**PROJECT DOCUMENTATION**

* **Chinmay Suresh**

**OBJECTIVE**

Main objective of this project is to create an end-to-end data pipeline and analyze data based on database of employees from 1980s-1995s belonging to a big corporation.

**Data Description**

1. **Employees (employees.csv):**

|  |  |  |
| --- | --- | --- |
| **Variable** | **Description** | **Type** |
| emp\_no | Employee Id | Integer |
| emp\_titles\_id | designation id | Character |
| birth\_date | Date of Birth | Date Time |
| first\_name | First Name | Character |
| last\_name | Last Name | Character |
| sex | Gender | Character |
| hire\_date | Employee Hire date | Date Time |
| no\_of\_projects | No. of projects worked on | Integer |
| left | Employee left the organization | Boolean |
| Last\_date | Last date of employment (Exit Date) | Date Time |

1. **Titles (titles.csv):**

title\_id – Unique id of type of employee (designation id) – Character – Not Null

title – Designation – Character – Not Null

1. **Salaries:**

emp\_no – Employee id – Integer – Not Null

salary – Employee’s Salary – Integer – Not Null

1. **Departments (departments.csv):**

dept\_no – Unique id for each dept. – Character – Not Null

dept\_name – Department Name – Character – Not null

1. **Department Managers (dept\_manager.csv):**

dept\_no – Unique id for each dept. – Character – Not Null

emp\_no – Employee no. (head of the dept.) – Integer – Not Null

1. **Department Employees (dept\_emp.csv):**

emp\_no – Employee id – Integer – Not Null

dept\_no – Unique id for each dept. – Character – Not Null

**Technology Stack**

Technologies:

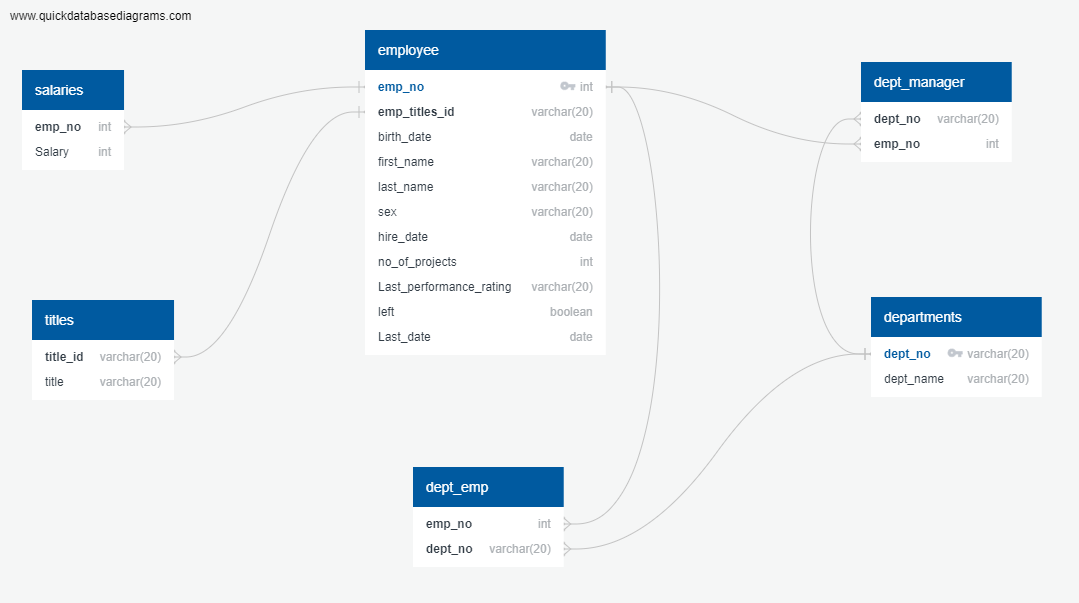
1. Linux
2. RDBMS (MySQL)
3. Sqoop (Transfer data from RDBMS to HDFS)
4. HDFS (Stores the data)
5. Hive (Create database and tables, load data into tables)
6. Impala (Performing EDA)
7. SparkSQL (Performing EDA)
8. SparkML (Model Building)

**Steps:**

1. Data Ingestion which involves receiving data from Created Tables in RDBMS (MySQL) to and storing in HDFS as AVRO files using SQOOP.
2. Create database and tables in Hive, and load AVRO files into the tables.

hive -f capstone1.hql

1. Establish connection between PySpark and Hive.
2. Data Analysis with Impala (ImpalaSQL) and PySpark (Spark SQL) on Hive tables.
3. Data preparation for model building.
4. Model Building in PySpark to predict whether a employee will leave or not. Then create a ML Pipeline encompassing all the process/stages which includes the transformation processes i.e, String Indexing, Vector Assembling, Classifier
5. Creating an end-to-end pipeline

**Entity Relationship Diagram (ERD Data Model)**

**Architecture of pipeline (stages)**

SOURCE(RDBMS:MySQL)

SQOOP

DISTRIBUTED STORAGE SYSTEM (HDFS)

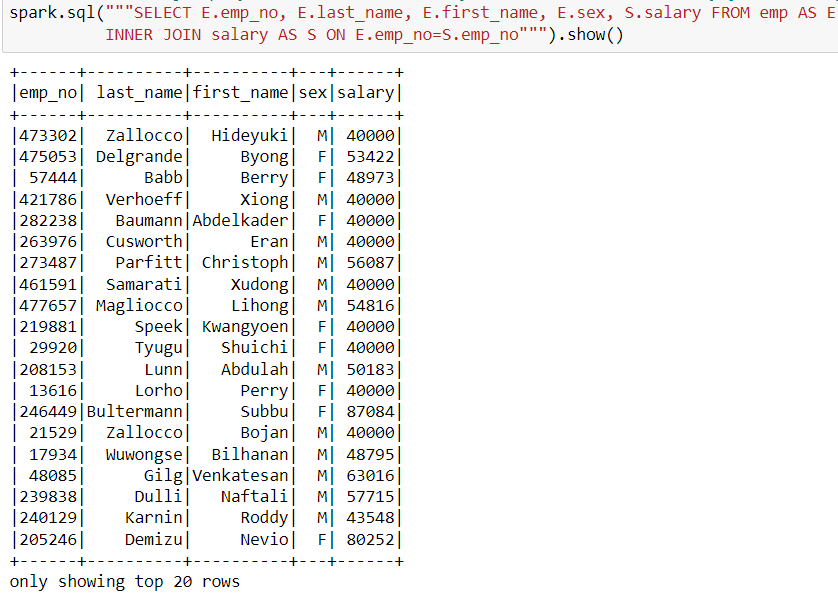
DATA WAREHOUSE (HIVE - hql)

PROCESSING FRAMEWORK (SPARK, SparkSQL)

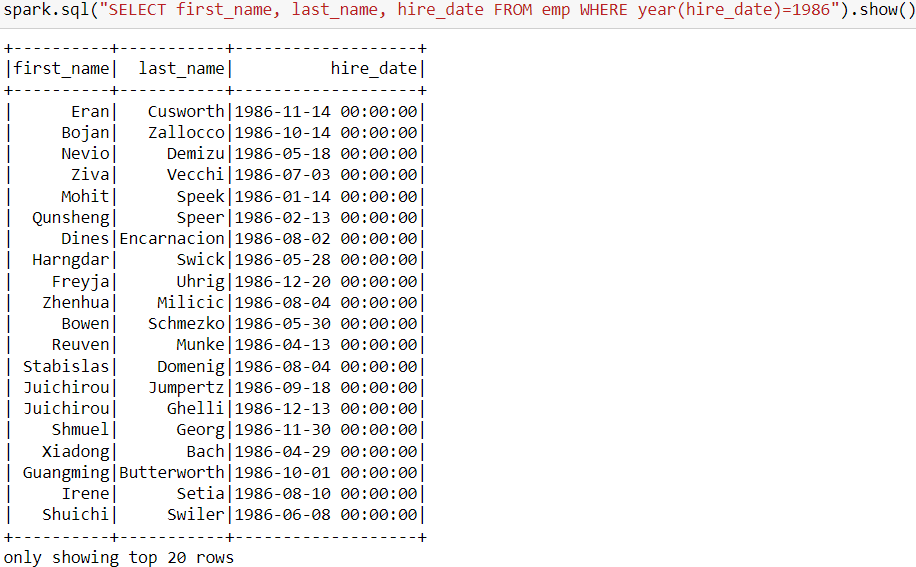
MODEL (SparkML)

**Exploratory Data Analysis**

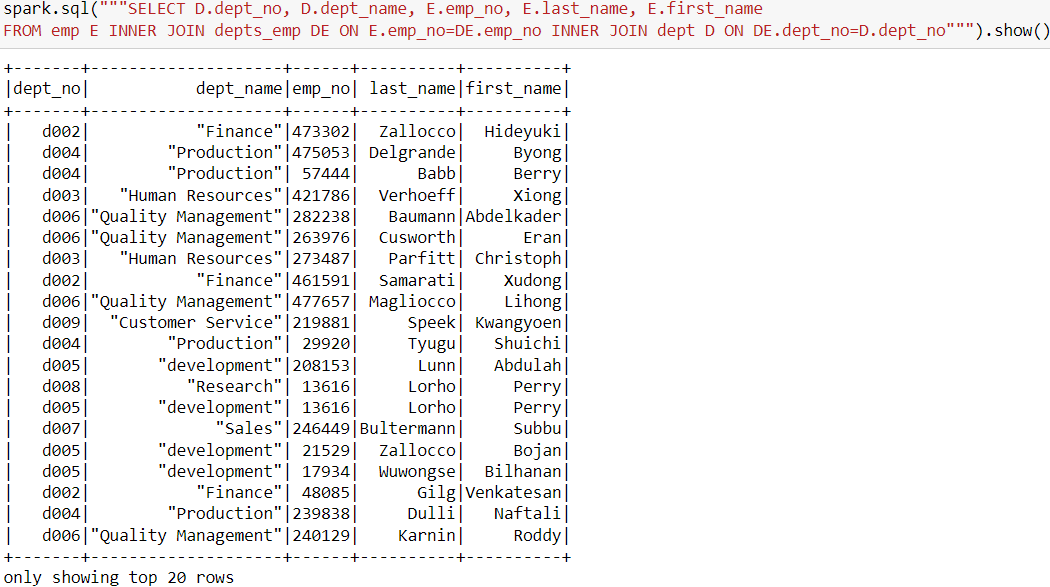
1. **A list showing employee number, last name, first name, sex, and salary for each employee.**

**Ans:**

1. **A list showing first name, last name, and hire date for employees who were hired in 1986.**

**Ans:**

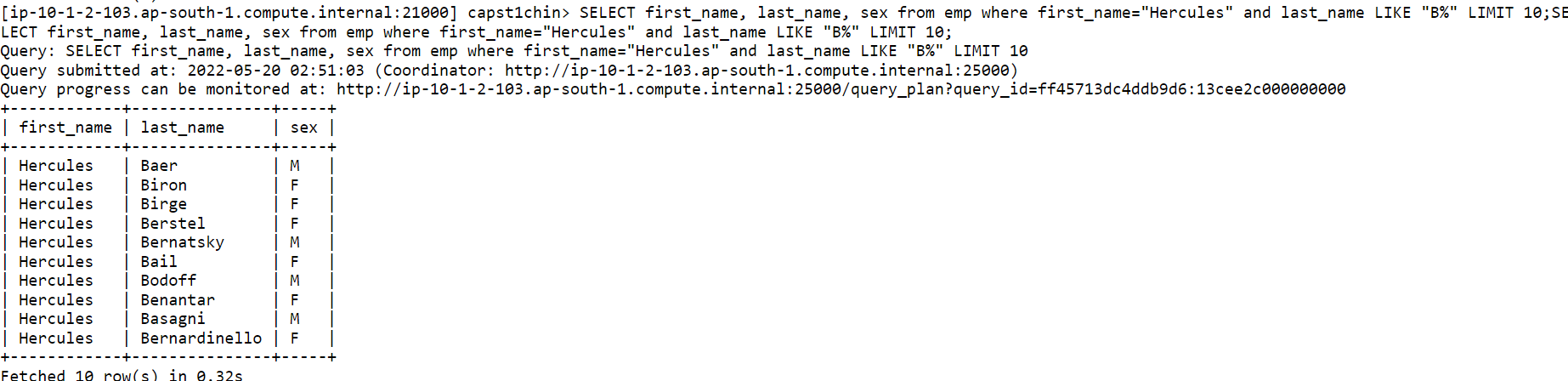
1. **A list showing the manager of each department with the following information: department number, department name, the manager's employee number, last name, first name.**

**Ans:**

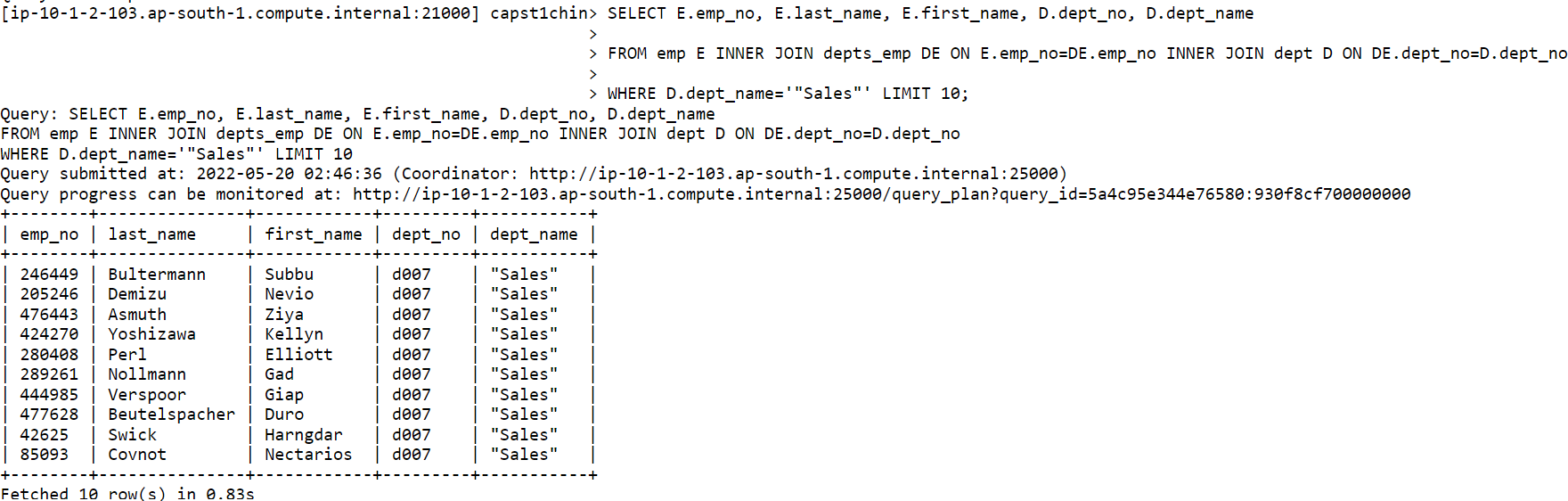
1. **A list showing the department of each employee with the following information: employee number, last name, first name, and department name.**

**Ans:**

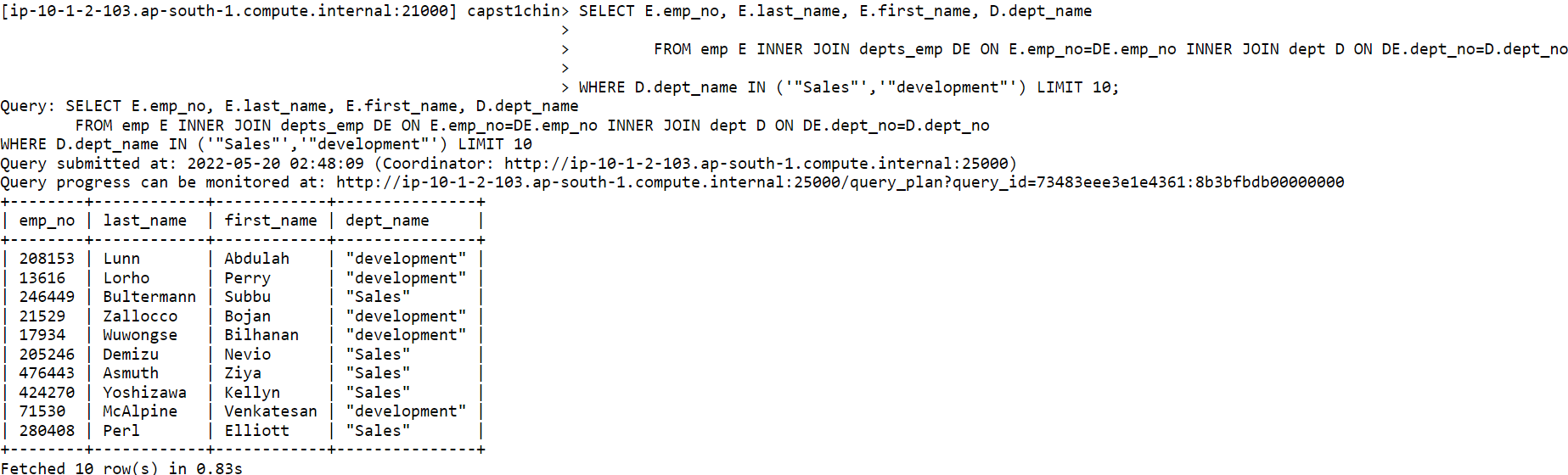
1. **A list showing first name, last name, and sex for employees whose first name is "Hercules" and last names begin with "B.“**

**Ans:**

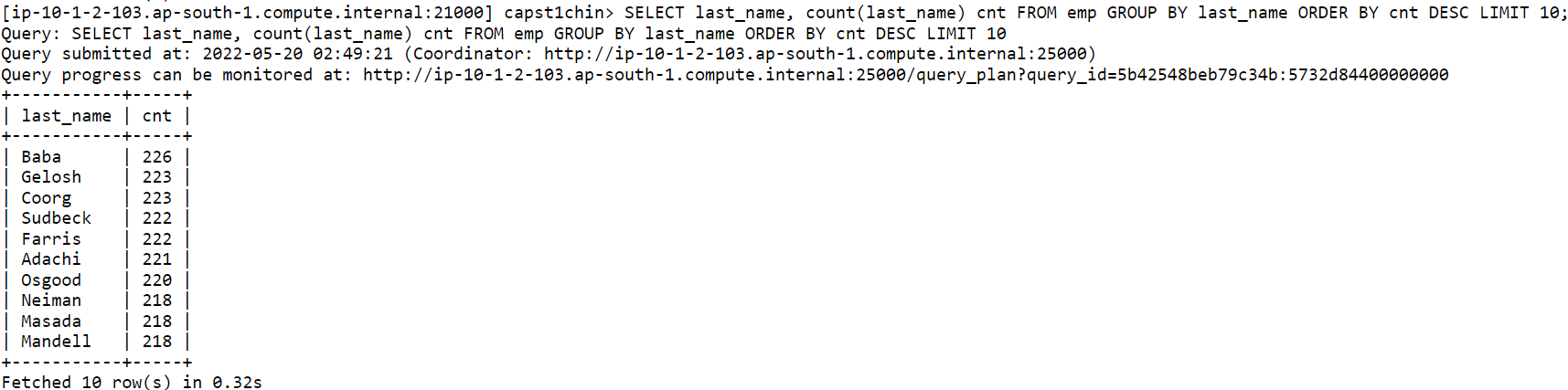
1. **A list showing all employees in the Sales department, including their employee number, last name, first name, and department name.**

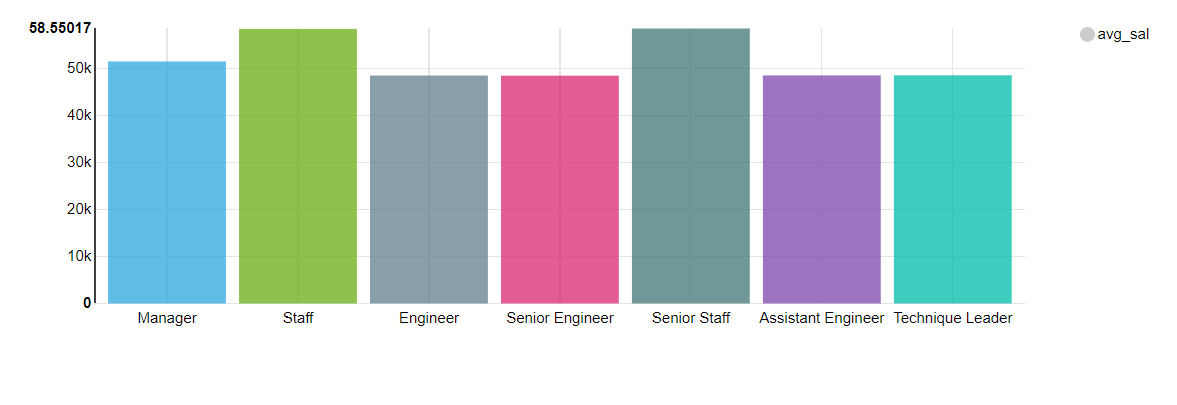
**Ans:**

1. **A list showing all employees in the Sales and Development departments, including their employee number, last name, first name, and department name.**

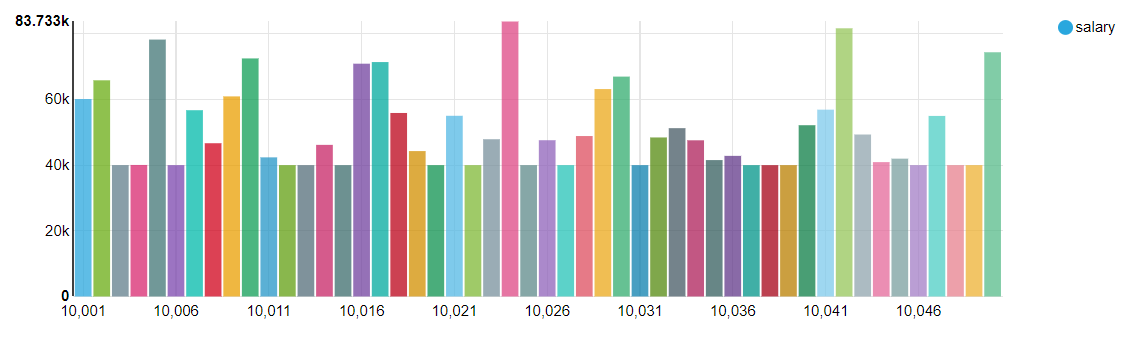
**Ans:**

1. **A list showing the frequency count of employee last names, in descending order. ( i.e., how many employees share each last name.**

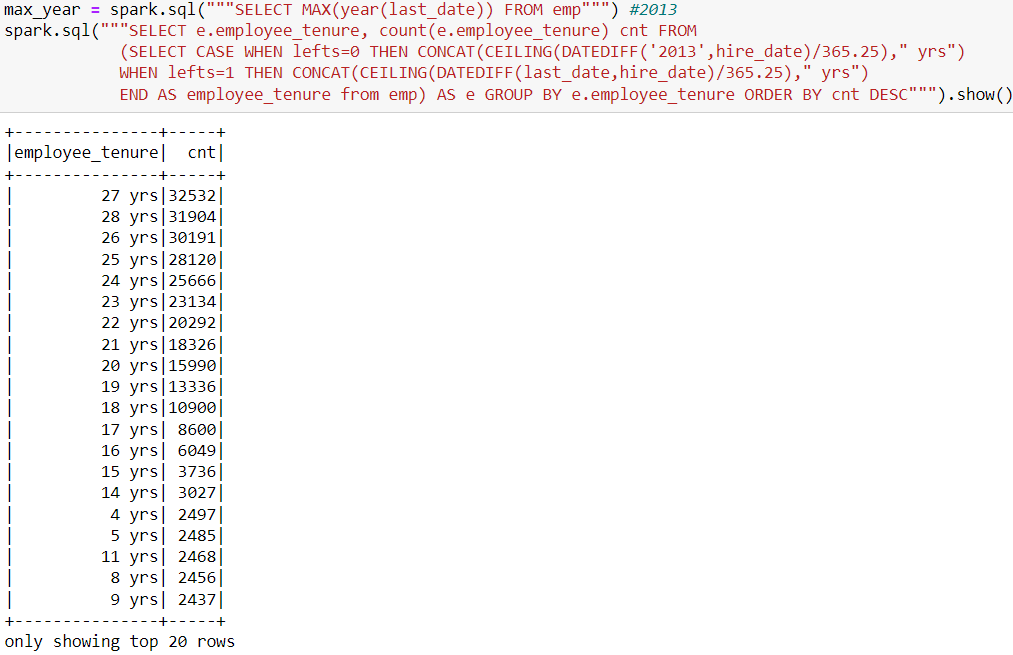
**Ans:**

1. **Bar graph to show the Average salary per title (designation)**

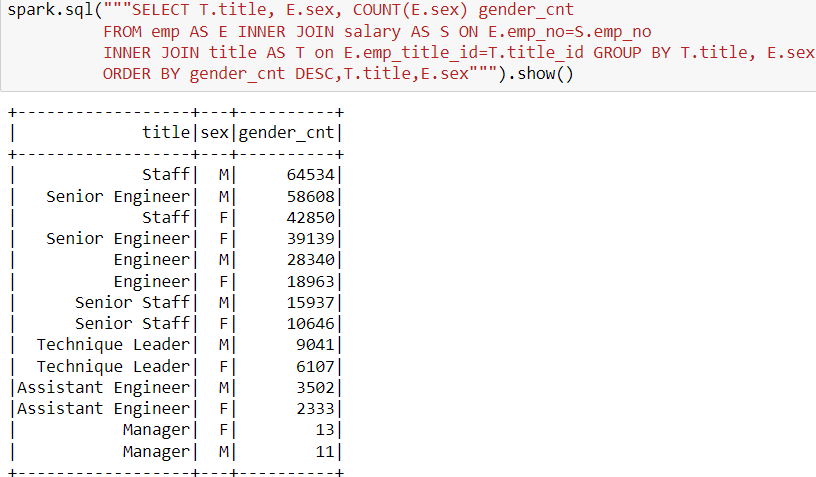
We can observe from the above bar graph thatSenior Staff and Staff have the highest average salary followed by Managers.

1. **Histogram to show the salary distribution among the employees.**
2. **Calculate employee tenure & show the tenure distribution among the employees**

**Ans:**

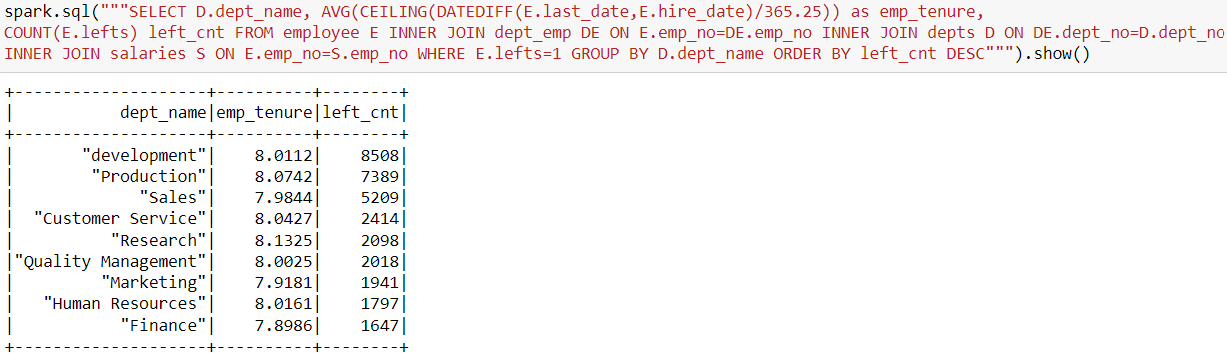
****

We can observe that most of the employees has 25+ years of tenure, whereas most of the employees who left had a shorter tenure of below 10 years.

1.  Gender-wise count by Designation.

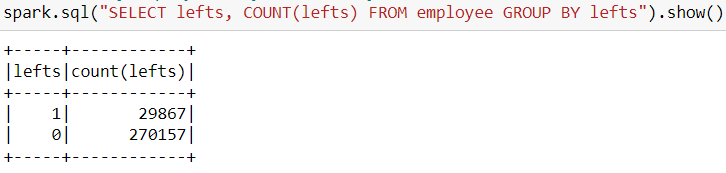
Staff designation has the highest number of both Male and Female workers whereas Manager has the least number of Male and Female workers.

1. **count and average tenure (in yrs) of employees who left by department**

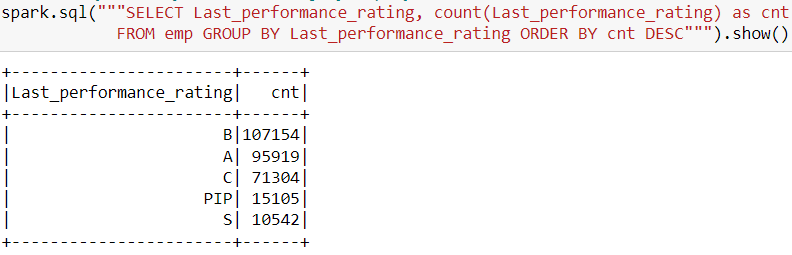
**Ans:**

We can observe that the count of employees who left is the maximum from Development department, whereas the least is from finance. Meanwhile average tenures of employees who left is around 8 years by each department.

1. **Count of employees who left vs stayed.**

****

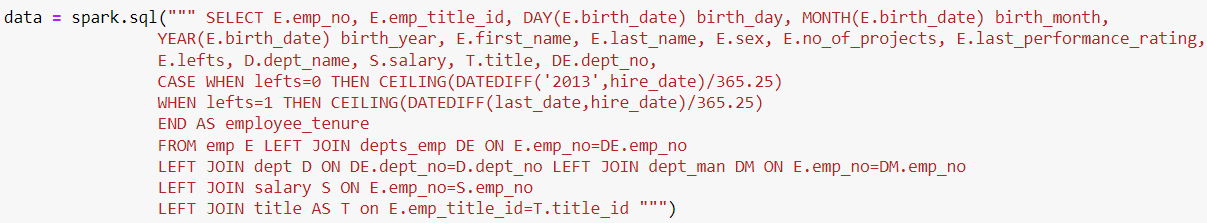
As we can notice, the employees who stayed are in much larger numbers than the employees who left.

1. **Last year performance ratings of employees.**

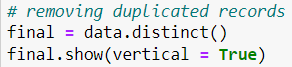
We can observe that Employees with average performance of B are present at maximum. Employees having excellent performance of S are present in less numbers.

**ML Model:**

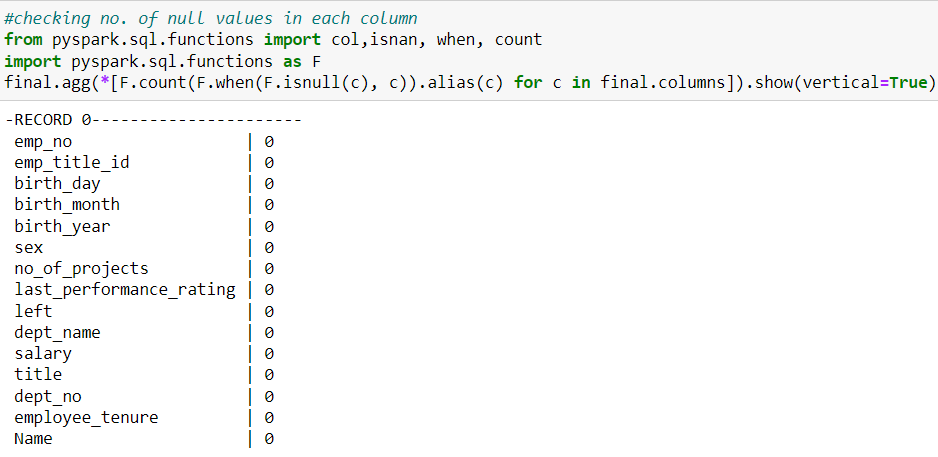
Selecting all the required columns from all the tables for further processes:



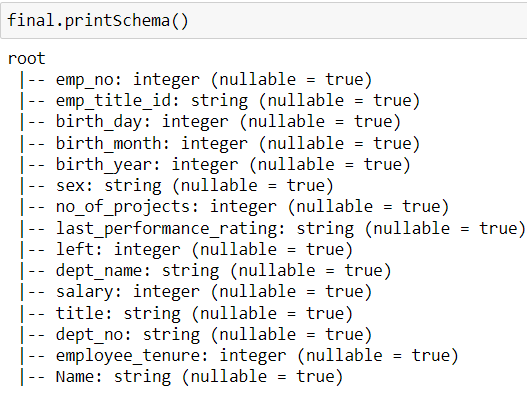
1. **Feature Engineering:**

****Removing duplicate records:

Checking for null values:

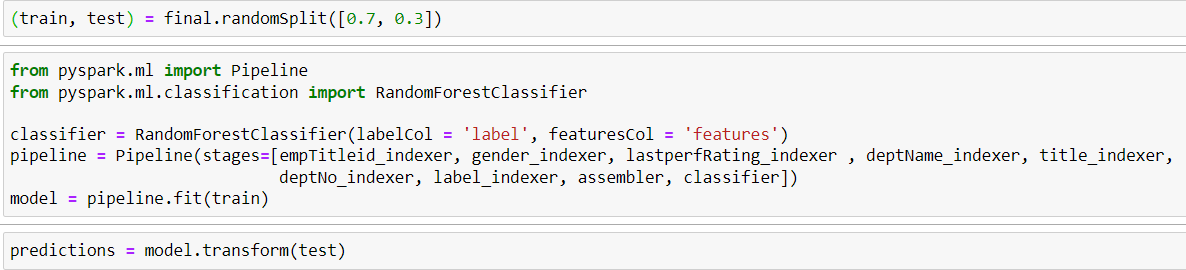
****

Schema:



String Indexing and Vector Assembling:

**ML Pipeline [Stages: String Indexer(s) + Label Indexer + Assembler+ ClassifierModel (RandomForest)]:**

****

Model Validation:

Save the Model:

**Challenges:**

1. Choosing the appropriate data format for data transfer from RDBMS to HDFS: I ended up choosing the AVRO format as it posed less problems and gave good performance.
2. Trying to find the connection of pyspark with hive was time-consuming. I finally managed to import the right modules to connect to the hive metastore.
3. Transforming data after joining all the tables: There were lot of duplicate records which had to be removed for further stage of analysis.
4. Choosing the right model for the prediction and building a pipeline.

**Next Steps:**

Monitoring of pipeline is an important aspect after creation of a pipeline, as it keeps the pipeline operational, capable of extracting and loading the data and also helps maintain data integrity in the process.

This way the data flows from source to destination can be easily accessible, and meaningful to the end-users.