Analyze the provided dataset using Spark.

Dataset - dataset/london_house_price_data.csv

Data Columns Overview:

> **Property details:** bathrooms, bedrooms, livingRooms, floorAreaSqM, tenure, propertyType
> **Location details:** fullAddress, postcode, country, outcode, latitude, longitude
> **Energy and pricing:** currentEnergyRating, saleEstimate_, *rentEstimate_*, saleEstimate_valueChange.*
> **Historical pricing:** history_*

Answer the following:

> 1. Find the postcode with the highest average property sale price (saleEstimate_currentPrice).
> 2. Find the property type (propertyType) with the highest average number of bathrooms.
> 3. Calculate the total number of properties available in each country.
> 4. Find the average percentage change in sale price (saleEstimate_valueChange.percentageChange) for each tenure type.
> 5. Identify the country with the highest average rent price (rentEstimate_currentPrice).
> 6. Find the property type (propertyType) with the highest average number of bedrooms.
> 7. Calculate the median sale price (saleEstimate_currentPrice) for each tenure type.
> 8. Any other problem you thaught off.

Process

> 1. Develop the application in Jupyter Notebook. Test it
> 2. Once it is working correctly migrate it as a Spark application
> 3. Make sure that following things are impemented in your code
>
>> - error handling.
>> - Use of Logger wherever applicable.
>> - Documentation comments and comments.
>> - Modularity.

4. Run the Spark application from a shell script
5. Do error handling and documentation comments in Shell Script.
6. Make the shell script parameterized so that user should be in a position to run the spark application in local mode or yarn cluster or client mode.
7. Note down the time required to implement this problem statement.

In [ ]: