SQL Queries

Background

As a service to our clients, a customer journey optimisation (cjo) vendor collects client website usage data and provides insights based upon their customers' buying behaviour. A CJO currently has a relational database on a SQL server where they store this information.

One of the tables of the database is called '**Baskets**' and acts as an event log whenever a user adds an item to their basket.

Extract from Baskets:

User_ID	Session_ID	Basket_ID	Product_Name	Time_stamp
1	1	A	Shirt	2020-05-18 12:00:00.00
2	6	B	Dress	2020-05-18 12:30:00.00
3	3	C	Trousers	2020-05-18 12:55:00.00
3	3	D	Trousers	2020-05-18 13:00:00.00
3	3	D	Tie	2020-05-18 13:00:00.00

- **User_ID:** a unique ID for each user. If the same user visits multiple times, the same User ID is used.
- **Session_ID:** the number of times a user has visited the website; e.g.:
    - '1' = this is their first visit
    - '3' = this is their third visit
- **Product_Name:** the name of the product added to the basket;

    - *note*: only one product can be stored in each row; if multiple products are added to the basket, then multiple rows are needed
- **Basket_ID:** a unique ID for each basket;
    - Each time a new product is added, a new basket_ID is created; i.e. if a user adds two items to his basket during a session, he will have two basket_ids, one Basket_ID with one item in one row (old) and a second Basket_ID with two items in two rows (new).
    - For simplicity, assume for this exercise no items are removed from baskets, nor do users add multiple of the same items.
- **Time_stamp:** a date time format timestamp; all basket IDs will have the same Time_stamp as they are all created simultaneously.

There is another table in the database that tracks all transactions. This second table is called '**Sales'** which acts as an event log whenever a user makes a purchase.

Extract from Sales:

Basket_ID	Sales_value	Time_stamp
A	25	2020-05-18 12:15:00.00
D	65	2020-05-18 13:07:00.00

Each basket that exists in this table was sold for the value indicated in the 'Sales_value' field at the time indicated in 'Time_stamp'. For simplicity, assume that the value in 'Sales_value' are integers and are all in GBP.

SQL Queries

For the following questions, assume that you have been provided with a full version of the Baskets and Sales table (of which the extracts shown were a small portion).

Question 1
Write a query that will determine the top 10 most abandoned products.

Question 2
Write a query that will show the month on month change in the total value of sales in 2020.

Question 3
Write a query that will show what percentage of users who only purchase once.

Question 4
Write a query that will show on average how many days users take to purchase again.

Question 5
Assume that the following query takes a long time to run. How could you optimise it to improve the performance?

```
SELECT s.sales_value FROM
        (SELECT * FROM
        BASKETS) b
LEFT JOIN
        (SELECT * FROM SALES) s
ON b.basket_id = s.basket_id
WHERE b.user_id = 3 AND b.session_id = 3
AND DATE(b.time_stamp) >= '2020-01-01'
```