Graded Project

Impala

Week 9

Table of Content

Business Requirement

Imagine that you are working with one of the largest gaming companies in the world. Your manager asks you to analyses the data from the game to get some more insights. The game that we are talking about is Pokemon Go. Pokémon Go is a free-to-play, location-based augmented reality game developed by Niantic for iOS and Android devices. It was released only in July 2016 and only in selected countries. You can download Pokémon for free of cost and start playing. You can also use PokéCoins to purchase Pokéballs, the in-game item you need to be able to catch Pokémon.

Data Set Description

The dataset consists of 11 columns and their respective description is as follows:

Pokemonid_Number: This column represents id of each Pokémon.

Name: This column represents the name of the Pokémon.

Type 1: This column represents the property of a Pokémon.

Type 2: This column represents the extended property of the same Pokémon.

A Pokémon may be one or both the types. For instance, Charmander is a Fire type, while Bulbasaur is both a Grass type as well as a Poison type. With the current 18-type system, there are 324 possible ways to assign these types to Pokémon, along with 171 unique combinations. As of Generation VI, 133 different type combinations have been used.

Total: This column represents the sum of all character points of a Pokémon (HP, Attack, Defense, Sp. Atk, Sp. Def, and Speed).

HP (Hit Points): This column represents Pokémon Hit Points, which is a value that determines how much damage a Pokémon can receive. When a Pokémon’s HP is down to ‘0’, the Pokémon will faint. HP is the most frequently affected stat of them all, as a depleting HP is a key factor in winning a battle.

Attack: This column represents the Attack stat.

Defense: This column represents the Defense stat.

Sp. Atk: This column represents a Pokémon’s Special Attack stat.

Sp. Def: This column represents a Pokémon’s Special Defense stat.

Speed: This column represents the speed stat of a Pokémon.

Learning Outcomes

After successfully completing the project, the participants will be able to

- Use Impala as a SQL tool for analysing Big Data
- Get understanding about writing queries using Impala

- Approach a business problem and model the solution

Grading Criteria

Participants can use hive shell to explore the problem and find the solution, since the queries of Hive and Impala are the same. Connect with a hive shell and perform the following analysis

Load Data Into HDFS

The first step is to create a folder and upload data into HDFS

On the CloudX Lab web console:

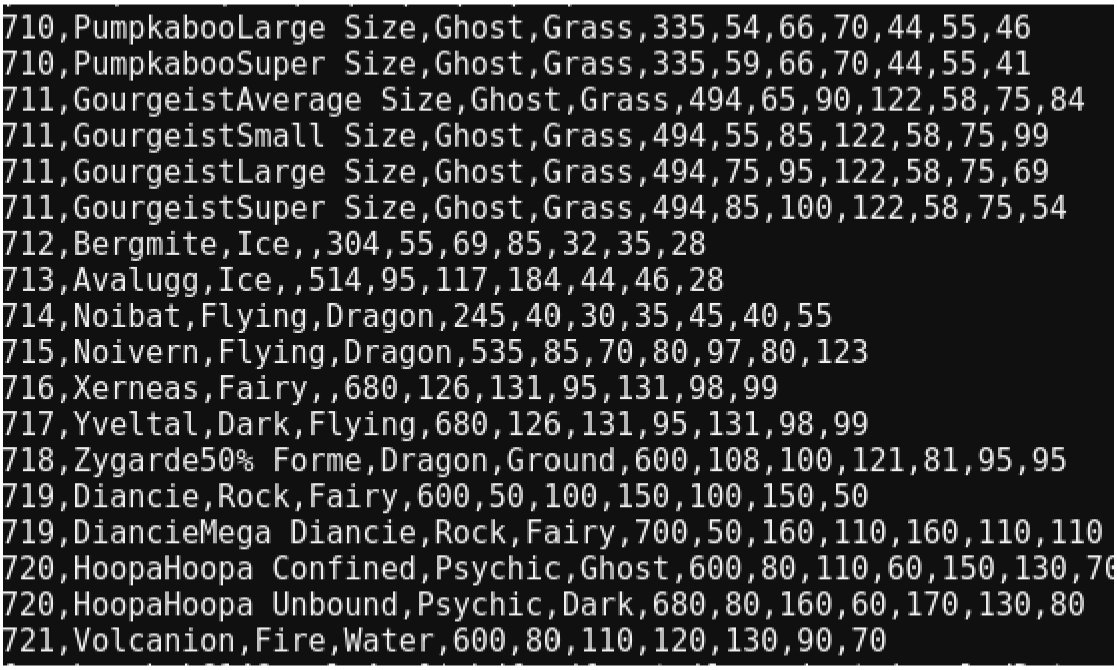
ls

hdfs dfs -ls

hdfs dfs -mkdir project-impala

hdfs dfs -put Dataset-Impala-Project.csv project-impala

hdfs dfs -tail project-impala/Dataset-Impala-Project.csv



```
710,PumpkabooLarge Size,Ghost,Grass,335,54,66,70,44,55,46
710,PumpkabooSuper Size,Ghost,Grass,335,59,66,70,44,55,41
711,GourgeistAverage Size,Ghost,Grass,494,65,90,122,58,75,84
711,GourgeistSmall Size,Ghost,Grass,494,55,85,122,58,75,99
711,GourgeistLarge Size,Ghost,Grass,494,75,95,122,58,75,69
711,GourgeistSuper Size,Ghost,Grass,494,85,100,122,58,75,54
712,Bergmite,Ice,,304,55,69,85,32,35,28
713,Avalugg,Ice,,514,95,117,184,44,46,28
714,Noibat,Flying,Dragon,245,40,30,35,45,40,55
715,Noivern,Flying,Dragon,535,85,70,80,97,80,123
716,Xerneas,Fairy,,680,126,131,95,131,98,99
717,Yveltal,Dark,Flying,680,126,131,95,131,98,99
718,Zygarde50% Forme,Dragon,Ground,600,108,100,121,81,95,95
719,Diancie,Rock,Fairy,600,50,100,150,100,150,50
719,DiancieMega Diancie,Rock,Fairy,700,50,160,110,160,110,110
720,HoopaHoopa Confined,Psychic,Ghost,600,80,110,60,150,130,7
720,HoopaHoopa Unbound,Psychic,Dark,680,80,160,60,170,130,80
721,Volcanion,Fire,Water,600,80,110,120,130,90,70
```

1. **Create a Database and use the same for analysis. Create a Table named pokemon and Load the data to table. Verify that the data has been loaded.**

Create database:
create database project2;
use project2;

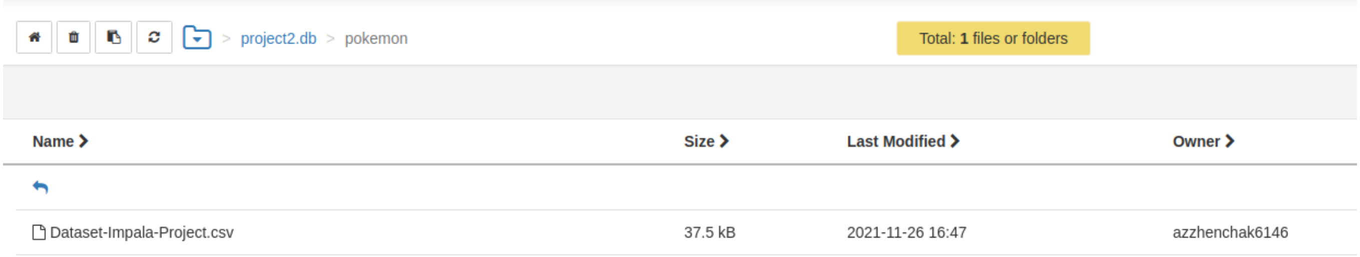
Create table pokemon:

create table if not exists pokemon (pokemonid_number int, name string, type1 string, type2 string, total int, hp int, attack int, defense int, sp_atk int, sp_def int, speed int)
row format delimited
fields terminated by ","
stored as textfile;

Load the data into the table:

load data inpath 'project-impala/Dataset-Impala-Project.csv' overwrite into table pokemon;

Check data in Ambari cloudxlab:



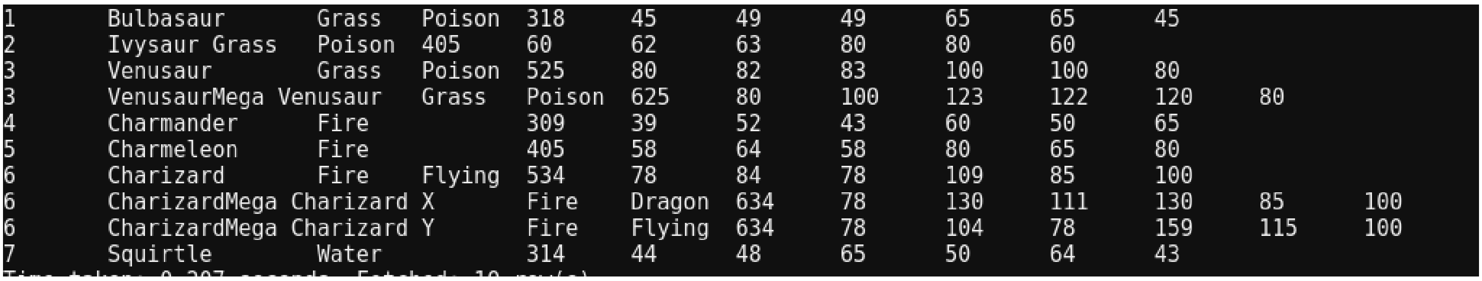
Name	Size	Last Modified	Owner
Dataset-Impala-Project.csv	37.5 kB	2021-11-26 16:47	azzhenchak6146

Verify that the data has been loaded:

Show first 10 rows:

select * from pokemon limit 10;

Output



```
1       Bulbasaur       Grass   Poison  318     45      49      49      65      65      45
2       Ivysaur Grass   Poison  405     60      62      63      80      80      60
3       Venusaur        Grass   Poison  525     80      82      83      100     100     80
3       VenusaurMega Venusaur   Grass   Poison  625     80      100     123     122     120     80
4       Charmander      Fire            309     39      52      43      60      50      65
5       Charmeleon      Fire            405     58      64      58      80      65      80
6       Charizard       Fire    Flying  534     78      84      78      109     85      100
6       CharizardMega Charizard X       Fire    Dragon  634     78      130     111     130     85      100
6       CharizardMega Charizard Y       Fire    Flying  634     78      104     78      159     115     100
7       Squirtle        Water           314     44      48      65      50      64      43
Time taken: 0.207 seconds, Fetched: 10 row(s)
```

Show number of rows in pokemon table:

select count(*) from pokemon;

Output:

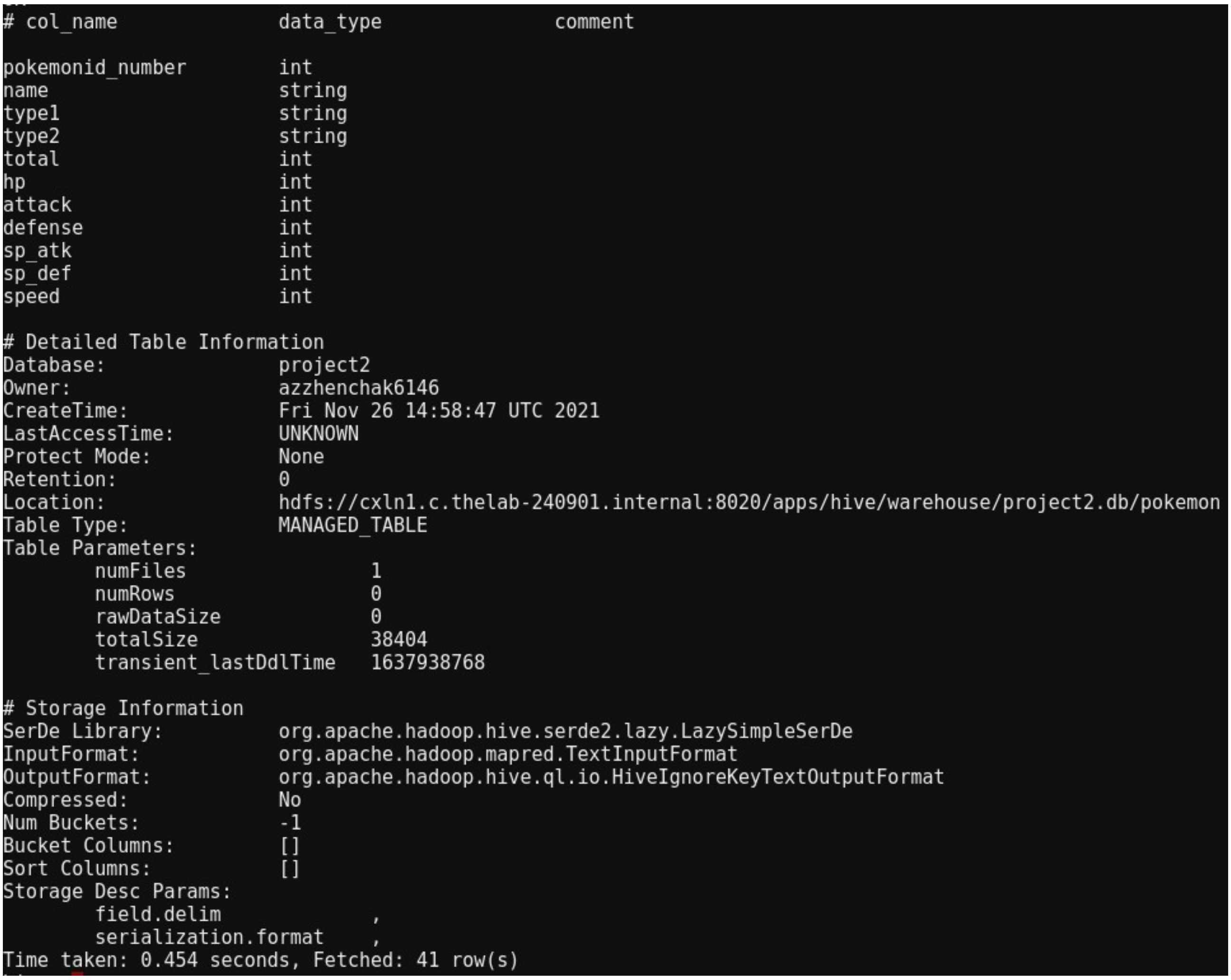
800

select count(distinct name) from pokemon;

Output:

800

There are 800 different pokemon in the dataset describe formatted pokemon;



```
# col_name              data_type               comment

pokemonid_number        int
name                    string
type1                   string
type2                   string
total                   int
hp                      int
attack                  int
defense                 int
sp_atk                  int
sp_def                  int
speed                   int

# Detailed Table Information
Database:               project2
Owner:                  azzhenchak6146
CreateTime:             Fri Nov 26 14:58:47 UTC 2021
LastAccessTime:         UNKNOWN
Protect Mode:           None
Retention:              0
Location:               hdfs://cxln1.c.thelab-240901.internal:8020/apps/hive/warehouse/project2.db/pokemon
Table Type:             MANAGED_TABLE
Table Parameters:
        numFiles                1
        numRows                 0
        rawDataSize             0
        totalSize               38404
        transient_lastDdlTime   1637938768

# Storage Information
SerDe Library:          org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe
InputFormat:            org.apache.hadoop.mapred.TextInputFormat
OutputFormat:           org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat
Compressed:             No
Num Buckets:            -1
Bucket Columns:         []
Sort Columns:           []
Storage Desc Params:
        field.delim             ,
        serialization.format    ,
Time taken: 0.454 seconds, Fetched: 41 row(s)
```

2. Find out the average HP (Hit points) of all the Pokémon

Average HP of all the pokemon

select avg(hp) from pokemon;

Output:

69.25875

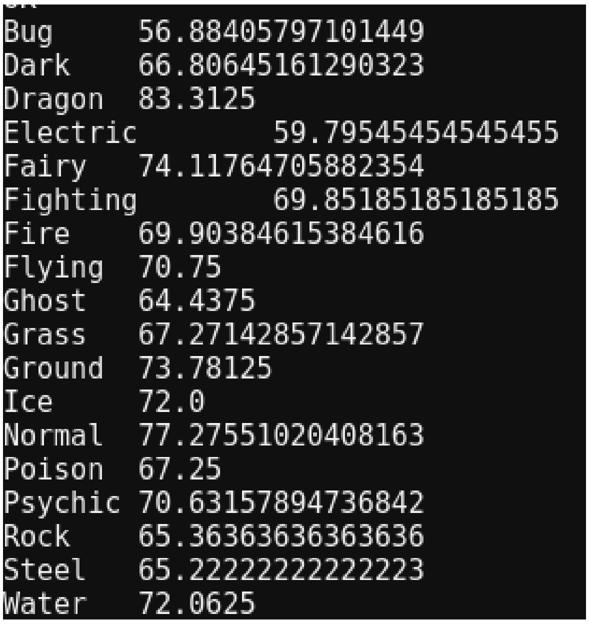
Average Hit point of the Pokémon is 69.25875

Let`s see what is the average hp in each group with same type1

Average HP of pokemons grouped by type1

select type1, avg(hp) from pokemon group by type1

Output:



```
Bug     56.88405797101449
Dark    66.80645161290323
Dragon  83.3125
Electric        59.79545454545455
Fairy   74.11764705882354
Fighting        69.85185185185185
Fire    69.90384615384616
Flying  70.75
Ghost   64.4375
Grass   67.27142857142857
Ground  73.78125
Ice     72.0
Normal  77.27551020408163
Poison  67.25
Psychic 70.63157894736842
Rock    65.36363636363636
Steel   65.22222222222223
Water   72.0625
```

Dragon Pokemons have the highest average score among other types (type1).

3. Create and insert values of existing table ‘pokemon’ into a new table ‘pokemon1’, with an additional column ‘power_rate’ to find the count of ‘powerful’ and ‘moderate’ from the table ‘pokemon1’

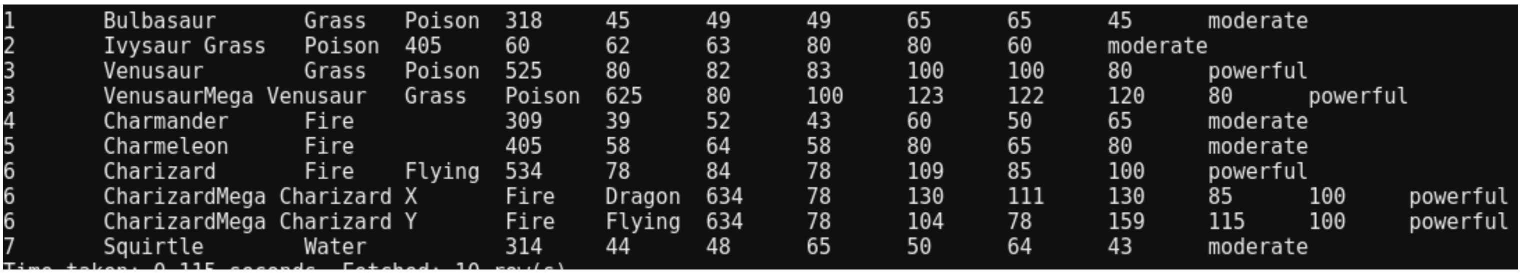
We will create additional column pokemon_rate baced on average hp. So, pokemons which have hp greater than average are considered as powerful and other are moderate.

create table if not exists pokemon1 as select *, IF(hp>=69.25875,'powerful',IF(hp<69.25875,'moderate', '')) AS power_rate from pokemon;

First 10 rows in pokemon1 table

select * from pokemon1 limit 10;

Output:

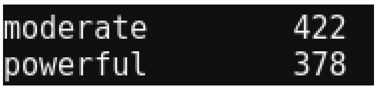


```
1       Bulbasaur       Grass   Poison  318     45      49      49      65      65      45      moderate
2       Ivysaur Grass   Poison  405     60      62      63      80      80      60      moderate
3       Venusaur        Grass   Poison  525     80      82      83      100     100     80      powerful
3       VenusaurMega Venusaur   Grass   Poison  625     80      100     123     122     120     80      powerful
4       Charmander      Fire            309     39      52      43      60      50      65      moderate
5       Charmeleon      Fire            405     58      64      58      80      65      80      moderate
6       Charizard       Fire    Flying  534     78      84      78      109     85      100     powerful
6       CharizardMega Charizard X       Fire    Dragon  634     78      130     111     130     85      100     powerful
6       CharizardMega Charizard Y       Fire    Flying  634     78      104     78      159     115     100     powerful
7       Squirtle        Water           314     44      48      65      50      64      43      moderate
Time taken: 0.115 seconds, Fetched: 10 row(s)
```

4. Find out the number of powerful and moderate HP Pokémons present

select power_rate, count(*) from pokemon1 group by power_rate;

Output:



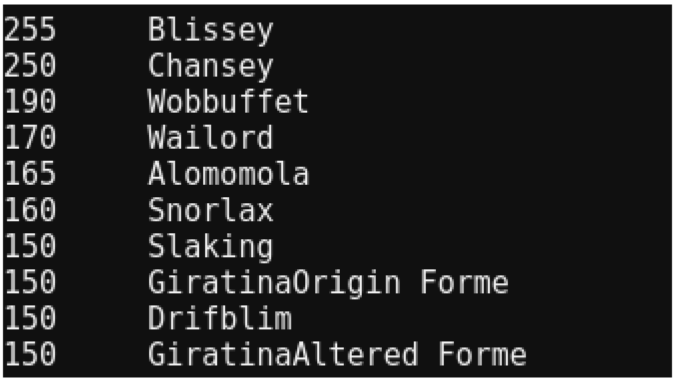
```
moderate        422
powerful        378
```

As a result we have 422 pokemons with moderate rate and 378 pokemons with powerful rate

5. Find out the top 10 Pokémons according to their HP’s

select hp, name from pokemon order by hp desc limit 10;

Output:



```
255     Blissey
250     Chansey
190     Wobbuffet
170     Wailord
165     Alomomola
160     Snorlax
150     Slaking
150     GiratinaOrigin Forme
150     Drifblim
150     GiratinaAltered Forme
```

In the above screenshot there is printed top 10 Pokemons according to their HP’s