Cloudera Certified Associate
CCA175 Exam Preparation Series

Part B - Data Analysis Focus

Problems 1-30

Verulam Blue

PROBLEM 1

Records for all the staff working at the US branches of the company CCA175 Exam Success Ltd are stored in the the metastore table ***hr_records*** in the database ***hr_db***.

Due to increased business from the Europe & Asia markets, coupled with the decline in demand from the US market, the company has decided to lay off some of its staff working in the US offices.

Identify all staff that have been working for the company for under 6 months so that management can decide on potential candidates for redundancy in the first wave of redundancies.

- Use a current date of 1st October 2020 (2020-10-01)
- Assume 365 days in a year.

DATA DESCRIPTION

The schema for the data is shown below:

```
|-- name_prefix: string
|-- first_name: string
|-- middle_initial: string
|-- last_name: string
|-- gender: string
|-- date_of_birth: date
|-- date_of_joining: date
|-- salary: integer
|-- last_pct_hike: integer
|-- ssn: string
|-- phone_nbr: string
|-- place_name: string
|-- county: string
|-- city: string
|-- state: string
|-- zip: string
|-- region: string
|-- user_name: string
```

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q1_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q1_soln***
- Results data should be saved as a single gzip compressed parquet file.
- Order results by ***years_since_joining*** in ascending order, followed by ***last_name*** in alphabetical order.

SAMPLE RESULTS

The first few lines of the results table are shown below:

first_name	middle_initial	last_name	date_of_joining	years_since_joining
Fredrick	U	Abad	2020-07-28	0.18
Bree	V	Abe	2020-07-28	0.18
Mira	Q	Abels	2020-07-28	0.18

PROBLEM 2

Using the metastore tables ***items*** and ***lineitems*** from the database ***store_db*** identify how many items with the description ***Channapatna Toy*** have been despatched.

DATA DESCRIPTION

Schema for ***items*** table:

col_name	data_type
itemID	string
description	string
unitcost	float
stocklevel	int

Schema for ***lineitems*** table:

col_name	data_type
orderID	int
itemID	string
quantity	int
despatched	int

OUTPUT REQUIREMENTS

- Results data should be stored as one csv file, in the following HDFS location: ***/verulam_blue/da/30_worked_examples/q2_soln***
- Results should include the following headers:

itemID	description	nbr_despatched

PROBLEM 3

Using the region and ***units_sold*** columns from the ***sales_records*** data, determine the amount of 'Cosmetics' purchased by each region.

- File format of data: parquet
- HDFS location of data: ***/verulam_blue/da/data/sales_records/***

DATA DESCRIPTION

The schema for the data is shown below:

```
|-- region: string
|-- country: string
|-- item_type: string
|-- sales_channel: string
|-- order_priority: string
|-- order_date: date
|-- order_id: long
|-- ship_date: date
|-- units_sold: float
|-- unit_price: float
|-- unit_cost: float
|-- total_revenue: float
|-- total_cost: float
|-- total_profit: float
```

OUTPUT REQUIREMENTS

- Results data should be stored as a single tab delimited text file and compressed using a gzip compression format.
- Save this data in the HDFS location: ***/verulam_blue/da/30_worked_examples/q3_soln***
- The first column should be the **region** and second column the number of cosmetics purchased.

PROBLEM 4

Use the metastore tables ***customers*** and ***orders*** from the database ***store_db*** together with the distinct order numbers to identify how many orders were placed by each customer that made an order.

DATA DESCRIPTION

Schema for ***customers*** table:

```
|-- custID: integer
|-- firstname: string
|-- familyname: string
|-- city: string
|-- country: string
```

Schema for ***orders*** table:

```
|-- orderID: integer
|-- custID: integer
|-- date: date
```

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q4_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q4_soln***
- Results data should be saved as a single gzip compressed parquet file.

Schema for ***q4_soln*** table:

```
|-- custID: integer
|-- firstname: string
|-- familyname: string
|-- nbr_orders: integer
```

- Order the data such that ***nbr_orders*** are in descending order, followed by ***familyname*** in alphabetical order.

SAMPLE RESULTS

The first & first lines of the results table are shown below:

custID	firstname	familyname	nbr_orders
10449	Safia	Hamid	6
.	.	.	.
.	.	.	.
10413	John	smith	1

PROBLEM 5

Use the ***emr*** data to find the total number of emergency department visits, for each month, that were due to influenza-like illness and/or pneumonia together with the total number that resulted in hospitalisation.

Show the ratio of hospitalisations vs visits due to influenza-like illness and/or pneumonia.

- The **extract_date** column represents the date of data extraction. For this question, use the latest extract date = "2020-09-28"
- The column ***ili_pne_visit*** represents the count of influenza-like illness and/or pneumonia visits.
- The column **li_pne_admissions** represent the count of influenza-like illness and/or pneumonia visits that went on to be admitted to the hospital.

- File format of data: parquet
- HDFS location of data: ***/verulam_blue/da/data/emr_data/***

DATA DESCRIPTION

The schema for the data is shown below:

col_name	data_type
extract_date	date
date_of_visit	date
mod_zcta	string
total_ed_visits	int
ili_pne_admissions	int

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q5_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q5_soln***
- Results data should be saved as a single gzip compressed parquet file.
- Order results by ***month*** in ascending order.

SAMPLE RESULTS

The first line of the results table is shown below:

extract_date	month	total_mthly_flu_visits	total_mthly_flu_admissions	proportion_hospitalised
2020-09-28	3	61582	15335	0.249

PROBLEM 6

Using the ***pickup_datetime*** & ***tip_amount*** columns from the ***2018 green taxi*** data identify the total tips made in each month.

Ignore results for which dates were incorrectly input (i.e. ignore nulls).

- File format of data: parquet
- HDFS location of data: ***/verulam_blue/da/data/taxi_data/***

DATA DESCRIPTION

The schema for the data is shown below:

```
|-- vendor_id: integer
|-- pickup_datetime: timestamp
|-- dropoff_datetime: timestamp
|-- store_and_fwd_flag: string
|-- RatecodeID: integer
|-- PULocationID: integer
|-- DOLocationID: integer
|-- passenger_count: integer
|-- trip_distance: float
|-- fare_amount: float
|-- extra: float
|-- mta_tax: float
|-- tip_amount: float
|-- tolls_amount: float
|-- ehail_fee: float
|-- improvement_surcharge: float
|-- total_amount: float
|-- payment_type: integer
|-- trip_type: integer
```

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q6_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q6_soln***
- Results data should be saved as a single csv file.
- Results should have the following schema:

```
|-- month: integer
|-- total_tip_for_month: double
```

PROBLEM 7

Use the metastore tables ***customers*** and ***orders*** from the database ***store_db*** to find out which individuals did not place any orders.

DATA DESCRIPTION

Schema for ***customers*** table:

```
|-- custID: integer
|-- firstname: string
|-- familyname: string
|-- city: string
|-- country: string
```

Schema for ***orders*** table:

```
|-- orderID: integer
|-- custID: integer
|-- date: date
```

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q7_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q7_soln***
- Results data should be saved as a single zlib compressed ORC file.
- Order results by **familyname** in alphabetical order.
- Schema for q7_soln table:

```
|-- custID: integer
|-- firstname: string
|-- familyname: string
```

custID	firstname	familyname
101010	firstname_1	familyname_1
.	.	.
.	.	.
n0n0n0	firstname_n	familyname_n

PROBLEM 8

Use the ***items*** column from the metastore table ***gp_rx*** together with the table ***gp_address*** from the database ***gp_db*** to find the total number of prescriptions made by each GP practice in the city of ***Brighton***.

- The ***items*** column, in the metastore table ***gp_rx***, represents the "total number of items prescribed".
- The first 4 letters of postcodes for GP practices in Brighton are:

  'BN1 ', 'BN2 ', 'BN41', 'BN42', 'BN45', 'BN50', 'BN51', 'BN88'

DATA DESCRIPTION

Schema for ***gp_address*** table:

col_name	data_type
date	string
practice_code	string
surgery_name	string
address_1	string
address_2	string
address_3	string
address_4	string
postcode	string

Schema for ***gp_rx*** table:

col_name	data_type
sha	string
pct	string
practice_code	string
bnf_code	string
bnf_name	string
items	string
nic	string
act_cost	string
quantity	string
period	string

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q8_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q8_soln***
- Results data should be saved as a single gzip compressed parquet file
- Order results by ***practice_code*** in ascending order.

SAMPLE RESULTS

The first line of the results table should look as the table below:

practice_code	surgery_name	nbr_prescriptions
G81006	ARDINGLY COURT SURGERY	16104

PROBLEM 9

Using the 2018 green taxi data determine the maximum total amount charged to passengers in any one day in each month.

Use the ***dropoff_datetime*** & ***total_amount*** columns.

- File format of data: parquet
- HDFS location of data: ***/verulam_blue/da/data/taxi_data/***

DATA DESCRIPTION

The schema for the data is shown below:

```
|-- vendor_id: integer
|-- pickup_datetime: timestamp
|-- dropoff_datetime: timestamp
|-- store_and_fwd_flag: string
|-- RatecodeID: integer
|-- PULocationID: integer
|-- DOLocationID: integer
|-- passenger_count: integer
|-- trip_distance: float
|-- fare_amount: float
|-- extra: float
|-- mta_tax: float
|-- tip_amount: float
|-- tolls_amount: float
|-- ehail_fee: float
|-- improvement_surcharge: float
|-- total_amount: float
|-- payment_type: integer
|-- trip_type: integer
```

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q9_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q9_soln***
- Results data should be saved in two lz4 compressed csv files.
- Ignore results for which dates were incorrectly input (i.e. ignore nulls).
- Results should have the following schema:

```
|-- month: integer
|-- max_amount_charged: float
```

PROBLEM 10

The Wills & Fergie Bank are concerned with credit card fraud.

They consider any one individual owning more than 5 credit cards as being high-risk.

You have been tasked with identifying those individuals that hold more than 5 credit cards.

Use the ***credit_cards*** data to determine these individuals.

- File format of data: parquet
- HDFS location of data: ***/verulam_blue/da/data/credit_cards/***

DATA DESCRIPTION

The schema for the data is shown below:

```
|-- card_type_code: string
|-- card_type_full_name: string
|-- issuing_bank: string
|-- card_number: long
|-- card_holder_name: string
|-- cvv_cvv2: integer
|-- issue_date: string
|-- card_pin: integer
|-- card_limit: long
```

OUTPUT REQUIREMENTS

- Results data should be stored as a single tab delimited text file and compressed using gzip.
- Save this data in the HDFS location: ***/verulam_blue/da/30_worked_examples/q10_soln***

The first column should be the card holders name and the second column the number of credit cards owned.

SAMPLE RESULTS

A sample line from the results table is shown below:

```
|Phyllis Watson 6    |
```

PROBLEM 11

Use the ***emr*** data to find the 3 months that had the highest percentage of emergency department visits that were due to influenza-like illness and/or pneumonia.

The ***extract_date*** column represents the date of data extraction.

For this question, use the latest extract date = "2020-09-28"

- The column ***ili_pne_visit*** represents the count of influenza-like illness and/or pneumonia visits.
- The column ***total_ed_visits*** represents the count of all emergency department visits.

- File format of data: parquet
- HDFS location of data: ***/verulam_blue/da/data/emr_data/***

DATA DESCRIPTION

The schema for the data is shown below:

col_name	data_type
extract_date	date
date_of_visit	date
mod_zcta	string
total_ed_visits	int
ili_pne_admissions	int

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q11_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q11_soln***
- Results data should be saved as a single snappy compressed parquet file.

SAMPLE RESULTS

The first line of the results table should look as the table below:

extract_date	month	pct_visits_to_emr_due_to_flu_symptoms
2020-09-28	4	23.9

PROBLEM 12

Using the 2018 green taxi data, use the sum of the total trip distances travelled for each day to determine, for each month, the 3 days with the overall longest distances travelled, as registered by the taximeter.

- Use the ***dropoff_datetime*** & ***trip_distance*** columns.
- File format of data: parquet
- HDFS location of data***: /verulam_blue/da/data/taxi_data/***

DATA DESCRIPTION

The schema for the data is shown below:

```
|-- vendor_id: integer
|-- pickup_datetime: timestamp
|-- dropoff_datetime: timestamp
|-- store_and_fwd_flag: string
|-- RatecodeID: integer
|-- PULocationID: integer
|-- DOLocationID: integer
|-- passenger_count: integer
|-- trip_distance: float
|-- fare_amount: float
|-- extra: float
|-- mta_tax: float
|-- tip_amount: float
|-- tolls_amount: float
|-- ehail_fee: float
|-- improvement_surcharge: float
|-- total_amount: float
|-- payment_type: integer
|-- trip_type: integer
```

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q12_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q12_soln***
- Results data should be saved as a single snappy compressed parquet file.
- Ignore results for which dates were incorrectly input (i.e. ignore nulls).

SAMPLE RESULTS

The first 4 lines of the results are shown below:

month	day	total_trip_distance
1	27	83811
1	20	83557
1	26	81449
2	24	87243

PROBLEM 13

Use the ***emr*** data to find the modified ZIP codes that saw the most patients visit emergency departments due to influenza-like illness and/or pneumonia.

Identify the top 10 areas.

- The ***extract_date*** column represents the date of data extraction. For this question, use the latest extract date = "2020-09-28"
- The column ***ili_pne_visit*** represents the count of influenza-like illness and/or pneumonia visits.
- The column ***mod_zcta*** represents the modified ZIP Code tabulation area of patient residence.

- File format of data: parquet
- HDFS location of data: ***/verulam_blue/da/data/emr_data/***

DATA DESCRIPTION

The schema for the data is shown below:

col_name	data_type
extract_date	date
date_of_visit	date
mod_zcta	string
total_ed_visits	int
ili_pne_admissions	int

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q13_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q13_soln***
- Results data should be saved as a single uncompressed parquet file.

SAMPLE RESULTS

The first line of the results table should look as the table below:

extract_date	mod_zcta	total_emr_visits_due_to_flu
2020-09-28	11368	3817

PROBLEM 14

Use the ***items*** column from the metastore table ***gp_rx*** together with the ***gp_address*** table from the database ***gp_db*** to find the top 10 most frequently prescribed medications by GP practices in the city of Brighton.

Include in the results the total actual cost of these most frequently prescribed medications (use the ***act_cost*** column)

- The ***items*** column, in the metastore table ***gp_rx***, represents the "total number of items prescribed".

- The first 4 letters of postcodes for GP practices in Brighton are:

  'BN1 ', 'BN2 ', 'BN41', 'BN42', 'BN45', 'BN50', 'BN51', 'BN88'

DATA DESCRIPTION

Schema for ***gp_address*** table:

col_name	data_type
date	string
practice_code	string
surgery_name	string
address_1	string
address_2	string
address_3	string
address_4	string
postcode	string

Schema for ***gp_rx*** table:

col_name	data_type
sha	string
pct	string
practice_code	string
bnf_code	string
bnf_name	string
items	string
nic	string
act_cost	string
quantity	string
period	string

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q14_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q14_soln***
- Results data should be saved as a single snappy compressed avro file.
- Order results by ***nbr_prescriptions*** in descending order.

SAMPLE RESULTS

The first line of the results table should look as the table below:

bnf_code	bnf_name	nbr_prescriptions	total_actual_cost
0103050P0AAAAAA	Omeprazole_Cap E/C 20mg	6005	8652

PROBLEM 15

Use the metastore tables ***customers***, ***items***, ***lineitems*** and ***orders*** from the database ***store_db*** to identify customers that bought a "*Channapatna Toy*" (***itemID*** G9) and/or a "*Carrom Board*" (***itemID*** H5).

DATA DESCRIPTION

Schema for ***customers*** table:

```
|-- custID: integer
|-- firstname: string
|-- familyname: string
|-- city: string
|-- country: string
```

Schema for ***items*** table:

```
|-- itemID: string
|-- description: string
|-- unitcost: float
|-- stocklevel: integer
```

Schema for ***lineitems*** table:

```
|-- orderID: integer
|-- itemID: string
|-- quantity: integer
|-- despatched: integer
```

Schema for ***orders*** table:

```
|-- orderID: integer
|-- custID: integer
|-- date: date
```

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q15_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q15_soln***
- Results data should be saved as a single snappy compressed avro file.
- Schema for q15_soln table:

```
|-- custID: integer
|-- firstname: string
|-- familyname: string
```

PROBLEM 16

The Wills & Fergie Bank are looking to launch their own credit card, w&f credit. They are looking to attract individuals that are on their records as having been issued with a Visa credit card issued by Citibank anytime before 2011.

You have been tasked with identifying these individuals in order to help your team attract new business.

Use the ***issue_date*** column from ***credit_cards*** data to help you.

- File format of data: parquet
- HDFS location of data: ***/verulam_blue/da/data/credit_cards/***

DATA DESCRIPTION

The schema for the data is shown below:

```
|-- card_type_code: string
|-- card_type_full_name: string
|-- issuing_bank: string
|-- card_number: long
|-- card_holder_name: string
|-- cvv_cvv2: integer
|-- issue_date: string
|-- card_pin: integer
|-- card_limit: long
```

OUTPUT REQUIREMENTS

- Results data should be stored as a single pipe delimited text file and compressed using gzip.
- Save this data a in the HDFS location: ***/verulam_blue/da/30_worked_examples/q16_soln***
- The first column should be ***card_holder_name***, the second ***card_type_full_name*** and the third column ***issue_date***.
- Order results by month & year in ascending order, followed by the card holder's name in alphabetical order.

SAMPLE RESULTS

The first and last lines from the results are shown below:

```
|Abel C Coyer|Visa|01/2010      |
    .              .         .
    .              .         .
|Zane E Lam  |Visa|12/2010      |
```

PROBLEM 17

Using the ***sales_records*** data, identify the total profits made from each ***item_type*** in Asia and Europe.

Order the results alphabetically by ***region*** followed by total profit made from each item type.

- File format of data: parquet
- HDFS location of data: ***/verulam_blue/da/data/sales_records/***

DATA DESCRIPTION

The schema for the data is shown below:

```
|-- region: string
|-- country: string
|-- item_type: string
|-- sales_channel: string
|-- order_priority: string
|-- order_date: date
|-- order_id: long
|-- ship_date: date
|-- units_sold: float
|-- unit_price: float
|-- unit_cost: float
|-- total_revenue: float
|-- total_cost: float
|-- total_profit: float
```

OUTPUT REQUIREMENTS

- Results data should be stored as a pipe delimited text file and compressed using deflate.
- Save this data in the HDFS location: ***/verulam_blue/da/30_worked_examples/q17_soln***
- The first column should be the ***region*** and second column ***item_type*** and the third column the total profits made from each item_type

SAMPLE RESULTS

Sample lines from the results file are shown below:

```
|Asia|Cosmetics|21000345306       |
|Asia|Household|20285107547       |
|Asia|Office Supplies|15409357016  |
```

PROBLEM 18

In the May 2018 the highest fare amount was found to be 2126.69

Using the 2018 green taxi data, find the 3rd, 5th and 7th highest fare amounts for that month and complete the table below.

day	fare_amount	rank
30	2126.69	1
x	x	3
x	x	5
x	x	7

- Use the ***dropoff_datetime*** & ***fare_amount*** columns.
- File format of data: parquet
- HDFS location of data***: /verulam_blue/da/data/taxi_data/***

DATA DESCRIPTION

The schema for the data is shown below:

```
|-- vendor_id: integer
|-- pickup_datetime: timestamp
|-- dropoff_datetime: timestamp
|-- store_and_fwd_flag: string
|-- RatecodeID: integer
|-- PULocationID: integer
|-- DOLocationID: integer
|-- passenger_count: integer
|-- trip_distance: float
|-- fare_amount: float
|-- extra: float
|-- mta_tax: float
|-- tip_amount: float
|-- tolls_amount: float
|-- ehail_fee: float
|-- improvement_surcharge: float
|-- total_amount: float
|-- payment_type: integer
|-- trip_type: integer
```

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q18_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q18_soln***
- Results data should be saved as a single gzip compressed parquet file.

- Results should have the following schema:

```
|-- day: integer
|-- fare_amount: float
|-- rank: integer
```

PROBLEM 19

Use the metastore tables ***customers***, ***items***, ***lineitems*** and ***orders*** from the ***database store_db*** to identify customers that bought both a "*Channapatna Toy*" (***itemID*** G9) and "*Carrom Board*" (***itemID*** H5).

DATA DESCRIPTION

Schema for ***customers*** table:

```
|-- custID: integer
|-- firstname: string
|-- familyname: string
|-- city: string
|-- country: string
```

Schema for ***items*** table:

```
|-- itemID: string
|-- description: string
|-- unitcost: float
|-- stocklevel: integer
```

Schema for ***lineitems*** table:

```
|-- orderID: integer
|-- itemID: string
|-- quantity: integer
|-- despatched: integer
```

Schema for ***orders*** table:

```
|-- orderID: integer
|-- custID: integer
|-- date: date
```

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q19_soln*** in the database ***solutions_db***
- Results data should be stored in the custom table path: ***/verulam_blue/da/30_worked_examples/q19_soln***
- The results should also be saved in a json file format.

Schema for q19_soln table:

```
|-- custID: integer
|-- firstname: string
|-- familyname: string
```

PROBLEM 20

The ***act_cost*** and ***items*** columns, in the metastore table ***gp_rx***, represent the "actual cost of prescribed medications" and the "total number of items prescribed", respectively.

Use the expression ***act_cost/items*** from the metastore table ***gp_rx*** together with the ***gp_address*** table from the database ***gp_db*** to find the top 10 most expensive medications prescribed by GP practices in the city of Bolton.

- The first 4 letters of postcodes for GP practices in Bolton are:

  'BL1 ', 'BL2 ', 'BL3 ', 'BL4 ', 'BL5 ', 'BL6 ', 'BL7 '

DATA DESCRIPTION

Schema for ***gp_address*** table:

col_name	data_type
date	string
practice_code	string
surgery_name	string
address_1	string
address_2	string
address_3	string
address_4	string
postcode	string

Schema for ***gp_rx*** table:

col_name	data_type
sha	string
pct	string
practice_code	string
bnf_code	string
bnf_name	string
items	string
nic	string
act_cost	string
quantity	string
period	string

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q20_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q20_soln***
- Results data should be saved as a single bzip2 compressed csv file.
- Order results by ***actual_cost_per_med*** in descending order.

SAMPLE RESULTS

The first line of the results table should look as the table below:

bnf_code	bnf_name	actual_cost_per_med
1001030D0AAACAC	Etanercept_Inj 50mg/ml Pfs	3967

PROBLEM 21

Records of all the staff working for the US branches of the company CCA175 Exam Success Ltd are stored in the the metastore table **hr_records** in the **database hr_db**

Due to increased business from the Europe & Asia markets, coupled with the decline in demand from the US market, the company has decided to lay off some of its staff working in the US offices.

Fearful of being sued for wrongful dismissal management are keen to ensure that there is no bias regarding the region an employee is from and have asked you for a breakdown, by region, of the number of staff that have been working for under 6 months.

- Use a current date of 1st October 2020 (2020-10-01)
- Assume 365 days in a year.

DATA DESCRIPTION

The schema for the data is shown below:

```
|-- name_prefix: string
|-- first_name: string
|-- middle_initial: string
|-- last_name: string
|-- gender: string
|-- date_of_birth: date
|-- date_of_joining: date
|-- salary: integer
|-- last_pct_hike: integer
|-- ssn: string
|-- phone_nbr: string
|-- place_name: string
|-- county: string
|-- city: string
|-- state: string
|-- zip: string
|-- region: string
|-- user_name: string
```

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q21_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q21_soln***
- Results data should be saved as a single snappy compressed parquet file.
- Order results by ***region*** in alphabetical order.

SAMPLE RESULTS

The results table should have the following columns:

region	nbr_employed_for_less_than_6_mths

PROBLEM 22

Using the ***sales_records*** data find out which 5 countries generated the least profits for the supplier.

Order the results by total profit made from each country, with the one generating the least profits at the top.

- File format of data: parquet
- HDFS location of data: ***/verulam_blue/da/data/sales_records/***

DATA DESCRIPTION

The schema for the data is shown below:

```
|-- region: string
|-- country: string
|-- item_type: string
|-- sales_channel: string
|-- order_priority: string
|-- order_date: date
|-- order_id: long
|-- ship_date: date
|-- units_sold: float
|-- unit_price: float
|-- unit_cost: float
|-- total_revenue: float
|-- total_cost: float
|-- total_profit: float
```

OUTPUT REQUIREMENTS

- Results data should be stored as a tab delimited text file and compressed using lz4.
- Save this data in the HDFS location: ***/verulam_blue/da/30_worked_examples/q22_soln***
- The first column should be the ***country*** and second column the ***total_profit***.

SAMPLE RESULTS

Sample line from the results file:

```
|Syria    4164936522  |
```

PROBLEM 23

Using the 2018 green taxi data find the 3 months that saw the most number of passengers picked up.

Use the ***dropoff_datetime*** & ***passenger*** columns.

- File format of data: parquet
- HDFS location of data: ***/verulam_blue/da/data/taxi_data/***

DATA DESCRIPTION

The schema for the data is shown below:

```
|-- vendor_id: integer
|-- pickup_datetime: timestamp
|-- dropoff_datetime: timestamp
|-- store_and_fwd_flag: string
|-- RatecodeID: integer
|-- PULocationID: integer
|-- DOLocationID: integer
|-- passenger_count: integer
|-- trip_distance: float
|-- fare_amount: float
|-- extra: float
|-- mta_tax: float
|-- tip_amount: float
|-- tolls_amount: float
|-- ehail_fee: float
|-- improvement_surcharge: float
|-- total_amount: float
|-- payment_type: integer
|-- trip_type: integer
```

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q23_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q23_soln***
- Results data should be saved as a single deflate compressed avro file.
- Ignore results for which dates were incorrectly input (i.e. ignore nulls).
- Results should have the following schema:

col_name	data_type
month	int
total_passengers	bigint

PROBLEM 24

Using the gp_address table from the database gp_db, produce a table that gives the number of GP practices per city for the following 5 cities:

Brighton, Coventry, Luton, Portsmouth & Southampton

- The first 4 letters of postcodes for GP practices in Brighton are: 'BN1 ','BN2 ','BN41 ','BN42','BN45','BN50','BN51','BN88'
- The first 4 letters of postcodes for GP practices in Coventry are: 'CV1 ','CV2 ','CV3 ','CV4 ','CV5 ','CV6 ','CV7 ','CV8 '
- The first 4 letters of postcodes for GP practices in Luton are: 'LU1 ','LU2 ','LU3 ','LU4 '
- The first 4 letters of postcodes for GP practices in Portsmouth are: 'PO1 ','PO2 ','PO3 ','PO4 ','PO5 ','PO6 ','PO7 ','PO8 '
- The first 4 letters of postcodes for GP practices in Southampton are:
  'SO14','SO15','SO16','SO17','SO18','SO19'

DATA DESCRIPTION

The schema for ***gp_address*** table:

col_name	data_type
date	string
practice_code	string
surgery_name	string
address_1	string
address_2	string
address_3	string
address_4	string
postcode	string

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q24_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q24_soln***
- Results data should be saved as a single zlib compressed orc file.
- Order results by nbr_gp_practices in descending order.

SAMPLE RESULTS

The first line of the results table should look as the table below:

city	nbr_gp_practices
coventry	103

PROBLEM 25

The Wills & Fergie Bank are looking to launch their own credit card, w&f credit.

They are looking to attract new business and have decided to target high net worth individuals with more than three credit cards, each with a minimum card limit of 150,000, in order to offer them an opportunity to sign up for the new w&f century-black-gold credit card with a no limit credit limit.

You have been tasked with identifying these individuals in order to help your bank attract new business.

Use the ***card_limit*** column from ***credit_cards*** data to help you determine who these potential new customers are.

- File format of data: parquet
- HDFS location of data: ***/verulam_blue/da/data/credit_cards/***

DATA DESCRIPTION

The schema for the data is shown below:

```
|-- card_type_code: string
|-- card_type_full_name: string
|-- issuing_bank: string
|-- card_number: long
|-- card_holder_name: string
|-- cvv_cvv2: integer
|-- issue_date: string
|-- card_pin: integer
|-- card_limit: long
```

OUTPUT REQUIREMENTS

- Results data should be stored as a single tab delimited text file and compressed using gzip.
- Save this data in the HDFS location: ***/verulam_blue/da/30_worked_examples/q25_soln***
- The first column should be the ***card_holder_name***, the second column the number of cards held and the third column the minimum card limit.
- Order results by the ***card_holder_name*** in alphabetical order.

SAMPLE RESULTS

The first 3 lines from the results file are shown below:

```
|Andrea Marks 4        178400     |
|Catherine Santiago    4        173200|
|Dale Durham  4        180900     |
```

PROBLEM 26

Use the expression **quantity** * **unitcost** to work out the total amount spent by each customer that made 2 or more orders.

Use the metastore tables **customers**, **items**, **lineitems** and **orders** from the database **store_db**.

DATA DESCRIPTION

Schema for ***customers*** table:

```
|-- custID: integer
|-- firstname: string
|-- familyname: string
|-- city: string
|-- country: string
```

Schema for ***items*** table:

```
|-- itemID: string
|-- description: string
|-- unitcost: float
|-- stocklevel: integer
```

Schema for ***lineitems*** table:

```
|-- orderID: integer
|-- itemID: string
|-- quantity: integer
|-- despatched: integer
```

Schema for ***orders*** table:

```
|-- orderID: integer
|-- custID: integer
|-- date: date
```

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q26_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q26_soln***
- Results data should be saved as a single gzip compressed parquet file.
- Order results by total_spend in descending order.

SAMPLE RESULTS

The results table should have the following form:

custID	firstname	familyname	city	total_spend

PROBLEM 27

Records of all the staff working for the US branches of the company CCA175 Exam Success Ltd are stored in the the metastore table ***hr_records*** in the database ***hr_db***

Management have asked you to provide a breakdown of the ages of the current workforce working in the US.

They would like to see how many staff they have that fall under the following age categories:

Under 25
25-35
35-45
45-60
Over age 60

- Use a current date of 1st October 2020 (2020-10-01)
- Assume 365 days in a year.

DATA DESCRIPTION

The schema for the data is shown below:

```
|-- name_prefix: string
|-- first_name: string
|-- middle_initial: string
|-- last_name: string
|-- gender: string
|-- date_of_birth: date
|-- date_of_joining: date
|-- salary: integer
|-- last_pct_hike: integer
|-- ssn: string
|-- phone_nbr: string
|-- place_name: string
|-- county: string
|-- city: string
|-- state: string
|-- zip: string
|-- region: string
|-- user_name: string
```

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q27_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q27_soln***
- Results data should be saved as a single bzip2 compressed avro file.
- Order results by ***region*** in alphabetical order, followed by age category in ascending order.

SAMPLE RESULTS

The results table should have the same form as below:

region	age_category	nbr_employees
Midwest	18-25	2875

PROBLEM 28

The ***act_cost*** column, in the metastore table ***gp_rx***, represents the total actual cost of each of the prescribed medications.

Use the above information together with metastore table ***gp_address*** to determine the total spend for each GP practice.

Both tables can be found in the database gp_db.

DATA DESCRIPTION

The schema for ***gp_address*** table:

col_name	data_type
date	string
practice_code	string
surgery_name	string
address_1	string
address_2	string
address_3	string
address_4	string
postcode	string

Schema for ***gp_rx*** table:

col_name	data_type
sha	string
pct	string
practice_code	string
bnf_code	string
bnf_name	string
items	string
nic	string
act_cost	string
quantity	string
period	string

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q28_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q28_soln***
- Results data should be saved as a single gzip compressed parquet file.
- Order results by ***practice_code*** in ascending order

SAMPLE RESULTS

The first line of the results table should look as the table below:

practice_code	surgery_name	practice_spend
A81001	THE DENSHAM SURGERY	83221

PROBLEM 29

Identify how many orders were made by each city.

Use the metastore tables ***customers*** and ***orders*** from the database ***store_db***.

DATA DESCRIPTION

Schema for ***customers*** table:

```
|-- custID: integer
|-- firstname: string
|-- familyname: string
|-- city: string
|-- country: string
```

Schema for ***orders*** table:

```
|-- orderID: integer
|-- custID: integer
|-- date: date
```

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q29_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q29_soln***
- Results data should be saved as a single gzip compressed pipe delimited text file.
- Order results by total_spend in descending order.

SAMPLE RESULTS

The results table should have the following form:

```
+--------------------+
|  c_0               |
+--------------------+
|string_1 | int_1|
|string_2 | int_2|
   .         .
   .         .
   .         .
|string_n | int_n|
+--------------------+
```

PROBLEM 30

The Wills & Fergie Bank are looking prevent fraud.

They are looking to create a campaign to tell card owners to improve personal security. They have decided to begin with cardholders with pin numbers that contain less than 3 digits.

You have been tasked with identifying these individuals in order to help your bank prevent fraud.

Use the ***card_pin*** column from ***credit_cards*** data to help you determine who these individuals are.

- File format of data: parquet
- HDFS location of data: ***/verulam_blue/da/data/credit_cards/***

DATA DESCRIPTION

The schema for the data is shown below:

```
|-- card_type_code: string
|-- card_type_full_name: string
|-- issuing_bank: string
|-- card_number: long
|-- card_holder_name: string
|-- cvv_cvv2: integer
|-- issue_date: string
|-- card_pin: integer
|-- card_limit: long
```

OUTPUT REQUIREMENTS

- Results data should be stored as a single tab delimited text file and compressed using gzip.
- Save this data in the HDFS location: ***/verulam_blue/da/30_worked_examples/q30_soln***
- The first column should be the ***card_holder_name*** & the second column the digits in the ***card_pin*** (foolishly & in breach of best practice, management want you to include customer pin numbers in the results).
- Order results by the card holder's name in alphabetical order.

SAMPLE RESULTS

The first 3 lines from the results are shown below:

```
|Aaron A Grimes 51    |
|Aaron B Dunn  87     |
|Aaron C Hayes 61     |
```

PROBLEM 31 (BONUS)

The ***act_cost*** column, in the metastore table ***gp_rx***, represents the actual cost of prescribed medications.

Use the above information together with metastore tables **gp_address** and **practice_demographics** to determine the cost per patient for each GP surgery.

DATA DESCRIPTION

The schema for ***gp_address*** table:

col_name	data_type
date	string
practice_code	string
surgery_name	string
address_1	string
address_2	string
address_3	string
address_4	string
postcode	string

Schema for ***gp_rx*** table:

col_name	data_type
sha	string
pct	string
practice_code	string
bnf_code	string
bnf_name	string
items	string
nic	string
act_cost	string
quantity	string
period	string

Schema for the table ***practice_demographics***:

col_name	data_type
practice_code	string
postcode	string
nbr_of_patients	int

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q31_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q31_soln***
- Results data should be saved as a single snappy compressed parquet file.
- Order results by ***practice_code*** in ascending order.

SAMPLE RESULTS

The first line of the results table should look as the table below:

practice_code	surgery_name	cost_per_patient
A81001	THE DENSHAM SURGERY	19.66

PROBLEM 32 (BONUS)

The *act_cost* column, in the metastore table ***gp_rx***, represents the actual cost of prescribed medications.

In the ***bnf_name*** column, statin prescriptions are any prescriptions for medications that contain the strings:

'simvastatin', 'atorvastatin', 'rosuvastatin', 'pravastatin', 'fluvastatin'

Use the above information together with metastore table ***practice_demographics*** to determine the total cost of statin prescriptions and the cost per patient for all statin prescriptions for each GP surgery.

DATA DESCRIPTION

Schema for ***gp_rx*** table:

col_name	data_type
sha	string
pct	string
practice_code	string
bnf_code	string
bnf_name	string
items	string
nic	string
act_cost	string
quantity	string
period	string

Schema for the table ***practice_demographics***:

col_name	data_type
practice_code	string
postcode	string
nbr_of_patients	int

OUTPUT REQUIREMENTS

- Results should be saved as a metastore table named ***q32_soln*** in the database ***solutions_db***
- Store table data in the custom table path: ***/verulam_blue/da/30_worked_examples/q32_soln***
- Results data should be saved as a single xz compressed avro file.
- Order results by practice_code in ascending order.

SAMPLE RESULTS

The first two lines of the results table should look as the table below:

practice_code	total_act_cost_statin	statin_cost_per_patient
A81001	1817	0.43
A81002	6514	0.33