

ΕΣΤΟΙΧΕΙΑ ΟΜΑΔΑΣ

1^ο μέλος: Διονυσία Ψυρρή, AM:1080424, 4^ο έτος, email: up1080424@upnet.gr

ΕΡΩΤΗΜΑ 1

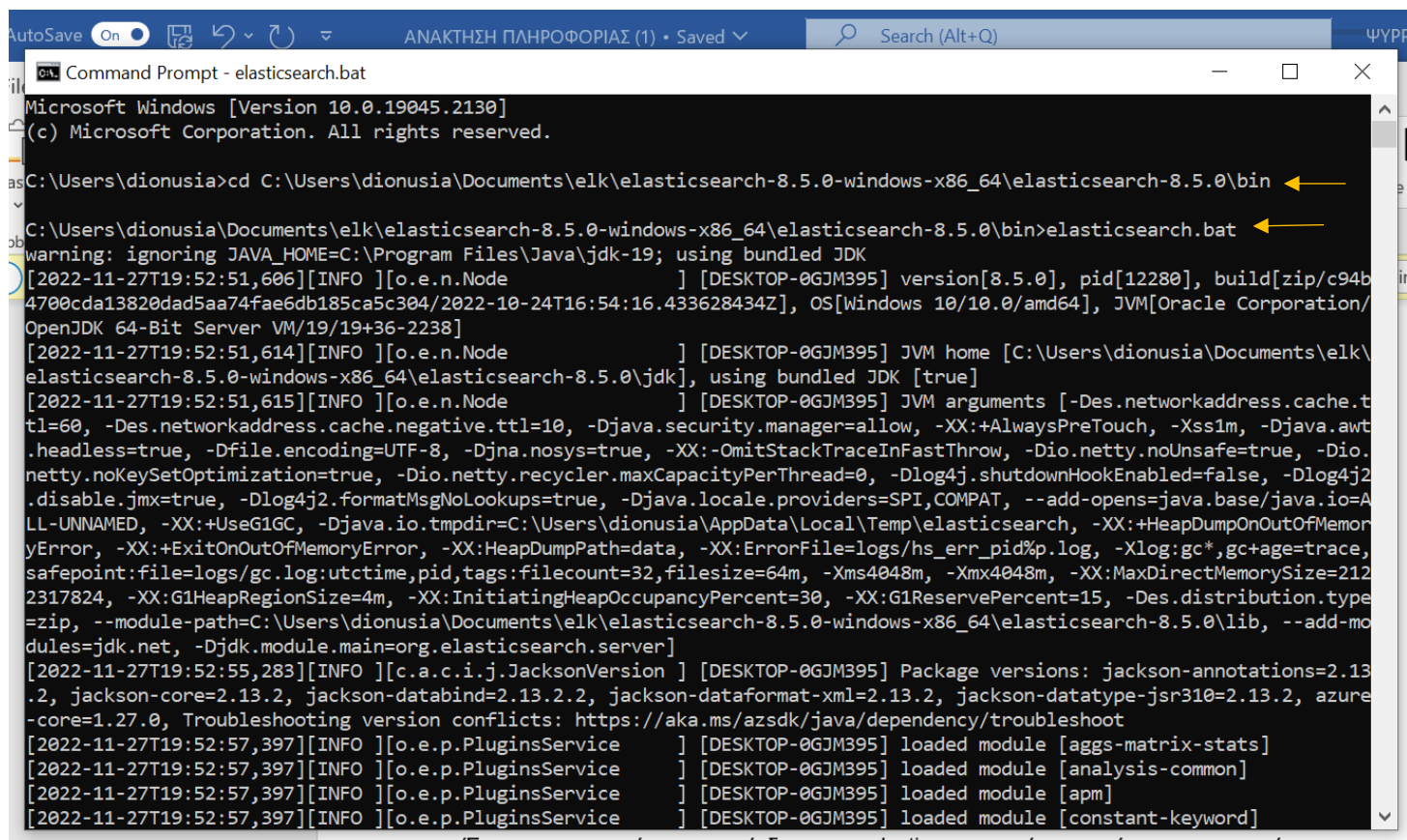
Εργαλεία εγκατάστασης: pip install elasticsearch(vsc)

Python: 3.7.9

Elasticsearch: 8.5.0

pip install pandas:numpy-1.21.6 pandas-1.3.5 python-dateutil-2.8.2 pytz-2022.6 six-1.16.0

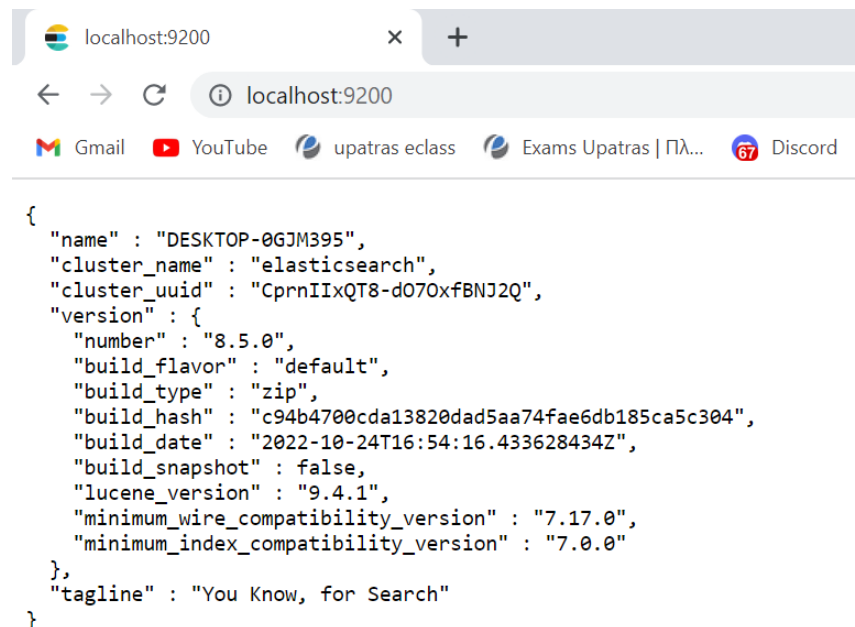
Έπειτα απο εγκατάσταση της έκδοσης της elastic που ανεφέρεται επάνω για να μπορέσουμε να συνδεθούμε εκτελούμε τις παραπάνω εντολές στο command prompt:



```
AutoSave On [Icons] [Undo] [Redo] [Refresh] [Close] ANΑΚΤΗΣΗ ΠΛΗΡΟΦΟΡΙΑΣ (1) • Saved Search (Alt+Q) ΨΥΡΡΗ
Command Prompt - elasticsearch.bat
Microsoft Windows [Version 10.0.19045.2130]
(c) Microsoft Corporation. All rights reserved.

C:\Users\dionusia>cd C:\Users\dionusia\Documents\elk\elasticsearch-8.5.0-windows-x86_64\elasticsearch-8.5.0\bin
C:\Users\dionusia\Documents\elk\elasticsearch-8.5.0-windows-x86_64\elasticsearch-8.5.0\bin>elasticsearch.bat
warning: ignoring JAVA_HOME=C:\Program Files\Java\jdk-19; using bundled JDK
[2022-11-27T19:52:51,606][INFO ][o.e.n.Node ] [DESKTOP-0GJM395] version[8.5.0], pid[12280], build[zip/c94b
4700cda13820dad5aa74fae6db185ca5c304/2022-10-24T16:54:16.433628434Z], OS[Windows 10/10.0/amd64], JVM[Oracle Corporation/
OpenJDK 64-Bit Server VM/19/19+36-2238]
[2022-11-27T19:52:51,614][INFO ][o.e.n.Node ] [DESKTOP-0GJM395] JVM home [C:\Users\dionusia\Documents\elk\
elasticsearch-8.5.0-windows-x86_64\elasticsearch-8.5.0\jdk], using bundled JDK [true]
[2022-11-27T19:52:51,615][INFO ][o.e.n.Node ] [DESKTOP-0GJM395] JVM arguments [-Des.networkaddress.cache.t
tl=60, -Des.networkaddress.cache.negative.ttl=10, -Djava.security.manager=allow, -XX:+AlwaysPreTouch, -Xss1m, -Djava.awt
.headless=true, -Dfile.encoding=UTF-8, -Djna.nosys=true, -XX:-OmitStackTraceInFastThrow, -Dio.netty.noUnsafe=true, -Dio.
netty.noKeySetOptimization=true, -Dio.netty.recycler.maxCapacityPerThread=0, -Dlog4j.shutdownHookEnabled=false, -Dlog4j2
.disable.jmx=true, -Dlog4j2.formatMsgNoLookups=true, -Djava.locale.providers=SPI,COMPAT, --add-opens=java.base/java.io=A
LL-UNNAMED, -XX:+UseG1GC, -Djava.io.tmpdir=C:\Users\dionusia\AppData\Local\Temp\elasticsearch, -XX:+HeapDumpOnOutOfMemory
Error, -XX:+ExitOnOutOfMemoryError, -XX:HeapDumpPath=data, -XX:ErrorFile=logs/hs_err_pid%p.log, -Xlog:gc*,gc+age=trace,
safepoint:file=logs/gc.log:utctime,pid,tags:filecount=32,filesize=64m, -Xms4048m, -Xmx4048m, -XX:MaxDirectMemorySize=212
2317824, -XX:G1HeapRegionSize=4m, -XX:InitiatingHeapOccupancyPercent=30, -XX:G1ReservePercent=15, -Des.distribution.type
=zip, --module-path=C:\Users\dionusia\Documents\elk\elasticsearch-8.5.0-windows-x86_64\elasticsearch-8.5.0\lib, --add-mo
dules=jdk.net, -Djdk.module.main=org.elasticsearch.server]
[2022-11-27T19:52:55,283][INFO ][c.a.c.i.j.JacksonVersion ] [DESKTOP-0GJM395] Package versions: jackson-annotations=2.13
.2, jackson-core=2.13.2, jackson-databind=2.13.2.2, jackson-dataformat-xml=2.13.2, jackson-datatype-jsr310=2.13.2, azure
-core=1.27.0, Troubleshooting version conflicts: https://aka.ms/azsdk/java/dependency/troubleshoot
[2022-11-27T19:52:57,397][INFO ][o.e.p.PluginsService ] [DESKTOP-0GJM395] loaded module [aggs-matrix-stats]
[2022-11-27T19:52:57,397][INFO ][o.e.p.PluginsService ] [DESKTOP-0GJM395] loaded module [analysis-common]
[2022-11-27T19:52:57,397][INFO ][o.e.p.PluginsService ] [DESKTOP-0GJM395] loaded module [apm]
[2022-11-27T19:52:57,397][INFO ][o.e.p.PluginsService ] [DESKTOP-0GJM395] loaded module [constant-keyword]
```

Έπειτα ανοίγουμε έναν browser και πληκτρολογούμε την εντολή localhost:9200:



Έπειτα συνδεθήκαμε επιτυχώς στην elasticsearch και προχωρούμε στα επόμενα βήματα.

Ο κώδικας για το διάβασμα και την φόρτωση των δεδομένων είναι στα αρχεία **loadBooks.py**, **loadRatings.py**, διαβάζουμε τα αρχεία με την pandas, τα μετατρέπουμε σε json file (για την αναπαράσταση και αποθήκευση δεδομένων) και έπειτα αυτά τα json τα μετατρέπουμε σε python dictionary (καθώς ένα λεξικό Python είναι η πραγματική δομή δεδομένων (αντικείμενο) που διατηρείται στη μνήμη ενώ εκτελείται ένα πρόγραμμα Python) ώστε να κάνουμε μαζική εισαγωγή δεδομένων στην elasticsearch με την helpers.bulk της οποίας το πλεονέκτημα που την χρησιμοποιήσαμε αντί για ένα for loop για παράδειγμα είναι ότι με μια κλήση κάνει μαζική εισαγωγή πολλαπλών εγγραφών χωρίς να υπολογίζει εκ νέου κάθε φορά όπως θα γινόταν εάν χρησιμοποιούσαμε for loop

Συντομη περιγραφή του κώδικα για το ερώτημα, ο κώδικας είναι στο αρχείο **quest1.py**:

συνδεόμαστε στην elasticsearch για να πάρουμε τα δεδομένα που χρειαζόμαστε, δημιουργήσαμε δυο άδεια dataframe ένα για να αποθηκεύσουμε τα δεδομένα του query για το match του string που έδωσε ο χρήστης με τους τίτλους και ένα για το query για το αναγνωριστικό του χρήστη και στις δυο περιπτώσεις χρησιμοποιήσαμε την συνάρτηση json_normalize() ώστε τα αποτελέσματά μας να τα πάρουμε σε επίπεδους πίνακες, όσο για το ταίριασμα του uid και του book_title, και στα δυο query παίρνουν το κοινό field isbn ώστε έπειτα και από την ολοκλήρωση και των δυο query με ένα for loop ελέγχουμε τα κοινά isbn ώστε να έχουμε κάνει το match που θέλουμε και να αποθηκεύσουμε τα ζητούμενα ratings στην λίστα search_user_rating. Τέλος την επιστροφή του 10% των βιβλίων με το καλύτερο ταίριασμα το πετύχαμε με την εντολή: **print(dfFreq.head(int(len(dfFreq)*(n/100))))**, όπου ορίσαμε το n = 10 και μέσω της head() η οποία μας δίνει την δυνατότητα να εμφανίσουμε τον αριθμό των σειρών της επιλογής μας, επιστρέψαμε το 10%

Παραθέτουμε ένα παράδειγμα για να δείξουμε το 10%:

Χωρίς την εντολή head():

```
quest1.py > search
34 for j in range(len(results["_source.isbn"])):
35     if isbn[i] == results["_source.isbn"][j]:
36         search_user_rating[i] = results["_source.rating"][j]
37
38 for i in range(len(books)):
39     books["_score"][i] = metric([scores[i], search_user_rating[i]])
40
41 n = 10
42 books = books.sort_values(by='_score', ascending=False)
43 print("Best Match with user with ID:" ,userId)
44 dffreq = pd.DataFrame(books[["_score", "_source.book_title"]])
45 print(dffreq)
46
47
48 if __name__ == "__main__":
49     ..
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

PS C:\Users\dionusia\Desktop\information retrieval> & C:/Users/dionusia/AppData/Local/Programs/Python/Python37/python.exe "c:/Users/dionusia/Desktop/information retrieval/quest1.py"

Enter Title: A

Enter Id: 2

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

_source.isbn		
0393324494	1.421167	A Murder, a Mystery and a Marriage: A Story
0393043762	1.384666	A Murder, A Mystery, and a Marriage
0843952768	1.384666	A Girl, a Guy, and a Ghost
0060554657	1.384666	A Year and a Day : A Novel
0091831040	1.381100	A Necessary Evil, A
...
0679734570	0.867713	The Golden Gate: A Novel in Verse
0763610844	0.867713	Maisy Takes a Bath (Maisy Books (Paperback))
0807070920	0.867713	Point Last Seen: A Woman Tracker's Story
0061043990	0.867713	She Came Back (A Miss Silver Mystery)
0446515892	0.867713	Darwin: The Life of a Tormented Evolutionist

[10000 rows x 2 columns]

PS C:\Users\dionusia\Desktop\information retrieval>

Με την εντολή head():

```
quest1.py >
34 for j in range(len(results["_source.isbn"])):
35     if isbn[i] == results["_source.isbn"][j]:
36         search_user_rating[i] = results["_source.rating"][j]
37
38 for i in range(len(books)):
39     books["_score"][i] = metric([scores[i], search_user_rating[i]])
40
41 n = 10
42 books = books.sort_values(by='_score', ascending=False)
43 print("Best Match with user with ID:" ,userId)
44 dffreq = pd.DataFrame(books[["_score", "_source.book_title"]])
45 print(dffreq.head(int(len(dffreq)/n*100)))
46
47
48 if __name__ == "__main__":
49     ..
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

PS C:\Users\dionusia\Desktop\information retrieval> & C:/Users/dionusia/AppData/Local/Programs/Python/Python37/python.exe "c:/Users/dionusia/Desktop/information retrieval/quest1.py"

Enter Title: A

Enter Id: 2

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

_source.isbn		
0393324494	1.421167	A Murder, a Mystery and a Marriage: A Story
0393043762	1.384666	A Murder, A Mystery, and a Marriage
0843952768	1.384666	A Girl, a Guy, and a Ghost
0060554657	1.384666	A Year and a Day : A Novel
0091831040	1.381100	A Necessary Evil, A
...
1400062772	1.148650	Donorboy : A Novel
0449219682	1.148650	A Soldier's Heart
0670822469	1.148650	A Stranger's House
0060540761	1.148650	Taft : A Novel
067172696X	1.148650	A BITTER PEACE

[1000 rows x 2 columns]

PS C:\Users\dionusia\Desktop\information retrieval>

Ln 44, Col 44 Spaces