Data Analyst Test

General SQL Skills

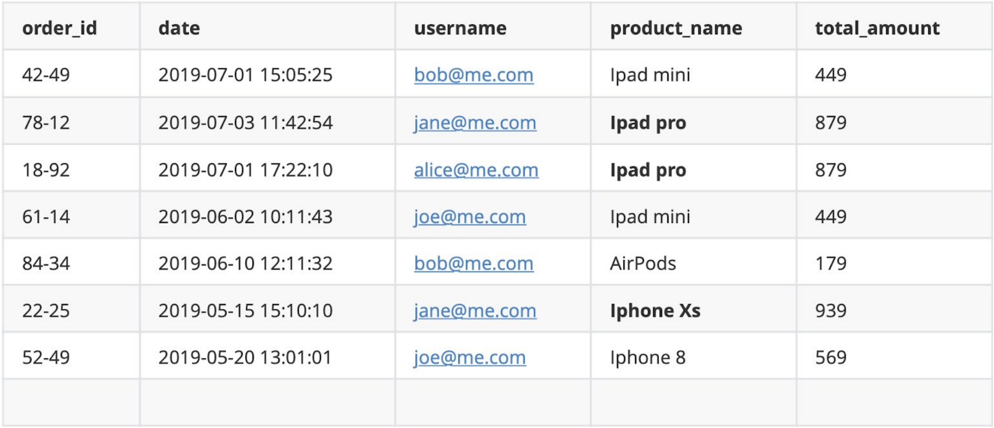
You have a `sales` relational table that stores the simplified sales information of a given company. The table has the following `CREATE DDL`.

```
CREATE TABLE `sales` (
`order_id` varchar(32) NOT NULL,
`username` varchar(64) NOT NULL,
`order_date` datetime NOT NULL,
`product_id` varchar(32) NOT NULL,
`product_name` varchar(256) NOT NULL,
`total_amount` decimal(6,2) NOT NULL,
PRIMARY KEY (`order_id`),
KEY `sales_idx_product_id`(`purchased_product_id`),
KEY  `sales_idx_username`  (`client_username`))  ENGINE=InnoDB  DEFAULT
CHARSET=utf8mb4;
```

The fields are self explanatory: they just represent **when** an order was made, **who** is the customer, **which** product they bought and the **amount** of the order.

This is a sample extract of this over simplified model:

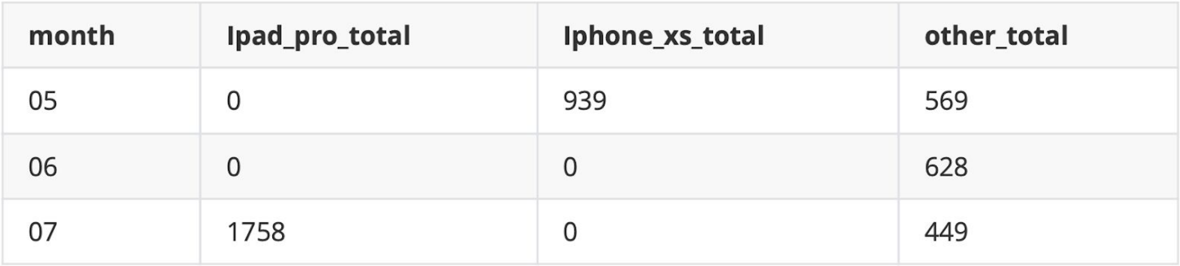
Table A



order_id	date	username	product_name	total_amount
42-49	2019-07-01 15:05:25	bob@me.com	Ipad mini	449
78-12	2019-07-03 11:42:54	jane@me.com	**Ipad pro**	879
18-92	2019-07-01 17:22:10	alice@me.com	**Ipad pro**	879
61-14	2019-06-02 10:11:43	joe@me.com	Ipad mini	449
84-34	2019-06-10 12:11:32	bob@me.com	AirPods	179
22-25	2019-05-15 15:10:10	jane@me.com	**Iphone Xs**	939
52-49	2019-05-20 13:01:01	joe@me.com	Iphone 8	569

The company wants an insight on the sales of the top Apple products the **Ipad pro** and the **Iphone Xs**, **aggregated by month**, so the first thing to do is to find a way to transform the table above into the following format:

Table B



month	Ipad_pro_total	Iphone_xs_total	other_total
05	0	939	569
06	0	0	628
07	1758	0	449

Questions:

-Describe, using SQL statements, how you would transform **Table A** into **Table B.**

-Would you use materialized tables or SQL Views?

-Is it possible to do this transformation using a single SQL statement?

-How would the transformation change if we wanted to group the sales by **week number**? And using the **name** of the month (ie. July instead of the **number** of the month 07) ?

General Python Skills

In this section, we will build two models: One for classification and one for regression analysis. We will provide the datasets needed and some guidance through the process, asking some final questions about the construction and evaluation of the models.

-Case 1: Classification

We have the `seeds_dataset` available, which shows the information regarding the characteristics of different seeds. We want to build a model to be able to predict the class of a given seed, taking into account their different features. To solve the classification problem, we will need to follow the next steps:

-Load the dataset from the file and name the columns. The features, in order of appearance, are the following:

1. Area.
2. Perimeter.
3. Compactness
4. Length of kernel.
5. Width of kernel.
6. Asymmetry coefficient.
7. Length of kernel groove.
8. Class (1, 2, 3) **(target)**

-Separate the data into the predictor variables (X) and target variable (y)

-Split X and y into training and testing datasets. **Tip:** Sklearn has great functions to do so.

-Use the model of your choice to solve the multiclass problem: Fit the data and make the predictions.

-Evaluate the results using the metric of your choice.

-Case 2: Regression

Now you are a data analyst in a Real Estate company based in Boston, USA. We have the `boston_housing` dataset, available using the next Python sentence:

```
from sklearn.datasets import load_boston()
boston = load_boston()
```

Once we have the dataset loaded, we need to build the dataframe with the predictor attributes (13) and the target (`MEDV`). The description to the dataset is available using the next sentence:

```
print(boston.DESCR)
```

Using the same methodology than in the classification problem, build a regression model to predict the `MEDV` variable using an algorithm of your choice, evaluating the results with your chosen metrics. Once this is done, create a visualization of a regression plot between the `MEDV` and `LSTAT` features.

Questions:

-Which models did you chose for each problem? Why?

-Regarding the evaluation, which metrics are you using for the classification problem? Why are you using them and how do you interpret the results? And for the regression?

-About the plot. How do you interpret the results? Which kind of relation is shown? Is it enough to take some conclusions or do we need to perform further analysis?