|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования **«МИРЭА − Российский технологический университет»**  **РТУ МИРЭА** |

**Институт информационных технологий (ИИТ)**

**Кафедра прикладной математики (ПМ)**

**ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ**

по дисциплине «Системы управления данными»

**Практическое занятие № 10**

|  |  |  |  |
| --- | --- | --- | --- |
| Студент группы ИНБО-01-17 | *ИМБО-02-22, Ким Кирилл Сергеевич* | (подпись) | |
| Преподаватель | *Алексеева Екатерина Сергеевна, преподаватель* | (подпись) | |
| Отчет представлен | «\_\_\_»\_\_\_\_\_\_\_\_202\_\_г. | |  | |

Москва 2025 г.

1. Практическая работа №10
   1. Цель работы

В этом упражнении данные анализируются с помощью редактора запросов Hue, а различные графики создаются с помощью функций визуализации данных в Hue.

## Получение данных

### Сначала зайдите на сайт kaggle и загрузите набор данных titanic (train.csv, test.csv, gender\_submission.csv). Пожалуйста, перейдите по ссылке ниже. https://www.kaggle.com/c/titanic/data

### Загруженный файл копируется в рабочий каталог с помощью общего каталога virtualbox (этот общий каталог может отличаться в зависимости от ваших настроек). Также можно скачать файл прямо в виртуальной машине.

### Создание таблицы (titanic) и загрузка данных в таблицу в Hue.

#### Запустите браузер Hue. http://localhost:8888

#### Для этого упражнения используйте базу данных mydb.

###### 

#### Если mydb не существует, создайте новую БД и используйте её.

create database mydb;

### Создайте таблицу под названием cars со следующей схемой:

|  |  |
| --- | --- |
| Column Name | Column Type |
| id | Int |
| survived | int |
| pclass | int |
| Name | string |
| sex | string |
| age | int |
| sibsp | int |
| parch | int |
| ticket | string |
| fair | float |
| cabin | string |
| embarked | string |

#### Разделителем поля является запятая, а поле name также содержит запятую, поэтому вам нужно создать таблицу, чтобы игнорировать запятую. Подсказка: используйте OpenCSVSerde в строке ROW FORMAT и

ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde'

WITH SERDEPROPERTIES (

"separatorChar" = ",",

"quoteChar" = "\"")

#### Расположение таблицы: /mywarehouse/titanic с настроенным типом.

###### 

#### Параметр tblproperties ("skip.header.line.count"="1") используется при создании таблицы для исключения заголовочной части данных.

#### 

#### Если вы посмотрите на titanic.csv, вы сможете проверить имя каждого поля в первой строке.

###### 

#### На самом деле, если таблица создается без исключения заголовка, этот параметр можно изменить позже с помощью команды alter.

ALTER TABLE tablename

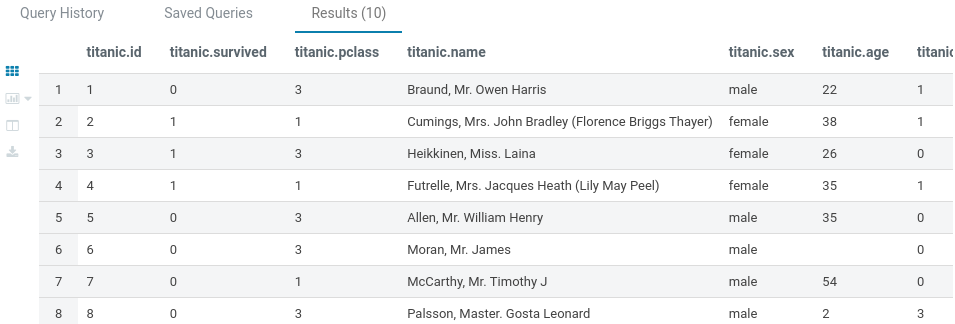
SET TBLPROPERTIES ("skip.header.line.count"="1");

#### Загрузите данные из /home/student/titanic.csv в таблицу titanic

#### 

load data inpath '/user/student/titanic.csv' into table titanic;

###### 



#### Если созданная схема отображается неправильно, выполните следующую команду set.

﻿set hive.serdes.using.metastore.for.schema=

org.apache.hadoop.hive.serde2.OpenCSVSerde;

#### Используйте команду Hive: Create Table As Select (CTAS) для создания таблицы «titanic\_ui» для визуализации. Обязательными столбцами являются:

﻿create table titanic\_ui as select id, survived, pclass, sex, age, ticket, fare, embarked from titanic;

###### 

## Создайте запрос для визуализации

### Введите следующий запрос в окно редактора: найти количество людей по полу, классу и выживанию.

﻿select survived, pclass, sex, count(id) as cnt from titanic\_ui group by survived, pclass, sex order by cnt desc;

### Щелкните стрелку вниз под значком выполнения и выберите “Форматировать”.

###### 

### Запрос переформатируется, как указано выше. Вы можете сохранить запрос, нажав кнопку сохранить.

###### 

### Выберите значок выполнить, чтобы запустить запрос.

###### 

## Создайте столбчатую диаграмму, основанную на результатах запроса:

### Нажмите на значок диаграммы и выберите пункт “Bars”

###### 

### Примечание: ось X = выжившие, ось Y = cnt (количество людей), 0 - погибший, 1 - выживший.

#### Создайте круговую диаграмму, выбрав значок диаграммы и нажав кнопку “Pie”

###### 

#### Создайте столбчатую диаграмму для каждой группы выживших в разбивке по полу.

###### 

Примечание: ось X = пол, ось Y = cnt (количество), группа = выжившие

### Нарисуйте гистограмму выживших и умерших в разбивке по возрасту.

###### 

#### Отрегулируйте ограничение таким образом, чтобы отображалось не более 10 человек, так как на графике возрасты настолько различны, что трудно идентифицировать значения.

###### 

#### Если нужно, загрузите результаты запроса, щелкнув значок загрузки и выбрав, в каком формате вы хотите загрузить, скопировать или экспортировать результаты.

###### 

#### Создайте желаемый запрос и визуализируйте результат.

# Визуализации на географической карте в HUE

В этой лабораторной работе вы узнаете, как создать карту с помощью визуализации Hue. Чтобы использовать функцию Map, у вас должна быть информация, указывающая широту и долготу в данных. Здесь мы используем таблицу “post\_latlon”, которую мы создали в предыдущей лабораторной работе.

## Создайте Maker Maps с помощью post\_latlon

### Для начала, если таблица post\_latlon не существует, создайте ее со следующей схемой и импортируйте данные с помощью sqoop. (Если у вас есть таблица, перейдите к разделу 1.3)

#### Используйте Sqoop для импорта таблицы posts в каталог HDFS /mywarehouse/posts

#### Установите строку --connect в jdbc:mysql://localhost/labs

#### Установите параметры доступа: --username и --password; и то и то - "student"

#### Установите --table для импорта posts

#### Установите такое значение --target-dir, при котором импортированные значения будут сохранены в /mywarehouse/posts

sqoop import \

--connect jdbc:mysql://localhost/labs \

--username student --password student \

--table posts --target-dir /mywarehouse/posts \

--hive-drop-import-delims \

### Создайте таблицу с именем post\_latlon со следующей схемой:

|  |  |
| --- | --- |
| Column Name | Column Type |
| id | Int |
| latitude | float |
| longitude | float |

#### Выполните следующий sql-запрос.

﻿create table post\_latlon (

id int,

latitude float,

longitude float )

row format delimited

fields terminated by ','

location '/mywarehouse/post\_latlon';

### ﻿Сначала создайте запрос для получения значений широты и долготы, затем диаграмму.

﻿select latitude, longitude from post\_latlon where latitude is not null;

###### 

## Создайте столбчатую диаграмму, основанную на результатах запроса

### Щелкните значок диаграммы и выберите “Marker Map”

###### 

### Постройте график, выбрав поля широты (latitude) и долготы (longitude) и тип (type) соответственно. Limit и Label не выбирайте.

###### 

### Вы можете увеличить масштаб области, нажав на увеличительное стекло. Используйте функцию ограничения, чтобы ограничить масштаб только на 25 единиц.

###### 

## Создайте Marker Map с помощью соединения таблиц

### Соедините таблицу posts и таблицу post\_latlon, чтобы создать визуализацию, отображающую заголовок и информацию о широте и долготе.

###### 

Подсказка:

﻿select p.author\_id, p.title, a.latitude, a.longitude from post\_latlon a

join posts p on a.id = p.author\_id;