και απαιτείται μόνο το e-mail.

Προκειμένου να εγκατασταθεί η βιβλιοθήκη οπτικοποίησης δεδομένων d3 πρέπει να πληκτρολογηθεί ο εξής κώδικας (Frerichs, 2019):

```
<script src="https://unpkg.com/d3@5.9.7/dist/d3.min.js"
type="text/javascript"></script>
```

Χρειάζεται απλώς ο χρήστης να συμπεριλάβει τη βιβλιοθήκη. Επιπρόσθετα, πρέπει να βεβαιωθεί ότι χρησιμοποιεί την πιο πρόσφατη έκδοση. Από τα παραπάνω μπορεί να εξαχθεί το συμπέρασμα του ότι η βιβλιοθήκη d3 υπερτερεί όταν λαμβάνεται ως κριτήριο σύγκρισης η ευκολία της εγκατάστασης.

Η βιβλιοθήκη οπτικοποίησης δεδομένων Pygal επίσης έχει ιδιαίτερος εύκολη στην εγκατάσταση. Η εγκατάσταση λαμβάνει χώρα μέσω μίας απλής εντολής `pip install`, με τον παρακάτω κώδικα (Programmer, 2021):

### **pip install pygal**

Εναλλακτικά, εάν οι εγχώριοι μικροί συνεργάτες θεωρούν ότι η ταχύτητα λήψης είναι χαμηλή, μπορούν επίσης να επιταχύνουν προσθέτοντας μια εικόνα τρίτου μέρους (Programmer, 2021):

### **pip install -i <http://pypi.douban.com/simple/> pygal**

Από τα παραπάνω μπορεί να εξαχθεί το συμπέρασμα του ότι η βιβλιοθήκη Pygal εγκαθίσταται περισσότερο εύκολα σε σχέση με εκείνες που βασίζονται σε Java.

## **7.3 Styling**

Το styling στη βιβλιοθήκη οπτικοποίησης δεδομένων Marbox λαμβάνει χώρα μέσω του Marbox Style Spec, ενώ στη βιβλιοθήκη d3 μέσω κανονικών χαρακτηριστικών SVG και CSS. Η διαμόρφωση του γεμίματος και του περιγράμματος των χαρακτηριστικών στη βιβλιοθήκη Marbox απαιτεί την υλοποίηση δύο επιπέδων. Αυτός είναι ένας από τους πολλούς περιορισμούς του ορισμού στυλ.



Από την άλλη πλευρά, ο σχεδιαστής έχει τη δυνατότητα πραγματοποίησης πληθώρας εργασιών με τη βιβλιοθήκη style spec, επομένως η εφαρμογή είναι μία πτυχή που δεν δημιουργεί σοβαρά ζητήματα.

Στα πλαίσια της βιβλιοθήκης οπτικοποίησης δεδομένων d3 μέσω των δυνατοτήτων που παρέχουν το SVG και το CSS ο σχεδιαστής μπορεί να διαμορφώσει σχεδόν οτιδήποτε. Επιπρόσθετα, η συγκεκριμένη δυνατότητα είναι περισσότερο βολική για τον σχεδιαστή.

Και στις δύο περιπτώσεις, ο σχεδιαστής πρέπει να μάθει το πως να διαμορφώνει τα στοιχεία.

Παρακάτω παρατίθεται ένα παράδειγμα styling με τη χρήση της βιβλιοθήκης οπτικοποίησης δεδομένων Mapbox (Frerichs, 2019):

```
map.addLayer({
  'id': 'berlin',
  'type': 'fill',
  'source': 'bb',
  'paint': {
    'fill-color': '#088',
    'fill-opacity': 0.8
  }
});
map.addLayer({
  'id': 'berlin-stroke',
  'type': 'line',
  'source': 'bb',
  'paint': {
    'line-width': 1,
    'line-color': '#000'}});
```

Παρακάτω παρατίθεται ένα παράδειγμα styling με τη χρήση της βιβλιοθήκης οπτικοποίησης δεδομένων d3 (Frerichs, 2019):

```
.attr('fill', '#088')  
.attr('stroke', '#000');
```

Σύμφωνα με την παραπάνω ανάλυση μπορεί να εξαχθεί το συμπέρασμα του ότι η βιβλιοθήκη d3 υπερτερεί όταν λαμβάνεται ως κριτήριο σύγκρισης το styling.

Η Pygal υποστηρίζει μόνο τα χαρακτηριστικά SVG. Παρέχει 14 στυλ μέσω της κλάσης pygal.style, τα οποία είναι τα παρακάτω (Geeks for Geeks, 2020):

- DarkStyle
- NeonStyle
- DarkSolarizedStyle
- LightSolarizedStyle
- LightStyle
- CleanStyle
- RedBlueStyle
- DarkColorizedStyle
- LightColorizedStyle
- TurquoiseStyle
- LightGreenStyle
- DarkGreenStyle
- DarkGreenBlueStyle
- BlueStyle

Τα παραπάνω στυλ χρησιμοποιούνται για τη σχεδίαση οποιουδήποτε γραφήματος. Ένα παράδειγμα styling με τη χρήση της βιβλιοθήκης Pygal παρατίθεται παρακάτω (Geeks for Geeks, 2020):

```
# importing pygal  
import pygal  
from pygal.style import NeonStyle  
# creating the chart object  
funnel = pygal.Funnel(style = NeonStyle)  
# naming the title  
funnel.title = 'Funnel'  
# Random data  
funnel.add('A', [26, 22, 39, 39, 32, 30, 33, 24, 24, 30])  
funnel.add('B', [31, 40, None, None, None, None, 40, 32, 25, 31])  
funnel.add('C', [37, 27, 31, 20, None, 32, 24, 39, 29, 22])  
funnel.add('D', [38, None, 20, 29, 33, 23, 32, 33, 32, 23])  
funnel
```

Λαμβάνοντας υπόψη το γεγονός του ότι η βιβλιοθήκη Pygal υποστηρίζει μόνο τα χαρακτηριστικά SVG, εν αντιθέσει με τις βιβλιοθήκες που βασίζονται σε Java, οι οποίες υποστηρίζουν και τα χαρακτηριστικά CSS, καθώς και το γεγονός του ότι διατίθενται λιγότερα στυλ σε σχέση με τις άλλες βιβλιοθήκες, μπορεί να εξαχθεί το συμπέρασμα του ότι η Pygal υστερεί αναφορικά με τις δυνατότητες styling που παρέχει στον σχεδιαστή. Από την άλλη πλευρά, παρουσιάζει το πλεονέκτημα του ότι η εκμάθηση της απαιτεί λιγότερο χρόνο και η χρήση της είναι αρκετά ευκολότερη.

## 7.4 Basemap

Η μεγαλύτερη ορατή διαφορά αναφορικά με την οπτικοποίηση σε μορφή χάρτη είναι η έλλειψη βασικού χάρτη στη βιβλιοθήκη οπτικοποίησης δεδομένων d3. Βέβαια, το χαρακτηριστικό αυτό προέκυψε από επιλογή. Ένα από τα σημαντικότερα μειονεκτήματα της βιβλιοθήκης οπτικοποίησης δεδομένων d3 είναι ότι δεν περιλαμβάνει βασικό χάρτη, καθώς με τον τρόπο αυτό τα δεδομένα είναι το επίκεντρο του χάρτη και όχι ο βασικός χάρτης.

Το χαρακτηριστικό αυτό αποτελεί ένα από τα πλέον σημαντικά στοιχεία. Ο σχεδιαστής επιλέγει τις βιβλιοθήκες με βάση τις ανάγκες του και η d3 και η Mapbox εξυπηρετούν διαφορετικές ανάγκες. Η Mapbox και άλλες βιβλιοθήκες που βασίζονται σε χάρτες βάσης πλεονεκτούν όταν είναι απαραίτητες περισσότερες πληροφορίες από τα δεδομένα, όπως για παράδειγμα τοποθεσίες δρόμων, περισσότερες ετικέτες κλπ.

Λαμβάνοντας υπόψη τις περισσότερες δυνατότητες που παρέχει η βιβλιοθήκη Mapbox, μπορεί να θεωρηθεί ότι υπερτερεί της d3 με βάση το κριτήριο του βασικού χάρτη (basemap).

Η βιβλιοθήκη οπτικοποίησης δεδομένων Pygal δεν περιέχει κάποιον βασικό χάρτη, αλλά αντ'αυτού χρησιμοποιεί μία βασική κλάση για τη δημιουργία της σχετικής επέκτασης. Μέσω της κλάσης αυτής προκύπτουν αντίστοιχες δυνατότητες με εκείνες της βιβλιοθήκης Mapbox, η οποία χρησιμοποιεί βασικό χάρτη (Pygal, 2016<sup>a</sup>).

Συνοψίζοντας τα παραπάνω, προκύπτει το συμπέρασμα του ότι οι βιβλιοθήκες Mapbox και Pygal υπερτερούν της d3 με βάση το κριτήριο του βασικού χάρτη.

## 7.5 Projections – Οπτικοποίηση σε μορφή χάρτη

Η δεύτερη μεγάλη διαφορά ανάμεσα στις δύο βιβλιοθήκες είναι οι προβολές του χάρτη (projections). Η βιβλιοθήκη οπτικοποίησης δεδομένων Mapbox χρησιμοποιεί το Mercator ως προεπιλογή και ο σχεδιαστής δεν έχει τη δυνατότητα να αλλάξει την προβολή. Στη βιβλιοθήκη d3 ο σχεδιαστής επιλέγει μόνος του την προβολή. Μπορείτε να επιλέξετε από μία λίστα 14 διαφορετικών προβολών ή να δημιουργήσετε τη δική του, όπως γίνεται στο παρακάτω παράδειγμα (Frerichs, 2019):

```
var projection = d3.geoEqualEarth();
```

Σύμφωνα με την παραπάνω ανάλυση, μπορεί να εξαχθεί το συμπέρασμα του ότι η βιβλιοθήκη d3 υπερτερεί με γνώμονα το κριτήριο των δυνατοτήτων οπτικοποίησης σε μορφή χάρτη.

Αναφορικά με τη βιβλιοθήκη οπτικοποίησης δεδομένων Pygal, παρέχονται 13 διαφορετικές δυνατότητες προβολών μέσω των αντίστοιχων κλάσεων με τον χρήστη να έχει τη δυνατότητα να επιλέξει μία από αυτές (Pygal, 2016<sup>b</sup>). Ως εκ τούτου, σύμφωνα με το κριτήριο της οπτικοποίησης σε μορφή χάρτη, η βιβλιοθήκη d3 αποτελεί τη βέλτιστη δυνατή επιλογή, με την Pygal να ακολουθεί και τη Mapbox να βρίσκεται στην τελευταία θέση.

## 7.6 Centering the map

Η βιβλιοθήκη οπτικοποίησης δεδομένων Mapbox βασίζεται στον σχεδιαστή, ο οποίος έχει τη δυνατότητα ρύθμισης εκ των προτέρων του ζουμ, του κέντρου και ούτω καθεξής. Από την άλλη πλευρά, η βιβλιοθήκη οπτικοποίησης δεδομένων d3 καθιστά δυνατή τη χρησιμοποίηση των δεδομένων για τον ορισμό της έκτασης. Σχετικά παραδείγματα με τον κώδικα τους παρατίθενται παρακάτω. Το πρώτο λαμβάνει χώρα με τη χρήση της βιβλιοθήκης Mapbox και το δεύτερο με την d3 (Frerichs, 2019):

```
center: [13.79,53.545],  
zoom: 5  
projection.fitExtent([[20, 20], [width, height]], bb);
```

Από την παραπάνω ανάλυση μπορεί να θεωρηθεί ότι η βιβλιοθήκη d3 υπερτερεί της Mapbox με βάσει το κριτήριο της επικέντρωσης του χάρτη.

Από την άλλη πλευρά, η ίδια διαδικασία στο πλαίσιο της βιβλιοθήκης Pygal είναι παρόμοια. Χρησιμοποιείται ένας πίνακας αντιστοίχισης, στον οποίο ο χρήστης πρέπει να εισάγει τις οριζόντιες και κάθετες συντεταγμένες της επιθυμητής επικέντρωσης. Ένα παράδειγμα περιλαμβάνεται στον παρακάτω κώδικα (Pythonmana, 2022):

```
M = np.float32([[1,0,120], [0,1,20]]) # towards the right 120, Next 20, translation  
image_Move = cv.warpAffine(image, M, (w,h))
```

## 7.7 Σύνοψη σύγκρισης

Στον Πίνακα 1 παρουσιάζεται μία σύνοψη της σύγκρισης μεταξύ των βιβλιοθηκών οπτικοποίησης δεδομένων βάσει των παραπάνω κριτηρίων.

<b>ΚΡΙΤΗΡΙΟ</b>	<b>Mapbox.gl.js</b>	<b>D3.js</b>	<b>Pygal</b>
<b>Ποσότητα κώδικα</b>	Μεγάλη ποσότητα	Μικρή ποσότητα	Μικρή ποσότητα
<b>Ευκολία εγκατάστασης</b>	Συμπερίληψη κώδικα βιβλιοθήκης Javascript – CSS Mapbox  Εγγραφή στον ιστότοπο Mapbox	Συμπερίληψη βιβλιοθήκης	Εντολή pip install
<b>Styling</b>	Mapbox Style Spec  Υλοποίηση δύο επιπέδων  Πληθώρα εφαρμογών	SVG – CSS  Πληθώρα εφαρμογών	SVG  14 τυποποιημένα στυλ
<b>Basemap</b>	Χρησιμοποιείται	Δεν χρησιμοποιείται	Βασική κλάση
<b>Projections</b>	Mercator–Δεν υπάρχει δυνατότητα αλλαγής της προβολής	14 διαφορετικές προβολές Δυνατότητα δημιουργίας προβολής από τον χρήστη	13 διαφορετικές προβολές
<b>Centering the map</b>	Δυνατότητα ρύθμισης εκ των προτέρων από τον σχεδιαστή	Χρησιμοποίηση δεδομένων για τον ορισμό της έκτασης	Πίνακας αντιστοίχισης

Πίνακας 1. Σύγκριση βιβλιοθηκών οπτικοποίησης δεδομένων.

## Κεφάλαιο 8: Συμπεράσματα

Στα πλαίσια της παρούσας εργασίας εξετάστηκαν τρεις από τις πλέον διαδεδομένες βιβλιοθήκες οπτικοποίησης δεδομένων. Οι δύο από αυτές (Marbox, d3) βασίζονταν στην Java, ενώ η τρίτη (Pygal) σε Python. Κάθε μία από τις βιβλιοθήκες παρουσίαζε δυνατά και αδύναμα σημεία. Το γεγονός αυτό υποδεικνύει ότι η βιβλιοθήκη που θα χρησιμοποιηθεί πρέπει να επιλέγεται σύμφωνα με τις ανάγκες του εκάστοτε χρήστη, καθώς και ότι δεν μπορεί να προταθεί μία συγκεκριμένη βιβλιοθήκη για όλους τους χρήστες και τις εφαρμογές.

Σε γενικές γραμμές, όταν ζητείται από τον σχεδιαστή η υλοποίηση μίας απλής εφαρμογής, η βιβλιοθήκη Marbox GL JS ενδέχεται να αποδειχτεί αρκετά εξειδικευμένη για τη συγκεκριμένη διαδικασία (δηλαδή να μπορούσε το έργο να ολοκληρωθεί ευκολότερα με τη χρήση μίας από τις άλλες δύο βιβλιοθήκες). Όμως, όσο περισσότερο πολύπλοκη γίνεται η διαχείριση των δεδομένων που πρέπει να επεξεργαστεί ο χρήστης και όσο περισσότερο εξειδικευμένη πρέπει να είναι η λύση που θα υλοποιήσει ο σχεδιαστής, τόσο περισσότερο κατάλληλη είναι η βιβλιοθήκη Marbox GL JS. Συχνά η χρήση της Marbox GL JS μπορεί να αποδειχθεί μία περιττή περιπλοκή, εκτός εάν απαιτείται να δημιουργηθούν εφαρμογές οπτικοποίησης δεδομένων εκ του μηδενός.

Η βιβλιοθήκη Marbox GL JS προσφέρει πρόσβαση στις δυνατότητες των προσαρμοσμένων επιπέδων βάσης χωρίς να χρειάζεται ο σχεδιαστής να αποκτήσει κάποιο εξειδικευμένο επίπεδο γνώσεων αναφορικά με τα πλακίδια (tiles). Επιπρόσθετα, η βιβλιοθήκη Marbox GL JS υπερτερεί όταν πρέπει να υλοποιηθεί η χαρτογράφηση καθαρών και συνοπτικών δεδομένων, με μεσαίο έως υψηλό επίπεδο πολυπλοκότητας.

Η βιβλιοθήκη Marbox GL JS παρουσιάζει επιπρόσθετα το πλεονέκτημα της εύκολης παραμετροποίησης. Η βιβλιοθήκη παρέχει βασικούς χάρτες με προεπιλεγμένο στυλ και προσαρμογές. Αντί ο σχεδιαστής να πρέπει να ξεκινήσει εκ του μηδενός, αυτοί οι βασικοί χάρτες ξεκινούν τη διαδικασία προσαρμογής με το μοναδικό, προεπιλεγμένο στυλ τους, επομένως η επιλογή του σωστού βασικού χάρτη είναι κρίσιμη για τον καθορισμό του τόνου του σχεδίου και για να έχει ο σχεδιαστής κάθε απαραίτητο στοιχείο κατά την αρχή της υλοποίησης. Επιπρόσθετα, παρέχεται η

δυνατότητα προσαρμογής μέσω GUI του φόντου και της γραμματοσειράς, της προσθήκης υφής, αλλά και της επιλογής χρωμάτων.

Από την άλλη πλευρά, η βιβλιοθήκη d3 παρουσιάζει το πλεονέκτημα της δυνατότητας υλοποίησης εξατομικευμένων εφαρμογών. Αποτελεί μία πολύ ισχυρή βιβλιοθήκη για τη δημιουργία οπτικοποιήσεων σε περιβάλλον JavaScript. Η αποτελεσματική χρήση της απαιτεί βαθύτερη κατανόηση των τεχνικών JavaScript σε σχέση με τις άλλες βιβλιοθήκες. Ωστόσο, δίνει τη δυνατότητα μεγαλύτερου ελέγχου και ευελιξίας ως προς τα αποτελέσματα. Η βιβλιοθήκη d3 δεν παρέχει προκαθορισμένες απεικονίσεις, όπως γραφήματα ή χάρτες. Αντίθετα, είναι μία εργαλειοθήκη οπτικοποίησης δεδομένων η οποία παρέχει τα εργαλεία για να δημιουργήσει ο χρήστης τα δικά του γραφήματα, χάρτες και λοιπές εφαρμογές.

Το πλέον χαρακτηριστικό στοιχείο της βιβλιοθήκης d3 είναι η φιλοσοφία της. Η βιβλιοθήκη δεν είναι εργαλείο για τη δημιουργία προκαθορισμένων τύπων γραφημάτων και οπτικοποίησης. Αντίθετα, είναι μία βιβλιοθήκη που βοηθά να δημιουργηθεί οποιαδήποτε οπτικοποίηση, συμπεριλαμβανομένων προσαρμοσμένων και μοναδικών παρουσιάσεων. Χρειάζεται περισσότερη προσπάθεια για τη δημιουργία ενός τυπικού γραφήματος με τη d3, αλλά η βιβλιοθήκη δεν περιορίζεται σε τυπικά γραφήματα.

Στην περίπτωση εφαρμογών ιστοσελίδων όπου απαιτείται η σύνδεση με άλλα κομμάτια κώδικα η d3 είναι η καταλληλότερη επιλογή. Είναι μία βιβλιοθήκη JavaScript η οποία χρησιμοποιείται για τη δημιουργία διαδραστικών απεικονίσεων στο πρόγραμμα περιήγησης. Η βιβλιοθήκη d3.js καθιστά δυνατή τη διαχείριση των στοιχείων μίας ιστοσελίδας στο πλαίσιο ενός συνόλου δεδομένων. Αυτά τα στοιχεία μπορεί να είναι στοιχεία HTML, SVG ή CSS και μπορούν να εισαχθούν, να αφαιρεθούν ή να επεξεργαστούν σύμφωνα με τα περιεχόμενα του συνόλου δεδομένων. Αποτελεί είναι πολύτιμο βοήθημα στην εξερεύνηση δεδομένων, δίνει τον έλεγχο της αναπαράστασης των δεδομένων στον σχεδιαστή της ιστοσελίδας και καθιστά δυνατή την προσθήκη διαδραστικότητας.

Η βιβλιοθήκη d3 λειτουργεί στον Παγκόσμιο Ιστό και οι οπτικοποιήσεις δεδομένων της είναι σχεδιασμένες ούτως ώστε να ενσωματώνονται άμεσα σε ιστοσελίδες. Ένα ακόμα σχετικό σημαντικό πλεονέκτημα της είναι η ευελιξία της. Δεδομένου ότι λειτουργεί άψογα με τις υπάρχουσες τεχνολογίες Ιστού και μπορεί να



χειριστεί οποιοδήποτε μέρος του μοντέλου αντικειμένου εγγράφου, είναι τόσο ευέλικτη όσο και οι HTML, CSS και SVG.

Αναφορικά με τη δυνατότητα οπτικοποίησης στοιχείων τα οποία εμπεριέχονται σε βάσεις δεδομένων, η βιβλιοθήκη Pygal παρουσιάζει σαφές πλεονέκτημα. Η Python αποτελεί ένα βασικό εργαλείο εφαρμογών ανάλυσης δεδομένων. Η Pygal, ως βιβλιοθήκη της Python παρέχει σημαντικές δυνατότητες αξιοποίησης αυτών των δυνατοτήτων για την υλοποίηση των σχετικών εφαρμογών.

Η Pygal περιλαμβάνει επίσης αρκετά καλή και ευανάγνωστη τεκμηρίωση. Ένα κοινό χαρακτηριστικό των βιβλιοθηκών Python είναι ότι η τεκμηρίωση μπορεί να κυμαίνεται από ένα καλά τεκμηριωμένο wiki με δυνατότητα αναζήτησης στο διαδίκτυο έως ένα απλό αρχείο readme.txt που παρουσιάζει μόνο τον τρόπο εγκατάστασης της βιβλιοθήκης. Η βιβλιοθήκη Pygal επίσης δεν απαιτεί πολλές εξαρτήσεις, κάτι που είναι ιδιαίτερος σημαντικό για τους νέους χρήστες, καθώς μία βιβλιοθήκη με πολλές εξαρτήσεις μπορεί να προκαλέσει προβλήματα σε νέους προγραμματιστές.

Συμπερασματικά θα έλεγε κανείς ότι τα τρία αυτά εργαλεία – βιβλιοθήκες (matplotlib, d3, pygal) απαντάνε σε ένα πολύ μεγάλο πλήθος εργασιών που θα έκανε κάποιος σήμερα στο γενικότερο αντικείμενο του data visualization.

## Βιβλιογραφικές αναφορές

AnyCodings. (2022). Python pygal code - World map [online]. Διαθέσιμο στον σύνδεσμο: <https://www.anycodings.com/1questions/4662610/python-pygal-code-world-map>. Προσπελάστηκε στις: 19/08/2022.

Allen Wirfs-Brock and Brendan Eich. 2020. JavaScript: The First 20 Years. 4, HOPL (2020). DOI:10.1145/3386327

Bostock, M., Ogievetsky, V., and Heer, J. (2011). D3: Data-Driven Documents, *The Stanford Visualization Group* [online], Διαθέσιμο στον σύνδεσμο: <http://vis.stanford.edu/files/2011-D3-InfoVis.pdf>. Προσπελάστηκε στις: 12/06/2022.

Christopher G. Healey, James. T. Enns. (2011). Attention and Visual Memory in Visualization and Computer Graphics.

Despina Meridou (2014), Improving Quality of Life with the Internet of Everything, DOI:10.1007/978-3-319-50758-3\_15

Eriksson, O., and Rydkvist, E. (2015). An in-depth analysis of dynamically rendered vector-based maps with WebGL using Mapbox GL JS, M.Sc. Thesis, Institutionen för datavetenskap, Linköping, Sweden.

Frerichs, M. (2019). Comparing Mapbox GL JS and D3.js, *Better Programming* [online], Διαθέσιμο στον σύνδεσμο: <https://betterprogramming.pub/comparison-of-mapbox-gl-js-and-d3-js-66dc4525b70b>. Προσπελάστηκε στις: 17/08/2022.

Geeks for Geeks. (2020). Styling Graphs in Pygal [online]. Διαθέσιμο στον σύνδεσμο: <https://www.geeksforgeeks.org/styling-graphs-in-pygal/>. Προσπελάστηκε στις: 18/08/2022.

J. Hunter, et al., matplotlib: Python plotting, <http://matplotlib.sourceforge.net/>. Προσπελάστηκε στις: 12/06/2022

Mapbox. (2022<sup>a</sup>). Introduction to Mapbox [online], Διαθέσιμο στον σύνδεσμο: <https://docs.mapbox.com/help/getting-started/#introduction-to-mapbox>. Προσπελάστηκε στις: 01/08/2022.

Mapbox,(2022b).Installation [online], Διαθέσιμο στον σύνδεσμο: <https://docs.mapbox.com/mapbox-gl-js/guides/install/>. Προσπελάστηκε στις: 01/08/2022.

Meeks, E. (2017). *D3.js in Action: Data visualization with JavaScript*, 2<sup>nd</sup> Ed., Manning, Shelter Island, NY.

Michael Friendly, Chun-houh Chen, Wolfgang Karl Härdle και Antony Unwin, «A brief history of data visualization», *Handbook of Data Visualization*, Βερολίνο 2008.

Netek, R., Masopust, J., Pavlicek, F., and Pehanec, F. (2020). Performance Testing on Vector vs. Raster Map Tiles—Comparative Study on Load Metrics, *International Journal of Geo-Information*, 9, pp. 101-125.

Nick Qi Zhu, “Data Visualization with D3.js Cookbook”, Packt, 2013

Pablo Navarro Castillo, “Mastering D3.js”, Packt, 2014

Pygal. (2016a). pygal.graph.map module [online], Διαθέσιμο στον σύνδεσμο: <https://www.pygal.org/en/3.0.0/api/pygal.graph.map.html>. Προσπελάστηκε στις: 20/08/2022.

Pygal. (2016b). pygal.view module [online], Διαθέσιμο στον σύνδεσμο: <https://www.pygal.org/en/3.0.0/api/pygal.view.html>. Προσπελάστηκε στις: 20/08/2022.

Python Data Analysis Library — Pandas: Python Data Analysis Library. <https://pandas.pydata.org/>. Accessed 3 Oct. 2019

Pythonmana. (2022). Comparing Mapbox GL JS and D3.js [online], Διαθέσιμο στον σύνδεσμο: <https://pythonmana.com/2022/189/202207080534031209.html> Προσπελάστηκε στις: 23/08/2022.

Python Programming Tutorials---. <https://pythonprogramming.net/dataanalysis-python-pandas-tutorial-introduction/>. Accessed 3 Oct. 2019.

Python, Real. Python Pandas: Tricks & Features Youσ May Not Know – Real Python. <https://realpython.com/python-pandas-tricks/>. Accessed 3 Oct. 2019

Ranjan, A., Sinha, A. and Battewad, R. (2020) *JavaScript for Modern Web Development*. [edition missing]. BPB Publications

Schwenke, J. R. and Fergen, B. J. (2002). *Graphical Techniques for Displaying Multivariate Data Using SAS/GRAPH Software*, Observations, <ftp.sas.com/techsup/download/observations/obswww22/obswww22.pdf>

Seaborn: Statistical Data Visualization — Seaborn 0.9.0 Documentation. <https://seaborn.pydata.org/>. Accessed 3 Oct. 2019.

Seaborn Tutorial - Tutorialspoint. <https://www.tutorialspoint.com/seaborn/index.htm>. Accessed 3 Oct. 2019.

Walia, M.S. (2022). A Comprehensive Guide On Pygal – The Next Generation Data W. McKinney, AQR Capital Management, pandas: a python data analysis library, <http://pandas.sourceforge.net>

Visualization Library In Python, Analytics Vidhya [online], Διαθέσιμο στον σύνδεσμο: <https://www.analyticsvidhya.com/blog/2022/04/a-comprehensive-guide-on-pygal-the-next-generation-data-visualization-library-in-python/>. Προσπελάστηκε στις: 14/07/2022.

X. Frank Zhang(2010), Information Uncertainty and Analyst Forecast Behavior, <https://doi.org/10.1506/92CB-P8G9-2A31-PV0R>

Zhicheng Liu,Biye Jiang,Jeffrey Heer( 2013), Real-time Visual Querying of Big Data, <https://doi.org/10.1111/cgf.12129>

Καλλίνικος, Γεώργιος Α.Μ. 0993Μελέτη σύγχρονων JavaScript Frameworks και σύγκριση αυτών στον τομέα των φορμών. <http://repository.library.teimes.gr/xmlui/handle/123456789/7498>.

## **Βιβλία**

Chad Adams (2014). *Learning-Python-Data-Visualization*. BirminghamUK.Packt Publishing.

D.Flanagan(2020) *.JavaScript: The Definitive Guide*. (7th Edition). Sebastopol, US: O'Reilly Media, Inc.

Stephen A. Thomas (2015). *Data,Visualization-with-JavaScript*. USA: No Starch Press.

E. Meeks (2018). *D3.js in Action* .(2nd Edition.) Canada: Manning Publications Co.