

Peer-graded Assignment: Submit your work and grade your peers

It looks like this is your first peer-graded assignment. [Learn more](#)

[Instructions](#)[My submission](#)[Discussions](#)

ETL Project: Acquiring and processing information on world's largest banks

Submitted on April 11, 2024

[Shareable Link](#)

PROMPT

Upload the image 'Task_1_log_function.png'. This should show the code for the function ``log_progress()`` used in the project. (1 pt)

Task 1. Log Function

Task 1. Log Function

```
1 # Code for ETL operations on Country-GDP data
2
3 # Importing the required libraries
4 from lxml import BeautifulSoup
5 import requests
6 import pandas as pd
7 import numpy as np
8 import re
9 from datetime import datetime
10
11 # Defining operation to record progress
12 def log_progress(message):
13     """ This function logs the mentioned message at a given stage of the
14     code execution to a log file. Function returns nothing. """
15     timestamp_format = '%Y-%m-%d-%H-%M-%S' # Year-Month-Day-Hour-Minute-Second
16     now = datetime.now() # get current timestamp
17     timestamp = now.strftime(timestamp_format)
18     with open("./code_log.txt", "a") as f:
19         f.write(timestamp + ' : ' + message + '\n')
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

Show the code for the function ``log_progress()`` used in the project.

RUBRIC

Did the learner upload the image 'Task_1_log_function.png'? This should show the code for the function ``log_progress()`` used in the project. (1 pt)

The image should be similar to the sample image shown:

```
def log_progress(message):
    """ This function logs the mentioned message at a given stage of the
    code execution to a log file. Function returns nothing. """
    timestamp_format = '%Y-%m-%d-%H-%M-%S' # Year-Month-Day-Hour-Minute-Second
    now = datetime.now() # get current timestamp
    timestamp = now.strftime(timestamp_format)
    with open("./code_log.txt", "a") as f:
        f.write(timestamp + ' : ' + message + '\n')
SAMPLE
```

Review the response by the learner and grade this question based on the criteria below.

- ☒ **1 point** DP
Correct. The screenshot showing the code for the function ``log_progress()`` used in the project was uploaded.
- ☐ **0 points**
Incorrect. The screenshot showing the code for the function ``log_progress()`` used in the project was not uploaded.

PROMPT

Upload the image 'Task_2a_extract.png'. This should be the screenshot of the html code obtained by inspecting the table on the webpage. The contents of the first row should be expanded and visible. (1 pt)

Task 2.a Extract

Task 2.a. Extract

```
1 # Code for ETL operations on Country-GDP data
2
3 # Importing the required libraries
4 from lxml import BeautifulSoup
5 import requests
6 import pandas as pd
7 import numpy as np
8 import re
9 from datetime import datetime
10
11 # Defining operation to record progress
12 def log_progress(message):
13     """ This function logs the mentioned message at a given stage of the
14     code execution to a log file. Function returns nothing. """
15     timestamp_format = '%Y-%m-%d-%H-%M-%S' # Year-Month-Day-Hour-Minute-Second
16     now = datetime.now() # get current timestamp
17     timestamp = now.strftime(timestamp_format)
18     with open("./code_log.txt", "a") as f:
19         f.write(timestamp + ' : ' + message + '\n')
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

RUBRIC

Did the learner upload the image 'Task_2a_extract.png'? This should be the screenshot of the html code obtained by inspecting the table on the webpage. The contents of the first row should be expanded and visible.

The image should be similar to the sample image shown:



Html code obtained by inspecting the table on the webpage.

```

<table class="wikitable sortable no-collapsible jquery-tablesorter no-wiki-collapsible"
  >
  <thead>
    <tr>
      <th>Name</th>
      <th>Age</th>
      <th>Gender</th>
      <th>Marital Status</th>
      <th>Occupation</th>
      <th>Education</th>
      <th>Income</th>
      <th>City</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Morgan Chase</td>
      <td>30</td>
      <td>Male</td>
      <td>Single</td>
      <td>Software Engineer</td>
      <td>Bachelor's</td>
      <td>100000</td>
      <td>New York</td>
    </tr>
    <tr>
      <td>Morgan Chase</td>
      <td>30</td>
      <td>Male</td>
      <td>Single</td>
      <td>Software Engineer</td>
      <td>Bachelor's</td>
      <td>100000</td>
      <td>New York</td>
    </tr>
    <tr>
      <td>Morgan Chase</td>
      <td>30</td>
      <td>Male</td>
      <td>Single</td>
      <td>Software Engineer</td>
      <td>Bachelor's</td>
      <td>100000</td>
      <td>New York</td>
    </tr>
    <tr>
      <td>Morgan Chase</td>
      <td>30</td>
      <td>Male</td>
      <td>Single</td>
      <td>Software Engineer</td>
      <td>Bachelor's</td>
      <td>100000</td>
      <td>New York</td>
    </tr>
    <tr>
      <td>Morgan Chase</td>
      <td>30</td>
      <td>Male</td>
      <td>Single</td>
      <td>Software Engineer</td>
      <td>Bachelor's</td>
      <td>100000</td>
      <td>New York</td>
    </tr>
  </tbody>
</table>

```

SAMPLE

Review the response by the learner and grade this question based on the criteria below.

- 1 point
Correct. The screenshot showing the snapshot of the html code obtained by inspecting the table on the webpage was uploaded with the contents of the first row expanded and visible.
- 0 points
Incorrect. The screenshot showing the snapshot of the html code obtained by inspecting the table on the webpage was not uploaded with the contents of the first row expanded and visible.

DP

PROMPT

Upload the image 'Task_2b_extract.png'. This should show the code for the function 'extract()' used in the project. (1 pt)

Task 2.b Extract

```

def extract(url, table_attr):
    page = requests.get(url).text
    data = BeautifulSoup(page, 'html.parser')
    df = pd.DataFrame(columns=table_attr)
    tables = data.find_all('table')
    rows = tables[0].find_all('tr')
    for row in rows:
        col = row.find('td')
        if col is not None:
            data_dict = {}
            data_dict["Name"] = col[0].text.strip()
            data_dict["MC_USD_Billion"] = col[1].text.strip()
            df1 = pd.DataFrame(data_dict, index=[0])
            df = pd.concat([df, df1], ignore_index=True)
    print(df) # Print the dataframe instead of returning it
    return df

```

Show the code for the function 'extract()' used in the project.

RUBRIC

Did the learner upload the image 'Task_2b_extract.png'? This should show the code for the function 'extract()' used in the project. (1 pt)

The image should be similar to the sample image shown:

```

def extract(url, table_attr):
    """ The purpose of this function is to extract the required
    information from the website and save it to a dataframe. The
    function returns the dataframe for further processing. """
    page = requests.get(url).text
    soup = BeautifulSoup(page, 'html.parser')
    df = pd.DataFrame(columns=table_attr)
    tables = soup.find_all('table')
    rows = tables[0].find_all('tr')
    for row in rows:
        if row.find('td') is not None:
            col = row.find('td')
            bank_name = col[1].find_all('a')[1]['title']
            market_cap = col[2].contents[0][:-1]
            data_dict = {"Name": bank_name,
                        "MC_USD_Billion": float(market_cap)}
            df1 = pd.DataFrame(data_dict, index=[0])
            df = pd.concat([df, df1], ignore_index=True)
    return df

```

SAMPLE

Review the response by the learner and grade this question based on the criteria below.

- 1 point
Correct. The screenshot showing the code for the function 'extract()' used in the project was uploaded.
- 0 points
Incorrect. The screenshot showing the code for the function 'extract()' used in the project was not uploaded.

DP

PROMPT

Upload the image 'Task_2c_extract.png'. This should be the output obtained by executing the function call. (1 pt)

Task 2.c Extract

Task 2.c. Extract

```

thelab@thelab-shreyas@10:~/thelab/projects
python3.11 bank1_project.py

# Name MC_USD_Billion
0 JPMorgan Chase 432.92
1 Bank of America 231.52
2 Industrial and Commercial Bank of China 194.56
3 Agricultural Bank of China 160.68
4 HSBC Bank 157.91
5 Wells Fargo 155.87
6 Morgan Stanley 148.83
7 China Construction Bank 139.82
8 Bank of China 136.81

[16 rows x 3 columns]

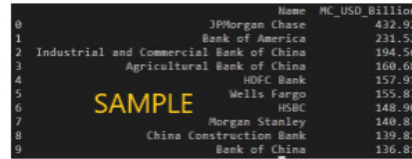
```

Output obtained by executing the function call.

RUBRIC

Did the learner upload the image 'Task_2c_extract.png'? This should be the output obtained by executing the function call. (1 pt)

The image should be similar to the sample image shown:



	Name	MC_USD_Billion
0	JPMorgan Chase	432.92
1	Bank of America	231.52
2	Industrial and Commercial Bank of China	194.56
3	Agricultural Bank of China	160.68
4	HSBC Bank	157.91
5	Wells Fargo	155.87
6	Morgan Stanley	148.83
7	China Construction Bank	139.82
8	Bank of China	136.81

Review the response by the learner and grade this question based on the criteria below.

- ☒ **1 point** DP
 Correct. The screenshot showing the output obtained by executing the function call was uploaded.
- ☐ **0 points**
 Incorrect. The screenshot showing the output obtained by executing the function call was not uploaded.

PROMPT

Upload the image 'Task_3a_transform.png'. This should show the code for the function 'transform' used in the project. (1 pt)

Task 3.a transform

Task 3.a Transform

```

def transform(df):
    # Read the exchange rate for the bank's currency
    exchange_rate_df = pd.read_csv('exchange_rate.csv')

    # Convert dataframe to dictionary using pandas syntax
    exchange_rate_dict = exchange_rate_df.set_index('Currency')['Rate'].to_dict()

    data['MC_USD_Billion'] = data['MC_USD_Billion'].apply(lambda x: x * exchange_rate_dict[x['Currency']])

    # Add columns for GDP, GDP, and DDI conversions
    data['GDP_USD_Billion'] = data['GDP_USD_Billion'] * exchange_rate_dict['USD']
    data['GDP_USD_Billion'] = data['GDP_USD_Billion'] * exchange_rate_dict['USD']
    data['GDP_USD_Billion'] = data['GDP_USD_Billion'] * exchange_rate_dict['USD']

    return df

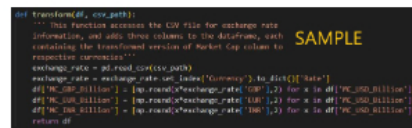
```

Show the code for the function 'transform' used in the project.

RUBRIC

Did the learner Upload the image 'Task_3a_transform.png'? This should show the code for the function 'transform' used in the project. (1 pt)

The image should be similar to the sample image shown:



```

def transform(df, csv_path):
    """ This function reads the CSV file for exchange rate
    information, and adds three columns to the dataframe, each
    containing the transformed version of Market Cap column to
    respective currencies """
    exchange_rate = pd.read_csv(csv_path)
    exchange_rate = exchange_rate.set_index('Currency')['Rate'].to_dict()
    df['MC_USD_Billion'] = [np.round(x*exchange_rate['USD'], 2) for x in df['MC_USD_Billion']]
    df['GDP_USD_Billion'] = [np.round(x*exchange_rate['GDP'], 2) for x in df['GDP_USD_Billion']]
    df['DDI_USD_Billion'] = [np.round(x*exchange_rate['DDI'], 2) for x in df['DDI_USD_Billion']]
    return df

```

Review the response by the learner and grade this question based on the criteria below.

- ☒ **1 point** DP
 Correct. The screenshot showing the code for the function 'transform' used in the project was uploaded
- ☐ **0 points**
 Incorrect. The screenshot showing the code for the function 'transform' used in the project was not uploaded

PROMPT

Upload the image 'Task_3b_transform.png'. This should be the output of the final transformed dataframe. (1 pt)

Task 3.b transform

Task 3.b Transform

```

[[{"bank": "JPMorgan Chase", "MC_USD_Billion": 432.92, "GDP_USD_Billion": 432.92, "DDI_USD_Billion": 432.92},
{"bank": "Bank of America", "MC_USD_Billion": 231.52, "GDP_USD_Billion": 231.52, "DDI_USD_Billion": 231.52},
{"bank": "Industrial and Commercial Bank of China", "MC_USD_Billion": 194.56, "GDP_USD_Billion": 194.56, "DDI_USD_Billion": 194.56},
{"bank": "Agricultural Bank of China", "MC_USD_Billion": 160.68, "GDP_USD_Billion": 160.68, "DDI_USD_Billion": 160.68},
{"bank": "HSBC Bank", "MC_USD_Billion": 157.91, "GDP_USD_Billion": 157.91, "DDI_USD_Billion": 157.91},
{"bank": "Wells Fargo", "MC_USD_Billion": 155.87, "GDP_USD_Billion": 155.87, "DDI_USD_Billion": 155.87},
{"bank": "Morgan Stanley", "MC_USD_Billion": 148.83, "GDP_USD_Billion": 148.83, "DDI_USD_Billion": 148.83},
{"bank": "China Construction Bank", "MC_USD_Billion": 139.82, "GDP_USD_Billion": 139.82, "DDI_USD_Billion": 139.82},
{"bank": "Bank of China", "MC_USD_Billion": 136.81, "GDP_USD_Billion": 136.81, "DDI_USD_Billion": 136.81}]]

```

RUBRIC

Did the learner upload the image 'Task_3b_transform.png'? This should be the output of the final transformed dataframe. (1 pt)

The image should be similar to the sample image shown:

```
1 # Merge the two dataframes into one
2 # Agricultural Bank of China
3 # Bank of America
4 # Bank of China
5 # Bank of India
6 # Bank of Japan
7 # Bank of Korea
8 # Bank of London
9 # Bank of Mexico
10 # Bank of New York
11 # Bank of Paris
12 # Bank of Spain
13 # Bank of Sweden
14 # Bank of Switzerland
15 # Bank of the United States
16 # Bank of the West
17 # Bank of the World
18 # Bank of the World
19 # Bank of the World
20 # Bank of the World
```

Output of the final transformed dataframe.

Bank of America	422.02	100.00	100.00	100.00
Bank of China	122.02	100.00	100.00	100.00
Bank of India	122.02	100.00	100.00	100.00
Bank of Japan	122.02	100.00	100.00	100.00
Bank of Korea	122.02	100.00	100.00	100.00
Bank of London	122.02	100.00	100.00	100.00
Bank of Mexico	122.02	100.00	100.00	100.00
Bank of New York	122.02	100.00	100.00	100.00
Bank of Paris	122.02	100.00	100.00	100.00
Bank of Spain	122.02	100.00	100.00	100.00
Bank of Sweden	122.02	100.00	100.00	100.00
Bank of Switzerland	122.02	100.00	100.00	100.00
Bank of the United States	122.02	100.00	100.00	100.00
Bank of the West	122.02	100.00	100.00	100.00
Bank of the World	122.02	100.00	100.00	100.00
Bank of the World	122.02	100.00	100.00	100.00
Bank of the World	122.02	100.00	100.00	100.00

Review the response by the learner and grade this question based on the criteria below.

- ☒ **1 point** DP
- Correct. The screenshot showing the output of the final transformed dataframe was uploaded.
- ☐ **0 points**
- Incorrect. The screenshot showing the output of the final transformed dataframe was not uploaded.

PROMPT

Upload the image 'Task_4_CSV.png'. This should be the contents of the CSV file created from the final table. (1 pt)

Task 4 CSV

```
1 # Merge the two dataframes into one
2 # Agricultural Bank of China
3 # Bank of America
4 # Bank of China
5 # Bank of India
6 # Bank of Japan
7 # Bank of Korea
8 # Bank of London
9 # Bank of Mexico
10 # Bank of New York
11 # Bank of Paris
12 # Bank of Spain
13 # Bank of Sweden
14 # Bank of Switzerland
15 # Bank of the United States
16 # Bank of the West
17 # Bank of the World
18 # Bank of the World
19 # Bank of the World
20 # Bank of the World
```

Contents of the CSV file created from the final table.

RUBRIC

Did the learner Upload the image 'Task_4_CSV.png'? This should be the contents of the CSV file created from the final table. (1 pt)

The image should be similar to the sample image shown:

Bank of America	422.02	100.00	100.00	100.00
Bank of China	122.02	100.00	100.00	100.00
Bank of India	122.02	100.00	100.00	100.00
Bank of Japan	122.02	100.00	100.00	100.00
Bank of Korea	122.02	100.00	100.00	100.00
Bank of London	122.02	100.00	100.00	100.00
Bank of Mexico	122.02	100.00	100.00	100.00
Bank of New York	122.02	100.00	100.00	100.00
Bank of Paris	122.02	100.00	100.00	100.00
Bank of Spain	122.02	100.00	100.00	100.00
Bank of Sweden	122.02	100.00	100.00	100.00
Bank of Switzerland	122.02	100.00	100.00	100.00
Bank of the United States	122.02	100.00	100.00	100.00
Bank of the West	122.02	100.00	100.00	100.00
Bank of the World	122.02	100.00	100.00	100.00
Bank of the World	122.02	100.00	100.00	100.00
Bank of the World	122.02	100.00	100.00	100.00

Review the response by the learner and grade this question based on the criteria below.

- ☒ **1 point** DP
- Correct. The screenshot showing the contents of the CSV file created from the final table was uploaded.
- ☐ **0 points**
- Incorrect. The screenshot showing the contents of the CSV file created from the final table was not uploaded.

PROMPT

Upload the image 'Task_4_5_save_file.png'. This should show the code for both 'load_to_csv()' and 'load_to_db()' functions used in the project. (1 pt)

Task 4_5 Save File

Task 4.5 save file

```
def load_to_csv(df, csv_path):
    """ This function saves the final dataframe as a 'CSV' file
    in the provided path. Function returns nothing. """
    df.to_csv(csv_path, index=False)

def load_to_db(df, db_name, table_name):
    """ This function saves the final dataframe as a database
    table with the provided name. Function returns nothing. """
    conn = sqlite3.connect(db_name)
    df.to_sql(table_name, conn, if_exists='replace', index=False)
    conn.close()
```

Show the code for both 'load_to_csv()' and 'load_to_db()' functions used in the project.

RUBRIC

Did the learner upload the image 'Task_4_5_save_file.png'? This should show the code for both 'load_to_csv()' and 'load_to_db()' functions used in the project. (1 pt)

The image should be similar to the sample image shown:

```
def load_to_csv(df, output_path):
    """ This function saves the final dataframe as a 'CSV' file in
    the provided path. Function returns nothing. """
    df.to_csv(output_path)

def load_to_db(df, sql_connection, table_name):
    """ This function saves the final dataframe as a database
    table with the provided name. Function returns nothing. """
    df.to_sql(table_name, sql_connection, if_exists='replace', index=False)
```

Review the response by the learner and grade this question based on the criteria below.

- ☒ **1 point** DP

Correct. The screenshot showing the code for both `load_to_csv()` and `load_to_db()` functions used in the project was uploaded.

- 0 points
Incorrect. The screenshot showing the code for both `load_to_csv()` and `load_to_db()` functions used in the project was not uploaded.

PROMPT

Upload the image 'Task_6_SQL.png'. This should be the output of the SQL queries run on the database table. (1 pt)

Task_6-SQL

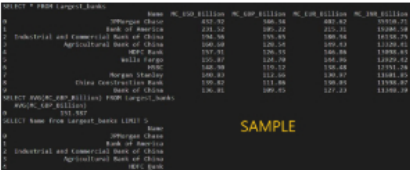
```
Task_6_SQL
thisagatha-shreya2022@name:~/p
jupyter python3.11 banks_project.py
Query Statement:
SELECT * FROM Largest_Banks
('JPMorgan Chase', 452.82, 480.42, 346.34, 33398.71)
('Bank of America', 233.42, 235.31, 225.22, 33284.52)
('Industrial and Commercial Bank of China', 334.52, 338.84, 335.45, 34238.75)
('Agricultural Bank of China', 148.48, 149.43, 138.64, 13328.43)
('HSBC Bank', 107.81, 144.88, 128.32, 13888.42)
('Wells Fargo', 155.87, 144.88, 154.7, 13328.43)
('HSBC Holdings PLC', 141.9, 138.48, 138.12, 12351.24)
('Morgan Stanley', 148.82, 138.87, 131.84, 13381.85)
('China Construction Bank', 138.82, 138.82, 133.88, 13388.87)
('Bank of China', 134.81, 127.23, 127.45, 13388.88)
Query Statement:
SELECT AVG(MC_GDP_Billion) FROM Largest_Banks
(131.85388888888889)
Query Statement:
SELECT Name FROM Largest_Banks LIMIT 5
('JPMorgan Chase',)
('Bank of America',)
('Industrial and Commercial Bank of China',)
('Agricultural Bank of China',)
('HSBC Bank',)
```

The output of the SQL queries run on the database table.

RUBRIC

Did the learner upload the image 'Task_6_SQL.png'? This should be the output of the SQL queries run on the database table. (1 pt)

The image should be similar to the sample image shown:



	Name	MC_GDP_Billion	MC_GDP_Billion	MC_GDP_Billion	MC_GDP_Billion
1	JPMorgan Chase	452.82	480.42	346.34	33398.71
2	Bank of America	233.42	235.31	225.22	33284.52
3	Industrial and Commercial Bank of China	334.52	338.84	335.45	34238.75
4	Agricultural Bank of China	148.48	149.43	138.64	13328.43
5	HSBC Bank	107.81	144.88	128.32	13888.42
6	Wells Fargo	155.87	144.88	154.7	13328.43
7	HSBC Holdings PLC	141.9	138.48	138.12	12351.24
8	Morgan Stanley	148.82	138.87	131.84	13381.85
9	China Construction Bank	138.82	138.82	133.88	13388.87
10	Bank of China	134.81	127.23	127.45	13388.88

Review the response by the learner and grade this question based on the criteria below.

- 1 point
Correct. The screenshot showing the output of the SQL queries run on the database table was uploaded.
- 0 points
Incorrect. The screenshot showing the output of the SQL queries run on the database table was not uploaded.

PROMPT

Upload the image 'Task_7_log_content.png'. This should be the contents of the log file 'code_log.txt'. (1 pt)

Task_7-SQL

```
Task_7_SQL
Welcome | banks_project.py | code_log.txt X | Largest_banks_data.csv
code_log.txt
1 2024-09-09-11:50:42 | pythian@py: process
2 2024-09-09-11:50:42 | Data extraction complete. Initiating transformation process
3 2024-09-09-11:50:42 | Data transformation complete. Initiating loading process
4 2024-09-09-11:50:42 | Data saved to CSV file
5 2024-09-09-11:50:42 | SQL Connection Initiated.
6 2024-09-09-11:50:42 | Data loaded to database as table. Running the query
7 2024-09-09-11:50:42 | Process Complete.
Contents of the log file 'code_log.txt':
```

RUBRIC

Did the learner upload the image 'Task_7_log_content.png'? This should be the contents of the log file 'code_log.txt'. (1 point)

The image should be similar to the sample image shown:



```
code_log.txt
1 2024-09-09-11:50:42 | pythian@py: process
2 2024-09-09-11:50:42 | Data extraction complete. Initiating transformation process
3 2024-09-09-11:50:42 | Data transformation complete. Initiating loading process
4 2024-09-09-11:50:42 | Data saved to CSV file
5 2024-09-09-11:50:42 | SQL Connection Initiated.
6 2024-09-09-11:50:42 | Data loaded to database as table. Running the query
7 2024-09-09-11:50:42 | Process Complete.
8 2024-09-09-11:50:42 | Server Connection Closed.
```

Review the response by the learner and grade this question based on the criteria below.

- 1 point
Correct. The screenshot showing the contents of the log file 'code_log.txt' was uploaded.
- 0 points
Incorrect. The screenshot showing the contents of the log file 'code_log.txt' was not uploaded.

Start new attempt

Comments

Comments left for the learner are visible only to that learner and the person who left the comment.

SS

Share your thoughts...

 Like

 Dislike

 Report an issue

