

ΒΑΣΕΙΣ II



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΠΑΤΡΩΝ
UNIVERSITY OF PATRAS

ΓΑΡΔΙΚΗΣ ΠΑΝΑΓΙΩΤΗΣ, Α.Μ:6013

ΤΣΑΛΙΔΗΣ ΑΝΔΡΕΑΣ , Α.Μ:6239

ΕΡΩΤΗΜΑ 1Α

Δημιουργούμε τα tables και τα families στην Hbase που μας ζητούνται με τις εντολές

```
CREATE 'USER02.YELPBUSINESS', 'BASE', 'ATTRIBUTES', 'HOURS'
```

```
CREATE 'USER02.YELPCHECKIN', 'PERHOUR'
```

Για να γεμίσουμε τα families με τα αντίστοιχα δεδομένα από τα csv θα κάνουμε χρήση του εργαλείου ImportTsv με την εντολή

```
hbase org.apache.hadoop.hbase.mapreduce.ImportTsv -Dimporttsv.separator=',' -  
Dimporttsv.columns=HBASE_ROW_KEY,BASE:NAME,BASE:NEIGHBORHOOD,BASE:ADDRESS,BASE  
:CITY,BASE:STATE,BASE:POSTALCODE,BASE:LATITUDE,BASE:LONGITUDE,BASE:STARS,BASE:REVIE  
WCOUNT,BASE:ISOPEN,BASE:CATEGORIES USER02.YELPBUSINESS hdfs://snf-  
817076.vm.okeanos.grnet.gr:/user/hbase/dataset/yelp_business.csv
```

Με παρόμοιο τρόπο και οι εντολές για τα υπόλοιπα families, οι οποίες βρίσκονται στο αρχείο Hbase.txt που σας παραθέτουμε μαζί με την ανάφορα.

ΕΡΩΤΗΜΑ 1Β

I) Για να γίνει η διασύνδεση από την HBASE στο PHOENIX θα πρέπει πρώτα να πάμε στον φάκελο /usr/hdp/2.6.3.0-235/phoenix/bin/ και να χρησιμοποιήσουμε το εργαλείο sqlline.py. Όταν βρεθούμε μέσα στον φάκελο /usr/hdp/2.6.3.0-235/phoenix/bin/(cd /usr/hdp/2.6.3.0-235/phoenix/bin/) εκτελούμε την εντολή ./sqlline.py για να συνδεθούμε στο phoenix. Για να μπορέσουμε να χρησιμοποιήσουμε στο περιβάλλον του Phoenix ένα table το οποίο έχει φτιαχτεί στην hbase, χρειάζεται να φτιάξουμε το View του table αυτού. Αυτό γίνεται με την εντολή

```
CREATE VIEW USER02.YELPCHECKIN (ROWKEY VARCHAR PRIMARY KEY, PERHOUR.BUSINESSID  
VARCHAR, PERHOUR.WEEKDAY VARCHAR, PERHOUR.HOUR VARCHAR, PERHOUR.CHECKINS  
VARCHAR);
```

Με παρόμοιο τρόπο και η εντολή για το άλλο table, η οποία βρίσκεται στο αρχείο Phoenix.txt που σας παραθέτουμε μαζί με την ανάφορα.

II) Για να εξυπηρετούνται πιο αποδοτικά τα ερωτήματα μας (σε πιο γρήγορο χρόνο) θα χρησιμοποιήσουμε τα secondary indexes. Με τη δευτερεύουσα ευρετηρίαση, οι στήλες ή οι εκφράσεις που ορίζουμε σχηματίζουν ένα εναλλακτικό κλειδί γραμμής για να επιτρέπουν αναζητήσεις σημείων και ανίχνευση εμβέλειας κατά μήκος αυτού του νέου άξονα.

Οι εντολές που εφαρμόσαμε για κάθε ερώτημα για αποδοτικότερη εξυπηρέτηση είναι οι εξής:

- ForQuery1:

```
CREATE INDEX Q1 ON USER02.YELPBUSINESS (BASE.ISOPEN)  
INCLUDE(BASE.NAME,BASE.STATE,BASE.STARS);
```
- ForQuery2:

```
CREATE INDEX Q2 ON USER02.YELPBUSINESS (BASE.CATEGORIES)  
INCLUDE(BASE.NAME,BASE.ADDRESS,BASE.CITY,BASE.REVIEWCOUNT);
```

- *ForQuery3: CREATE INDEX Q3 ON USER02.YELPBUSINESS (ATTRIBUTES.OPEN24HOURS, BASE.ISOPEN, HOURS.MONDAY, HOURS.TUESDAY, HOURS.WEDNESDAY, HOURS.THURSDAY, HOURS.FRIDAY, HOURS.SATURDAY, HOURS.SUNDAY) INCLUDE (BASE.CATEGORIES, BASE.REVIEWCOUNT);*
- *ForQuery4: CREATE INDEX Q4 ON USER02.YELPBUSINESS (ATTRIBUTES.SMOKING, HOURS.SUNDAY) INCLUDE(BASE.STATE);*
- *ForQuery5: CREATE INDEX Q5 ON USER02.YELPCHECKIN (PERHOUR.CHECKINS) INCLUDE(PERHOUR.WEEKDAY, PERHOUR.HOUR) ASYNC;*
- *ForQuery6: CREATE INDEX Q6C ON USER02.YELPCHECKIN (PERHOUR.WEEKDAY, PERHOUR.HOUR) INCLUDE (PERHOUR.CHECKINS) ASYNC;*
- *ForQuery6: CREATE INDEX Q6B ON USER02.YELPCHECKIN (BASE.ISOPEN) INCLUDE (BASE.CATEGORIES);*
- *ForQuery7: CREATE INDEX Q7C ON USER02.YELPCHECKIN (PERHOUR.WEEKDAY) INCLUDE(PERHOUR.CHECKINS) ASYNC;*
- *ForQuery7: CREATE INDEX Q7B ON USER02.YELPBUSINESS (BASE.NAME,BASE.NEIGHBORHOOD,BASE.ADDRESS,BASE.CITY,BASE.STATE,BASE.POSTALCODE,BASE.LATITUDE,BASE.LONGITUDE,BASE.STARS,BASE.REVIEWCOUNT,BASE.ISOPEN,BASE.CATEGORIES);*

III) Οι εντολές για την λήψη των αποτελεσμάτων των 7 ερωτημάτων είναι οι εξής:

Query1

SELECT BASE.NAME, BASE.STATE, BASE.STARS FROM USER02.YELPBUSINESS WHERE BASE.ISOPEN='1' LIMIT 1000;

Query2

SELECT BASE.NAME, BASE.ADDRESS, BASE.CITY, BASE.REVIEWCOUNT FROM USER02.YELPBUSINESS WHERE BASE.CATEGORIES='Drugstores' ORDER BY TO_NUMBER(BASE.REVIEWCOUNT) DESC;

Query3

SELECT SUM(TO_NUMBER(BASE.REVIEWCOUNT)) AS REVIEWS, BASE.CATEGORIES FROM USER02.YELPBUSINESS WHERE HOURS.MONDAY!='None' AND HOURS.THURSDAY!='None' AND HOURS.WEDNESDAY!='None' AND HOURS.TUESDAY!='None' AND HOURS.FRIDAY!='None' AND HOURS.SATURDAY!='None' AND HOURS.SUNDAY!='None' AND BASE.ISOPEN='1' AND ATTRIBUTES.OPEN24HOURS='True' GROUP BY BASE.CATEGORIES;

Query4

SELECT COUNT() AS COMPANIES,BASE.STATE FROM USER02.YELPBUSINESS WHERE ATTRIBUTES.SMOKING='False' AND HOURS.SUNDAY!='None' GROUP BY BASE.STATE;*

Query5

```
SELECT PERHOUR.WEEKDAY, PERHOUR.HOUR, SUM(TO_NUMBER(PERHOUR.CHECKINS)) AS  
CHECKINS FROM USER02.YELPCHECKIN WHERE TO_NUMBER(PERHOUR.CHECKINS)>=0 GROUP  
BY PERHOUR.WEEKDAY, PERHOUR.HOUR;
```

Query6

```
SELECT BASE.CATEGORIES AS CATEGORIES, SUM(TO_NUMBER(PERHOUR.CHECKINS)) AS  
CHECKINS  
FROM USER02.YELPCHECKIN AS C  
INNER JOIN USER02.YELPBUSINESS AS B  
ON B.ROWKEY = C.BUSINESSID  
WHERE PERHOUR.WEEKDAY!='Sat' AND PERHOUR.WEEKDAY!='Sun' AND PERHOUR.HOUR  
BETWEEN '14:00' AND '16:00' AND BASE.ISOPEN = '1'  
GROUP BY BASE.CATEGORIES;
```

Query7

```
SELECT /*+ INDEX(USER02.YELPBUSINESS Q7B) */ BASE.NAME AS NAME, BASE.NEIGHBORHOOD  
AS NEIGHBORHOOD, BASE.ADDRESS AS ADDRESS, BASE.CITY AS CITY, BASE.STATE AS STATE,  
BASE.POSTALCODE AS POSTALCODE, BASE.LATITUDE AS LATITUDE, BASE.LONGITUDE AS  
LONGITUDE, BASE.STARS AS STARS, BASE.REVIEWCOUNT AS REVIEWCOUNT, BASE.ISOPEN AS  
ISOPEN, BASE.CATEGORIES AS CATEGORIES, PERHOUR.CHECKINS AS CHECKINS  
FROM USER02.YELPCHECKIN AS C  
INNER JOIN USER02.YELPBUSINESS AS B  
ON B.ROWKEY=C.BUSINESSID  
WHERE PERHOUR.WEEKDAY='Sat' ORDER BY TO_NUMBER(PERHOUR.CHECKINS) DESC LIMIT  
100;
```