**Final Assignment in Python  
Report**

**Βήμα 1:**

Ξεκινάμε ανοίγοντας το «**finance\_liquor\_sales**» αρχείο με το MySQL Workbench και το τρέχουμε ώστε να δημιουργηθεί η βάση δεδομένων. Έπειτα δημιουργούμε ένα query

|  |
| --- |
| select \*  from finance\_liquor\_sales  where year(date) between 2016 and 2019 |

ώστε να μας εμφανίσει όλες τις εγγραφές από το 2016 έως το 2019 και στην συνέχεια το κάνουμε export ως «**results16-19.csv**»

**Βήμα 2:**

Αφού πλέον έχουμε τελειώσει με το MySQL Workbench ανοίγουμε το PyCharm ώστε να γράψουμε τον απαραίτητο κώδικα για τα ζητούμενα της άσκησης.

**Βήμα 3:**   
Ξεκινάμε ανοίγοντας το αρχείο που δημιουργήσαμε προηγουμένως

df = pd.read\_csv('results16-19.csv')

και στην συνέχεια θα βρούμε το πιο διάσημο προϊόν βάσει του ταχυδρομικού κώδικα.

Ξεκινάμε ομαδοποιώντας τα δεδομένα μας βάσει των «**item\_description**» και «**zip\_code**» και στην συνέχεια αθροίζουμε τα «**bottles\_sold**»

grouped = df.groupby(['item\_description', 'zip\_code'])  
item\_sold = grouped['bottles\_sold'].sum()

Για να εκτυπώσουμε το πρώτο προϊόν, πρώτα θα ταξινομήσουμε τα δεδομένα μας κατά φθίνουσα σειρά και έπειτα θα εμφανίσουμε μόνο την πρώτη γραμμή

print("The most popular item sold based on zip code is:\n", item\_sold.sort\_values(ascending=False).head(1))

Το αποτέλεσμα που παίρνουμε είναι το ακόλουθο

|  |
| --- |
| The most popular item sold based on zip code is:  item\_description zip\_code  Juarez Gold Dss 52314.0 1560  Name: bottles\_sold, dtype: int64 |

**Βήμα 4:**

Τώρα θέλουμε να βρούμε το ποσοστό των πωλήσεων ανά μαγαζί.

Ξεκινάμε ομαδοποιώντας τα δεδομένα μας βάσει του «**store\_number**» και έπειτα αθροίζουμε βάσει «**sale\_dollars**»

grouped = df.groupby('store\_number')  
store\_sales = grouped['sale\_dollars'].sum().reset\_index()

Χρησιμοποιούμε το «**.reset\_index()**» καθώς οι δείκτες χρησιμοποιήθηκαν στο προηγούμενο κάλεσμα της **sum()**, οπότε πρέπει να τους μηδενίσουμε. Εδώ ήταν και η πρώτη δυσκολία που αντιμετώπισα καθώς δεν γνώριζα ότι οι δείκτες συνεχίζουν την αρίθμηση από εκεί που σταμάτησαν.

Τώρα θα βρούμε το ποσοστό των πωλήσεων και θα το εκχωρήσουμε σε μία καινούργια στήλη με όνομα «**percent\_sales**»

total\_sales = store\_sales['sale\_dollars'].sum()  
store\_sales['percent\_sales'] = store\_sales['sale\_dollars'] / total\_sales \* 100

Τον υπολογισμό των συνολικών αγορών όλων τον καταστημάτων τον εκχώρησα πρώτα στην τιμή «**total\_sales**» ώστε να είναι πιο ευδιάκριτος ο υπολογισμός του ποσοστού. Τέλος για να εμφανίσουμε τα αποτελέσματα θα στρογγυλοποιήσουμε το ποσοστό μας για να είναι πιο ευδιάκριτο, προτίμησα τα 3 δεκαδικά.

print("The percentage of sales per store is:\n", round(store\_sales, 3))

Το αποτέλεσμα που παίρνουμε είναι το ακόλουθο:

|  |  |  |  |
| --- | --- | --- | --- |
| The percentage of sales per store is: | | | |
|  | store\_number | sale\_dollars | percent\_sales |
| 0 | 2178 | 9.74 | 0.016 |
| 1 | 2465 | 41.55 | 0.068 |
| 2 | 2512 | 706.32 | 1.153 |
| 3 | 2515 | 900.24 | 1.469 |
| 4 | 2522 | 324.96 | 0.530 |
| .. | … | … | … |
| 56 | 5204 | 81.60 | 0.133 |
| 57 | 5244 | 75.12 | 0.123 |
| 58 | 5416 | 112.47 | 0.184 |
| 59 | 5446 | 21.00 | 0.034 |
| 60 | 9001 | 3712.50 | 6.060 |
|  |  |  |  |
| [61 rows x 3 columns] | | | |

Βήμα 5:

Για την δημιουργία ενός scatter plot θα ομαδοποιήσουμε τα δεδομένα μας βάσει «zip\_code» και θα αθροίσουμε τα «bottles\_sold», έτσι θα έχουμε τα μπουκάλια που πουλήθηκαν συνολικά με βάσει τον ταχυδρομικό κώδικα.

bottles\_sold\_by\_zip = df.groupby('zip\_code')['bottles\_sold'].sum().reset\_index()

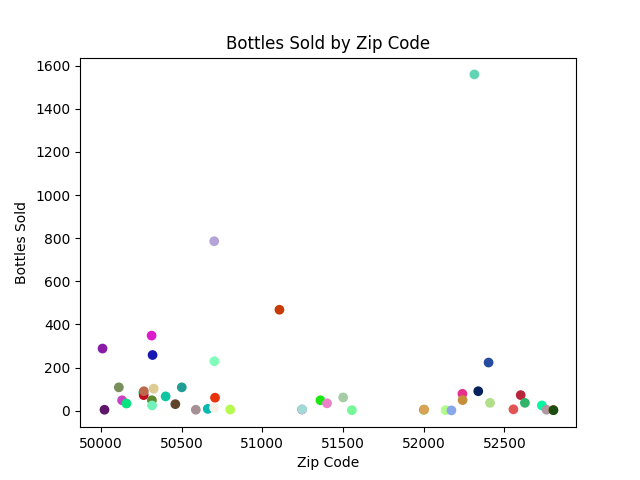
και σε αυτήν την περίπτωση πρέπει να χρησιμοποιήσουμε το «.reset\_index()»

Για την δημιουργία διαφορετικών χρωμάτων (για κάθε ταχυδρομικό κώδικα) θα καταχωρήσουμε σε μία μεταβλητή χ το μήκος της μεταβλητής «bottles\_sold\_by\_zip» και στην συνέχεια στο κάλεσμα της εντολής «plt.scatter()» θα χρησιμοποιήσουμε την συνάρτηση «np.random.rand(x,3)». Υπήρξε κι εδώ δυσκολία ώστε να βρω έναν τρόπο για να δώσω διαφορετικά χρώματα σε κάθε τιμή. Η συνάρτηση αυτή δημιουργεί έναν πίνακα χΧ3 διαστάσεων, όπου περιέχει τιμές RGB για κάθε γραμμή.

x = len(bottles\_sold\_by\_zip)  
plt.scatter(bottles\_sold\_by\_zip['zip\_code'], bottles\_sold\_by\_zip['bottles\_sold'], c=np.random.rand(x,3))

Τώρα θα δώσουμε όνομα στον τίτλο και στο χ,ψ άξονα και με την εντολή «plt.show()» θα εμφανίσουμε το γράφημά μας

plt.title('Bottles Sold by Zip Code')  
plt.xlabel('Zip Code')  
plt.ylabel('Bottles Sold')  
plt.show()



Όλα τα απαραίτητα αρχεία και οι κώδικες βρίσκονται στον ακόλουθο σύνδεσμο

<https://github.com/nlymperidis/FinalAssignment>

Σαν επιπλέον εξάσκηση έκανα τα αντίστοιχα γραφήματα της άσκησης και στο

tableau. Τα αποτελέσματα βρίσκονται στο ακόλουθο λινκ:

<https://public.tableau.com/views/Final-Assignment-PythonData/Dashboard1?:language=en-US&publish=yes&:display_count=n&:origin=viz_share_link>