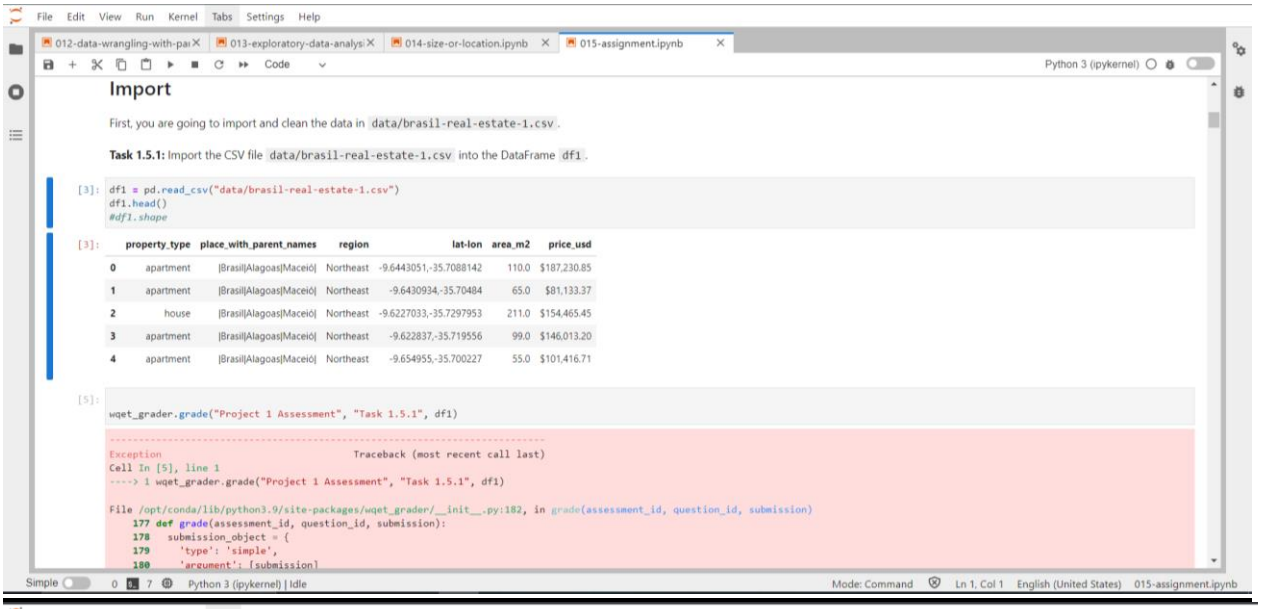
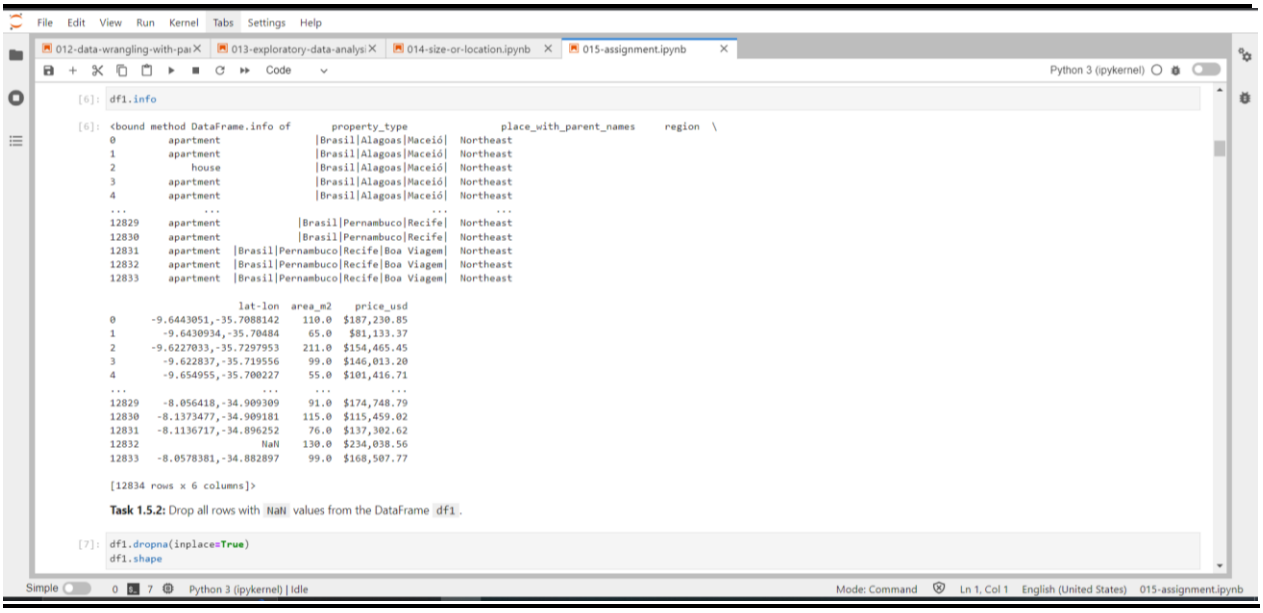
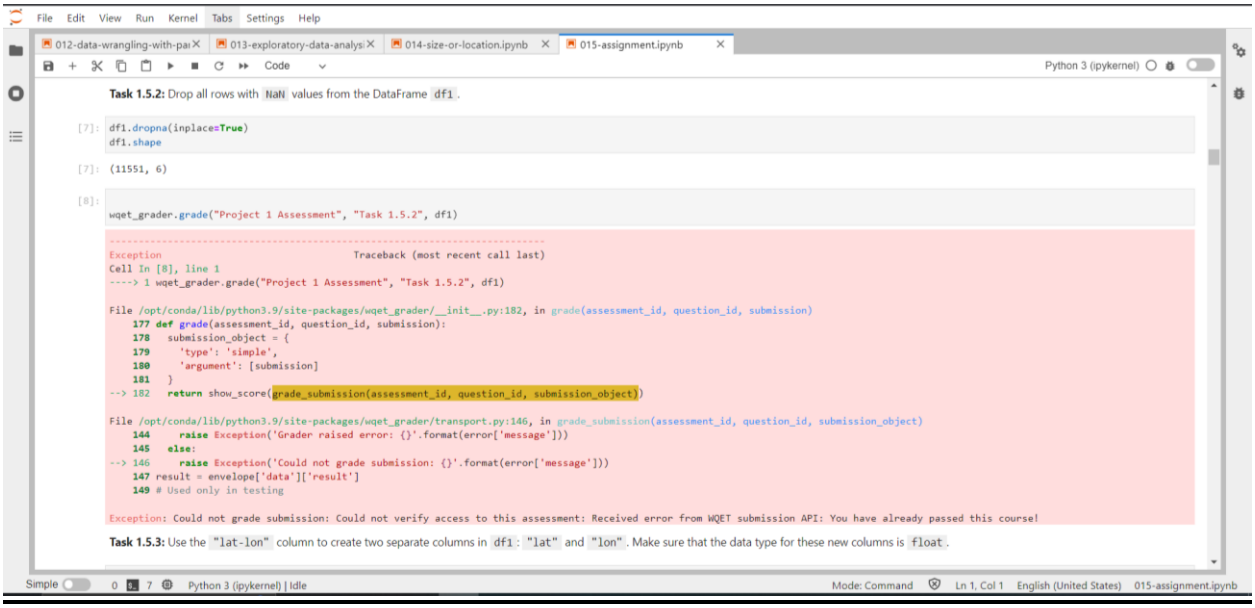
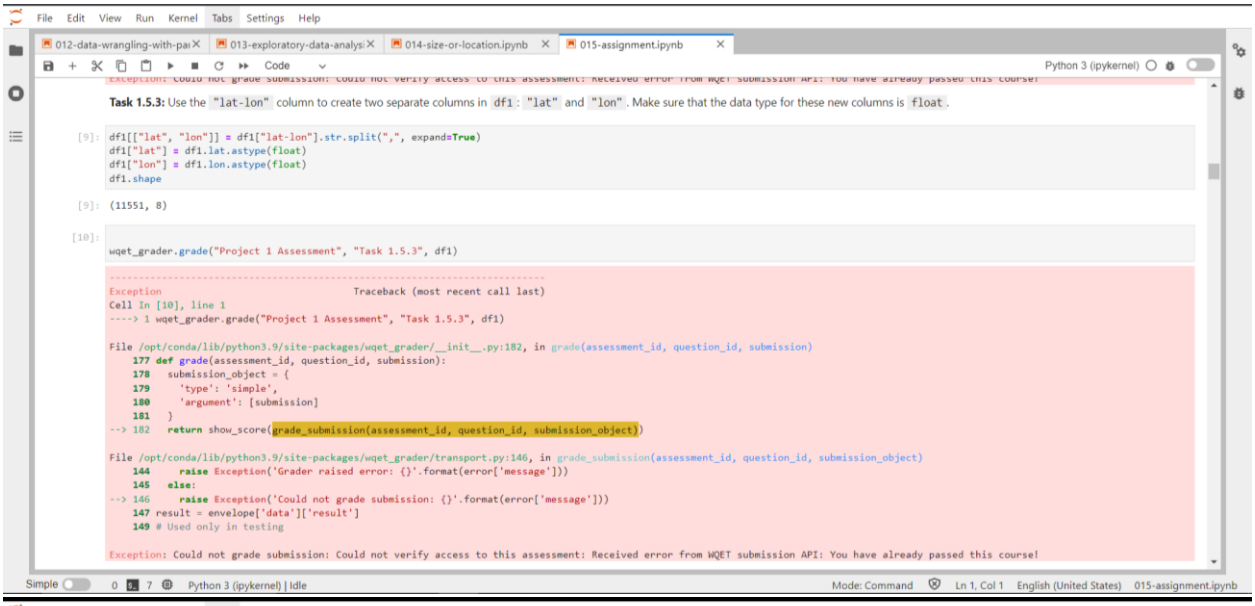
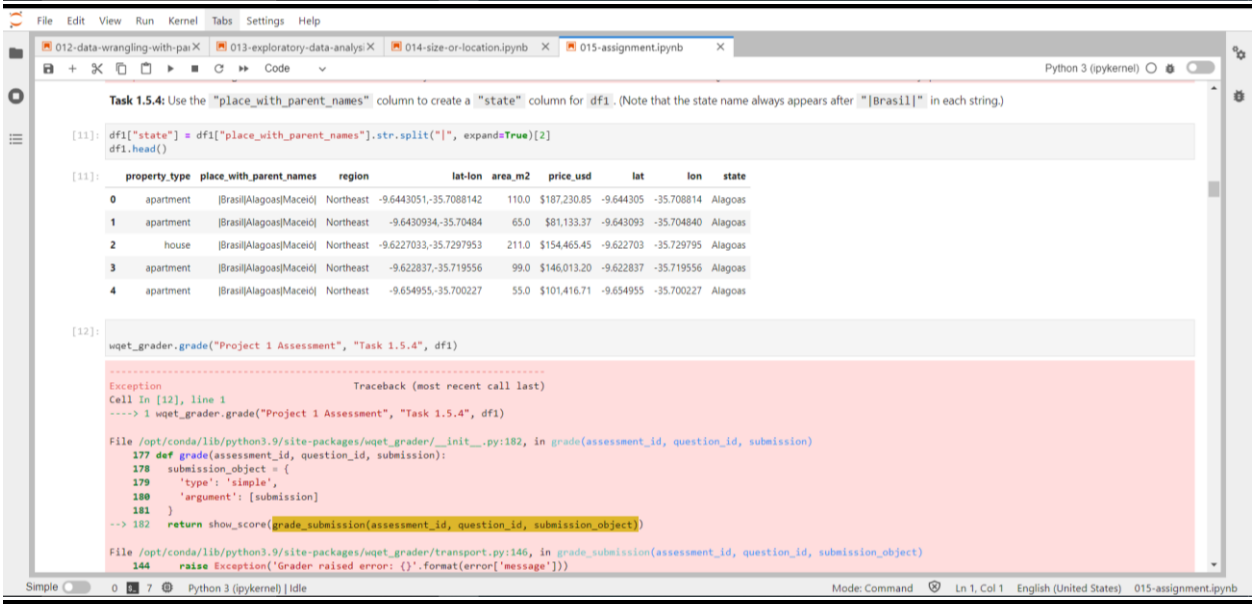
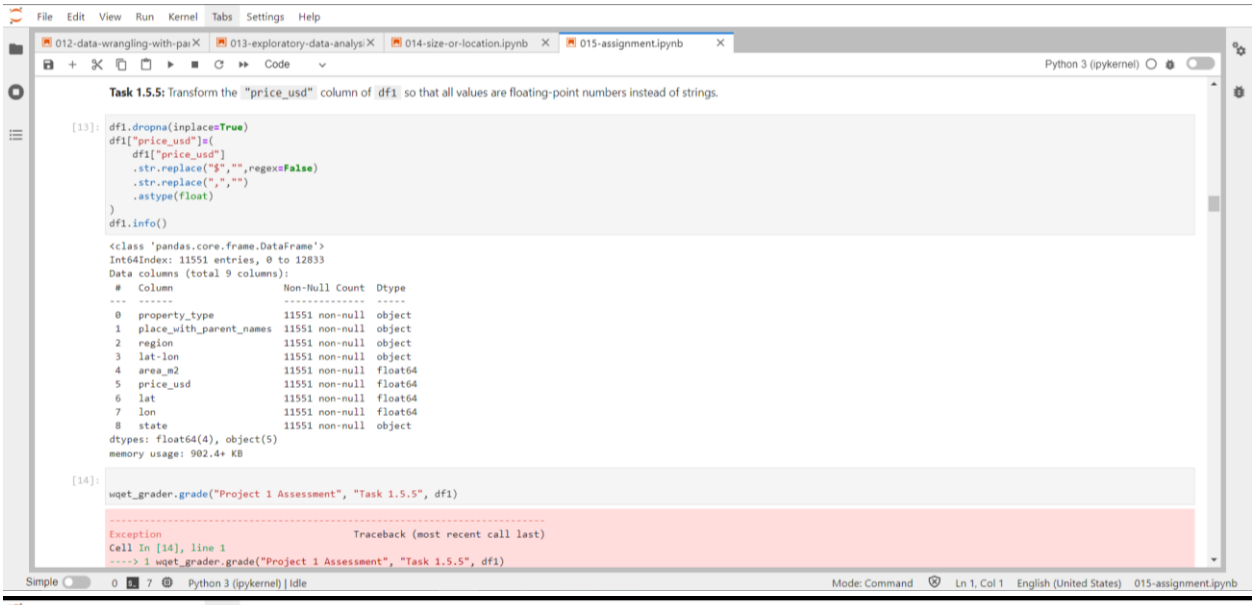

File Edit View Run Kernel Tabs Settings Help
012-data-wrangling-with-pai
013-exploratory-data-analysis
014-size-or-location.ipynb
015-assignment.ipynb
Code
Python 3 (ipykernel)
Import
First, you are going to import and clean the data in data/brasil-real-estate-1.csv .
Task 1.5.1: Import the CSV file data/brasil-real-estate-1.csv into the DataFrame df1 .
[3]: df1 = pd.read_csv("data/brasil-real-estate-1.csv")
df1.head()
#df1.shape
[3]:
property_type place_with_parent_names region lat-lon area_m2 price_usd
0 apartment |Brasil|Alagoas|Maceió| Northeast -9.6443051,-35.7088142 110.0 $187,230.85
1 apartment |Brasil|Alagoas|Maceió| Northeast -9.6430934,-35.70484 65.0 $81,133.37
2 house |Brasil|Alagoas|Maceió| Northeast -9.6227033,-35.7297953 211.0 $154,465.45
3 apartment |Brasil|Alagoas|Maceió| Northeast -9.622837,-35.719556 99.0 $146,013.20
4 apartment |Brasil|Alagoas|Maceió| Northeast -9.654955,-35.700227 55.0 $101,416.71
[5]: wqet_grader.grade("Project 1 Assessment", "Task 1.5.1", df1)
Exception Traceback (most recent call last)
Cell In [5], line 1
----> 1 wqet_grader.grade("Project 1 Assessment", "Task 1.5.1", df1)
File /opt/conda/lib/python3.9/site-packages/wqet_grader/__init__.py:182, in grade(assessment_id, question_id, submission)
177 def grade(assessment_id, question_id, submission):
178 submission_object = {
179 'type': 'simple',
180 'argument': [submission]
Simple
Python 3 (ipykernel) | Idle
Mode: Command
Ln 1, Col 1
English (United States)
015-assignment.ipynb

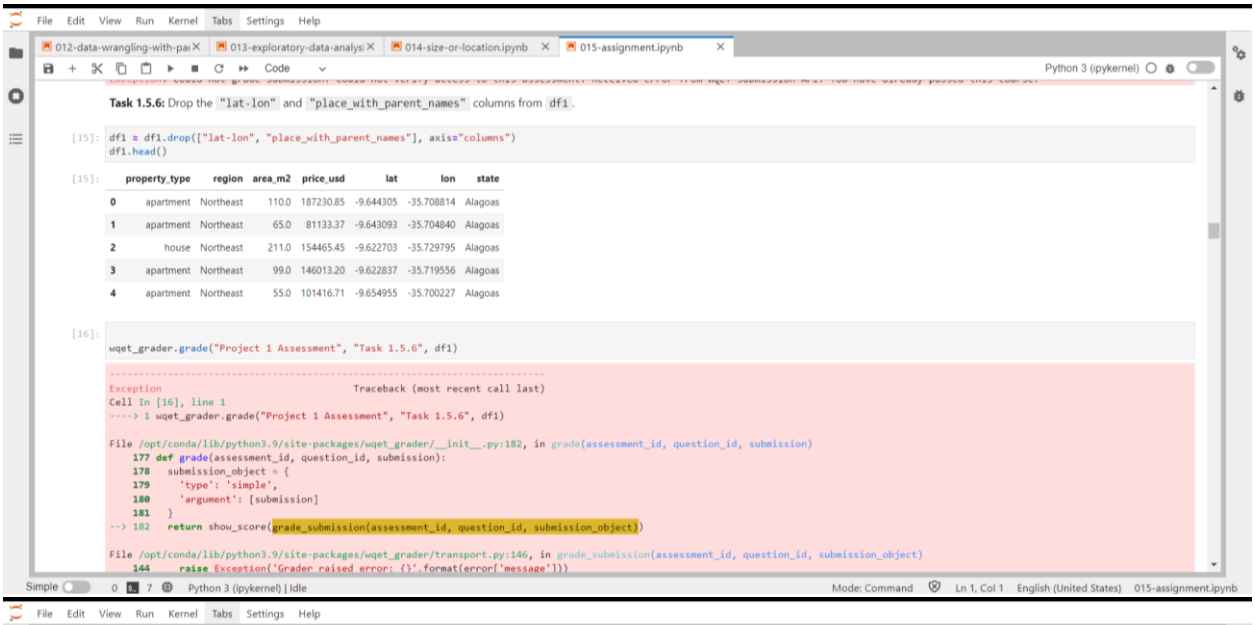

File Edit View Run Kernel Tabs Settings Help
012-data-wrangling-with-pai
013-exploratory-data-analysis
014-size-or-location.ipynb
015-assignment.ipynb
Code
Python 3 (ipykernel)
[6]: df1.info
[6]: <bound method DataFrame.info of property_type place_with_parent_names region \
0 apartment |Brasil|Alagoas|Maceió| Northeast
1 apartment |Brasil|Alagoas|Maceió| Northeast
2 house |Brasil|Alagoas|Maceió| Northeast
3 apartment |Brasil|Alagoas|Maceió| Northeast
4 apartment |Brasil|Alagoas|Maceió| Northeast
... ...
12829 apartment |Brasil|Pernambuco|Recife| Northeast
12830 apartment |Brasil|Pernambuco|Recife| Northeast
12831 apartment |Brasil|Pernambuco|Recife|Boa Viagem| Northeast
12832 apartment |Brasil|Pernambuco|Recife|Boa Viagem| Northeast
12833 apartment |Brasil|Pernambuco|Recife|Boa Viagem| Northeast
lat-lon area_m2 price_usd
0 -9.6443051,-35.7088142 110.0 $187,230.85
1 -9.6430934,-35.70484 65.0 $81,133.37
2 -9.6227033,-35.7297953 211.0 $154,465.45
3 -9.622837,-35.719556 99.0 $146,013.20
4 -9.654955,-35.700227 55.0 $101,416.71
... ...
12829 -8.056418,-34.909309 91.0 $174,748.79
12830 -8.1373477,-34.909181 115.0 $115,459.02
12831 -8.1136717,-34.896252 76.0 $137,302.62
12832 NaN 130.0 $234,038.56
12833 -8.0578381,-34.882897 99.0 $168,507.77
[12834 rows x 6 columns]>
Task 1.5.2: Drop all rows with NaN values from the DataFrame df1 .
[7]: df1.dropna(inplace=True)
df1.shape
Simple
Python 3 (ipykernel) | Idle
Mode: Command
Ln 1, Col 1
English (United States)
015-assignment.ipynb

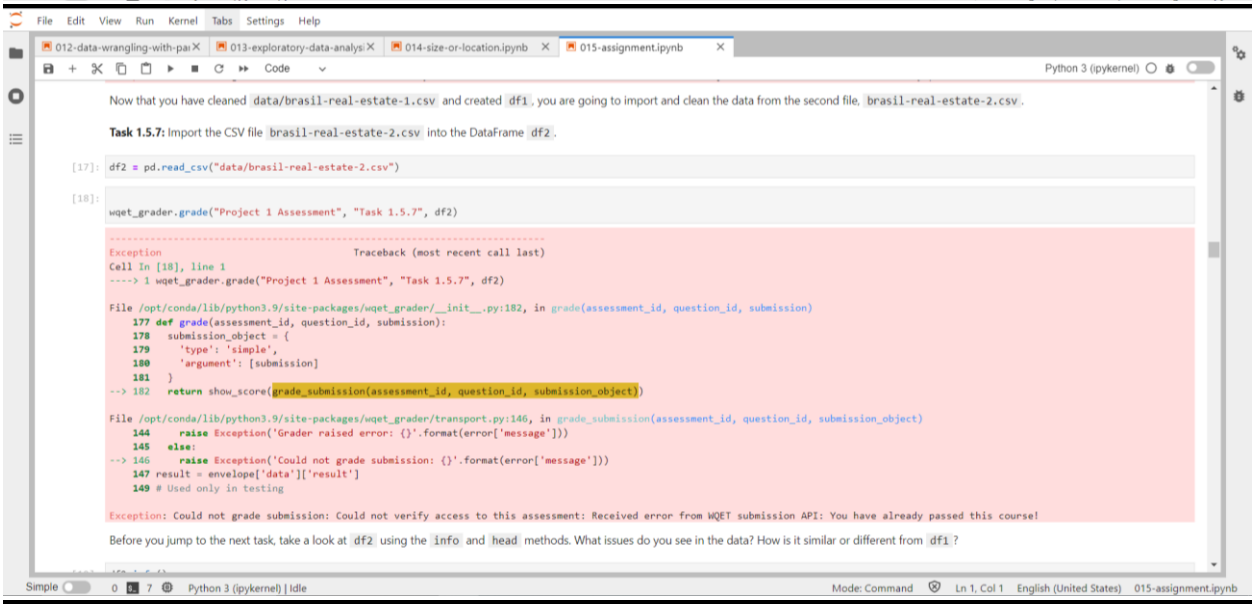

File Edit View Run Kernel Tabs Settings Help
012-data-wrangling-with-pai
013-exploratory-data-analysi
014-size-or-location.ipynb
015-assignment.ipynb
Code
Python 3 (ipykernel)
Task 1.5.2: Drop all rows with NaN values from the DataFrame df1.
[7]: df1.dropna(inplace=True)
df1.shape
[7]: (11551, 6)
[8]: wqet_grader.grade("Project 1 Assessment", "Task 1.5.2", df1)
Exception Traceback (most recent call last)
Cell In [8], line 1
----> 1 wqet_grader.grade("Project 1 Assessment", "Task 1.5.2", df1)
File /opt/conda/lib/python3.9/site-packages/wqet_grader/__init__.py:182, in grade(assessment_id, question_id, submission)
177 def grade(assessment_id, question_id, submission):
178 submission_object = {
179 'type': 'simple',
180 'argument': [submission]
181 }
--> 182 return show_score(grade_submission(assessment_id, question_id, submission_object))
File /opt/conda/lib/python3.9/site-packages/wqet_grader/transport.py:146, in grade_submission(assessment_id, question_id, submission_object)
144 raise Exception('Grader raised error: {}'.format(error['message']))
145 else:
--> 146 raise Exception('Could not grade submission: {}'.format(error['message']))
147 result = envelope['data']['result']
149 # Used only in testing
Exception: Could not grade submission: Could not verify access to this assessment: Received error from WQET submission API: You have already passed this course!
Task 1.5.3: Use the "lat-lon" column to create two separate columns in df1: "lat" and "lon". Make sure that the data type for these new columns is float.
Simple
Python 3 (ipykernel) | Idle
Mode: Command
Ln 1, Col 1
English (United States)
015-assignment.ipynb


File Edit View Run Kernel Tabs Settings Help
012-data-wrangling-with-pai
013-exploratory-data-analysi
014-size-or-location.ipynb
015-assignment.ipynb
Code
Python 3 (ipykernel)
Task 1.5.3: Use the "lat-lon" column to create two separate columns in df1: "lat" and "lon". Make sure that the data type for these new columns is float.
[9]: df1[["lat", "lon"]] = df1["lat-lon"].str.split(",", expand=True)
df1["lat"] = df1.lat.astype(float)
df1["lon"] = df1.lon.astype(float)
df1.shape
[9]: (11551, 8)
[10]: wqet_grader.grade("Project 1 Assessment", "Task 1.5.3", df1)
Exception Traceback (most recent call last)
Cell In [10], line 1
----> 1 wqet_grader.grade("Project 1 Assessment", "Task 1.5.3", df1)
File /opt/conda/lib/python3.9/site-packages/wqet_grader/__init__.py:182, in grade(assessment_id, question_id, submission)
177 def grade(assessment_id, question_id, submission):
178 submission_object = {
179 'type': 'simple',
180 'argument': [submission]
181 }
--> 182 return show_score(grade_submission(assessment_id, question_id, submission_object))
File /opt/conda/lib/python3.9/site-packages/wqet_grader/transport.py:146, in grade_submission(assessment_id, question_id, submission_object)
144 raise Exception('Grader raised error: {}'.format(error['message']))
145 else:
--> 146 raise Exception('Could not grade submission: {}'.format(error['message']))
147 result = envelope['data']['result']
149 # Used only in testing
Exception: Could not grade submission: Could not verify access to this assessment: Received error from WQET submission API: You have already passed this course!
Simple
Python 3 (ipykernel) | Idle
Mode: Command
Ln 1, Col 1
English (United States)
015-assignment.ipynb


File Edit View Run Kernel Tabs Settings Help
012-data-wrangling-with-pai
013-exploratory-data-analysi
014-size-or-location.ipynb
015-assignment.ipynb
Code
Python 3 (ipykernel)
Task 1.5.4: Use the "place_with_parent_names" column to create a "state" column for df1. (Note that the state name always appears after "|Brasil|" in each string.)
[11]: df1["state"] = df1["place_with_parent_names"].str.split("|", expand=True)[2]
df1.head()
[11]:
property_type place_with_parent_names region lat-lon area_m2 price_usd lat lon state
0 apartment |Brasil|Alagoas|Maceió| Northeast -9.6443051,-35.7088142 110.0 $187,230.85 -9.644305 -35.708814 Alagoas
1 apartment |Brasil|Alagoas|Maceió| Northeast -9.6430934,-35.70484 65.0 $81,133.37 -9.643093 -35.704840 Alagoas
2 house |Brasil|Alagoas|Maceió| Northeast -9.6227033,-35.7297953 211.0 $154,465.45 -9.622703 -35.729795 Alagoas
3 apartment |Brasil|Alagoas|Maceió| Northeast -9.622837,-35.719556 99.0 $146,013.20 -9.622837 -35.719556 Alagoas
4 apartment |Brasil|Alagoas|Maceió| Northeast -9.654955,-35.700227 55.0 $101,416.71 -9.654955 -35.700227 Alagoas
[12]: wqet_grader.grade("Project 1 Assessment", "Task 1.5.4", df1)
Exception Traceback (most recent call last)
Cell In [12], line 1
----> 1 wqet_grader.grade("Project 1 Assessment", "Task 1.5.4", df1)
File /opt/conda/lib/python3.9/site-packages/wqet_grader/__init__.py:182, in grade(assessment_id, question_id, submission)
177 def grade(assessment_id, question_id, submission):
178 submission_object = {
179 'type': 'simple',
180 'argument': [submission]
181 }
--> 182 return show_score(grade_submission(assessment_id, question_id, submission_object))
File /opt/conda/lib/python3.9/site-packages/wqet_grader/transport.py:146, in grade_submission(assessment_id, question_id, submission_object)
144 raise Exception('Grader raised error: {}'.format(error['message']))
Simple
Python 3 (ipykernel) | Idle
Mode: Command
Ln 1, Col 1
English (United States)
015-assignment.ipynb


File Edit View Run Kernel Tabs Settings Help
012-data-wrangling-with-pai
013-exploratory-data-analysi
014-size-or-location.ipynb
015-assignment.ipynb
Code
Python 3 (ipykernel)
Task 1.5.5: Transform the "price_usd" column of df1 so that all values are floating-point numbers instead of strings.
[13]:
df1.dropna(inplace=True)
df1["price_usd"]=(
df1["price_usd"]
.str.replace("$","",regex=False)
.str.replace(",","")
.astype(float)
)
df1.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 11551 entries, 0 to 12833
Data columns (total 9 columns):
Column Non-Null Count Dtype
0 property_type 11551 non-null object
1 place_with_parent_names 11551 non-null object
2 region 11551 non-null object
3 lat-lon 11551 non-null object
4 area_m2 11551 non-null float64
5 price_usd 11551 non-null float64
6 lat 11551 non-null float64
7 lon 11551 non-null float64
8 state 11551 non-null object
dtypes: float64(4), object(5)
memory usage: 902.4+ KB
[14]:
wqet_grader.grade("Project 1 Assessment", "Task 1.5.5", df1)
Exception Traceback (most recent call last)
Cell In [14], line 1
----> 1 wqet_grader.grade("Project 1 Assessment", "Task 1.5.5", df1)
Simple
Python 3 (ipykernel) | Idle
Mode: Command
Ln 1, Col 1
English (United States)
015-assignment.ipynb


File Edit View Run Kernel Tabs Settings Help
012-data-wrangling-with-pai
013-exploratory-data-analysi
014-size-or-location.ipynb
015-assignment.ipynb
Code
Python 3 (ipykernel)
Task 1.5.6: Drop the "lat-lon" and "place_with_parent_names" columns from df1.
[15]:
df1 = df1.drop(["lat-lon", "place_with_parent_names"], axis="columns")
df1.head()
[15]:
property_type region area_m2 price_usd lat lon state
0 apartment Northeast 110.0 187230.85 -9.644305 -35.708814 Alagoas
1 apartment Northeast 65.0 81133.37 -9.643093 -35.704840 Alagoas
2 house Northeast 211.0 154465.45 -9.622703 -35.729795 Alagoas
3 apartment Northeast 99.0 146013.20 -9.622837 -35.719556 Alagoas
4 apartment Northeast 55.0 101416.71 -9.654955 -35.700227 Alagoas
[16]:
wqet_grader.grade("Project 1 Assessment", "Task 1.5.6", df1)
Exception Traceback (most recent call last)
Cell In [16], line 1
----> 1 wqet_grader.grade("Project 1 Assessment", "Task 1.5.6", df1)
File /opt/conda/lib/python3.9/site-packages/wqet_grader/__init__.py:182, in grade(assessment_id, question_id, submission)
177 def grade(assessment_id, question_id, submission):
178 submission_object = {
179 'type': 'simple',
180 'argument': [submission]
181 }
--> 182 return show_score(grade_submission(assessment_id, question_id, submission_object))
File /opt/conda/lib/python3.9/site-packages/wqet_grader/transport.py:146, in grade_submission(assessment_id, question_id, submission_object)
144 raise Exception('Grader raised error: {}'.format(error['message']))
Simple
Python 3 (ipykernel) | Idle
Mode: Command
Ln 1, Col 1
English (United States)
015-assignment.ipynb


File Edit View Run Kernel Tabs Settings Help
012-data-wrangling-with-pai
013-exploratory-data-analysi
014-size-or-location.ipynb
015-assignment.ipynb
Code
Python 3 (ipykernel)
Now that you have cleaned data/brasil-real-estate-1.csv and created df1, you are going to import and clean the data from the second file, brasil-real-estate-2.csv.
Task 1.5.7: Import the CSV file brasil-real-estate-2.csv into the DataFrame df2.
[17]:
df2 = pd.read_csv("data/brasil-real-estate-2.csv")
[18]:
wqet_grader.grade("Project 1 Assessment", "Task 1.5.7", df2)
Exception Traceback (most recent call last)
Cell In [18], line 1
----> 1 wqet_grader.grade("Project 1 Assessment", "Task 1.5.7", df2)
File /opt/conda/lib/python3.9/site-packages/wqet_grader/__init__.py:182, in grade(assessment_id, question_id, submission)
177 def grade(assessment_id, question_id, submission):
178 submission_object = {
179 'type': 'simple',
180 'argument': [submission]
181 }
--> 182 return show_score(grade_submission(assessment_id, question_id, submission_object))
File /opt/conda/lib/python3.9/site-packages/wqet_grader/transport.py:146, in grade_submission(assessment_id, question_id, submission_object)
144 raise Exception('Grader raised error: {}'.format(error['message']))
145 else:
--> 146 raise Exception('Could not grade submission: {}'.format(error['message']))
147 result = envelope['data']['result']
149 # Used only in testing
Exception: Could not grade submission: Could not verify access to this assessment: Received error from WQET submission API: You have already passed this course!
Before you jump to the next task, take a look at df2 using the info and head methods. What issues do you see in the data? How is it similar or different from df1?
Simple
Python 3 (ipykernel) | Idle
Mode: Command
Ln 1, Col 1
English (United States)
015-assignment.ipynb

```
[19]: df2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 12833 entries, 0 to 12832
Data columns (total 7 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   property_type  12833 non-null  object
 1   state          12833 non-null  object
 2   region         12833 non-null  object
 3   lat            12833 non-null  float64
 4   lon            12833 non-null  float64
 5   area_m2        11293 non-null  float64
 6   price_brl      12833 non-null  float64
dtypes: float64(4), object(3)
memory usage: 701.9+ KB
```

Task 1.5.8: Use the `"price_brl"` column to create a new column named `"price_usd"`. (Keep in mind that, when this data was collected in 2015 and 2016, a US dollar cost 3.19 Brazilian reals.)

```
[20]: df2["price_usd"] = df2["price_brl"] / 3.19
      df2.head()
```

[20]:

	property_type	state	region	lat	lon	area_m2	price_brl	price_usd
0	apartment	Pernambuco	Northeast	-8.134204	-34.906326	72.0	414222.98	129850.463950
1	apartment	Pernambuco	Northeast	-8.126664	-34.903924	136.0	848408.53	265958.786834
2	apartment	Pernambuco	Northeast	-8.125550	-34.907601	75.0	299438.28	93867.799373
3	apartment	Pernambuco	Northeast	-8.120249	-34.895920	187.0	848408.53	265958.786834
4	apartment	Pernambuco	Northeast	-8.142666	-34.906906	80.0	464129.36	145495.097179

```
[21]:
```

Task 1.5.9: Drop the `"price_brl"` column from `df2`, as well as any rows that have `NaN` values.

```
[22]: df2 = df2.drop("price_brl", axis="columns")
      df2.dropna(inplace=True)
```

```
[23]: wqet_grader.grade("Project 1 Assessment", "Task 1.5.9", df2)
```

```
---------------------------------------------------------------------------
Exception                                 Traceback (most recent call last)
Cell In [23], line 1
----> 1 wqet_grader.grade("Project 1 Assessment", "Task 1.5.9", df2)

File /opt/conda/lib/python3.9/site-packages/wqet_grader/__init__.py:182, in grade(assessment_id, question_id, submission)
    177 def grade(assessment_id, question_id, submission):
    178     submission_object = {
    179         'type': 'simple',
    180         'argument': [submission]
    181     }
--> 182     return show_score(grade_submission(assessment_id, question_id, submission_object))

File /opt/conda/lib/python3.9/site-packages/wqet_grader/transport.py:146, in grade_submission(assessment_id, question_id, submission_object)
    144         raise Exception('Grader raised error: {}'.format(error['message']))
    145     else:
--> 146         raise Exception('Could not grade submission: {}'.format(error['message']))
    147 result = envelope['data']['result']
    149 # Used only in testing

Exception: Could not grade submission: Could not verify access to this assessment: Received error from WQET submission API: You have already passed this course!
```

OK! Now that you've cleaned the data from both CSV files and created `df1` and `df2`, it's time to combine them into a single DataFrame.

Task 1.5.10: Concatenate `df1` and `df2` to create a new DataFrame named `df`.

Task 1.5.10: Concatenate `df1` and `df2` to create a new DataFrame named `df`.

```
[24]: df = pd.concat([df1, df2])
      print("df shape:", df.shape)
```

```
df shape: (22844, 7)
```

```
[25]: wqet_grader.grade("Project 1 Assessment", "Task 1.5.10", df)
```

```
---------------------------------------------------------------------------
Exception                                 Traceback (most recent call last)
Cell In [25], line 1
----> 1 wqet_grader.grade("Project 1 Assessment", "Task 1.5.10", df)

File /opt/conda/lib/python3.9/site-packages/wqet_grader/__init__.py:182, in grade(assessment_id, question_id, submission)
    177 def grade(assessment_id, question_id, submission):
    178     submission_object = {
    179         'type': 'simple',
    180         'argument': [submission]
    181     }
--> 182     return show_score(grade_submission(assessment_id, question_id, submission_object))

File /opt/conda/lib/python3.9/site-packages/wqet_grader/transport.py:146, in grade_submission(assessment_id, question_id, submission_object)
    144         raise Exception('Grader raised error: {}'.format(error['message']))
    145     else:
--> 146         raise Exception('Could not grade submission: {}'.format(error['message']))
    147 result = envelope['data']['result']
    149 # Used only in testing

Exception: Could not grade submission: Could not verify access to this assessment: Received error from WQET submission API: You have already passed this course!
```

Frequent Question: I can't pass this question, and I don't know what I've done wrong. What's happening?

File Edit View Run Kernel Tabs Settings Help

012-data-wrangling-with-pai | 013-exploratory-data-analysi | 014-size-or-location.ipynb | 015-assignment.ipynb

Code | Python 3 (ipykernel)

Explore

It's time to start exploring your data. In this section, you'll use your new data visualization skills to learn more about the regional differences in the Brazilian real estate market.

Complete the code below to create a `scatter_mapbox` showing the location of the properties in `df`.

[26]:
```
fig = px.scatter_mapbox(
    df,
    lat="lat",
    lon="lon",
    center={"lat": -14.2, "lon": -51.9},  # Map will be centered on Brazil
    width=600,
    height=600,
    hover_data=["price_usd"],  # Display price when hovering mouse over house
)

fig.update_layout(mapbox_style="open-street-map")

fig.show()
```


Simple | 0 | 7 | Python 3 (ipykernel) | Idle | Mode: Command | Ln 1, Col 1 | English (United States) | 015-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

012-data-wrangling-with-pai | 013-exploratory-data-analysi | 014-size-or-location.ipynb | 015-assignment.ipynb

Code | Python 3 (ipykernel)

Simple | 0 | 7 | Python 3 (ipykernel) | Idle | Mode: Command | Ln 1, Col 1 | English (United States) | 015-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

012-data-wrangling-with-pai | 013-exploratory-data-analysi | 014-size-or-location.ipynb | 015-assignment.ipynb

Code | Python 3 (ipykernel)

Task 1.5.12: Create a histogram of `"price_usd"`. Make sure that the x-axis has the label `"Price [USD]"`, the y-axis has the label `"Frequency"`, and the plot has the title `"Distribution of Home Prices"`. Use Matplotlib (`plt`).

[29]:
```
# Build histogram
plt.hist(df["price_usd"])

# Label axes
plt.xlabel("Price [USD]")
plt.ylabel("Frequency")
# Add title
plt.title("Distribution of Home Prices")

# Don't change the code below 👇
plt.savefig("images/1-5-12.png", dpi=150)
```

[30]:
```
with open("images/1-5-12.png", "rb") as file:
```

Simple | 0 | 7 | Python 3 (ipykernel) | Idle | Mode: Command | Ln 1, Col 1 | English (United States) | 015-assignment.ipynb

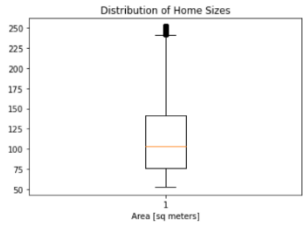
File Edit View Run Kernel Tabs Settings Help

012-data-wrangling-with-pai | 013-exploratory-data-analysi | 014-size-or-location.ipynb | 015-assignment.ipynb

Code | Python 3 (ipykernel)

Task 1.5.13: Create a horizontal boxplot of `"area_m2"`. Make sure that the x-axis has the label `"Area [sq meters]"` and the plot has the title `"Distribution of Home Sizes"`. Use Matplotlib (`plt`).

[31]:
```
# Build box plot
plt.boxplot(df["area_m2"])


# Label x-axis
plt.xlabel("Area [sq meters]")

# Add title
plt.title("Distribution of Home Sizes")

# Don't change the code below 👇
plt.savefig("images/1-5-13.png", dpi=150)
```



[32]:
```
with open("images/1-5-13.png", "rb") as file:
```

Simple | 0 | 7 | Python 3 (ipykernel) | Idle | Mode: Command | Ln 1, Col 1 | English (United States) | 015-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

012-data-wrangling-with-pai | 013-exploratory-data-analysi | 014-size-or-location.ipynb | 015-assignment.ipynb

Code | Python 3 (ipykernel)

Task 1.5.14: Use the `groupby` method to create a Series named `mean_price_by_region` that shows the mean home price in each region in Brazil, sorted from smallest to largest.

[33]:
```
mean_price_by_region = df.groupby("region")["price_usd"].mean().sort_values(ascending=True)
mean_price_by_region.head()
```

[33]:
```
region
Central-West    178596.283663
North           181308.958207
Northeast       185422.985441
South           189012.345265
Southeast       208996.762778
Name: price_usd, dtype: float64
```

[ ]:
```
wqet_grader.grade("Project 1 Assessment", "Task 1.5.14", mean_price_by_region)
```

Task 1.5.15: Use `mean_price_by_region` to create a bar chart. Make sure you label the x-axis as `"Region"` and the y-axis as `"Mean Price [USD]"`, and give the chart the title `"Mean Home Price by Region"`. Use pandas.

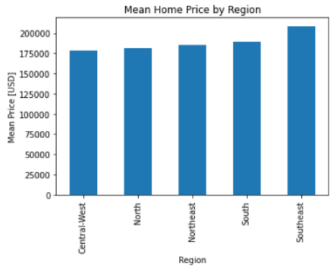
[34]:
```
# Build bar chart, label axes, add title
mean_price_by_region.plot(
    kind="bar",
    xlabel="Region",
    ylabel="Mean Price [USD]",
    title="Mean Home Price by Region"
);

# Don't change the code below 👇
plt.savefig("images/1-5-15.png", dpi=150)
```

Simple | 0 | 7 | Python 3 (ipykernel) | Idle | Mode: Command | Ln 1, Col 1 | English (United States) | 015-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

012-data-wrangling-with-pai | 013-exploratory-data-analysi | 014-size-or-location.ipynb | 015-assignment.ipynb

Code | Python 3 (ipykernel)



[35]:
```
with open("images/1-5-15.png", "rb") as file:
    wqet_grader.grade("Project 1 Assessment", "Task 1.5.15", file)
```

```
---------------------------------------------------------------------------
Exception                                 Traceback (most recent call last)
Cell In [35], line 2
      1 with open("images/1-5-15.png", "rb") as file:
----> 2     wqet_grader.grade("Project 1 Assessment", "Task 1.5.15", file)

File /opt/conda/lib/python3.9/site-packages/wqet_grader/__init__.py:182, in grade(assessment_id, question_id, submission)
    177 def grade(assessment_id, question_id, submission):
    178     submission_object = {
    179         'type': 'simple',
    180         'argument': [submission]
```

Simple | 0 | 7 | Python 3 (ipykernel) | Idle | Mode: Command | Ln 1, Col 1 | English (United States) | 015-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

012-data-wrangling-with-pai | 013-exploratory-data-analysi | 014-size-or-location.ipynb | 015-assignment.ipynb

Code | Python 3 (ipykernel)

You're now going to shift your focus to the southern region of Brazil, and look at the relationship between home size and price.

Task 1.5.16: Create a DataFrame `df_south` that contains all the homes from `df` that are in the `"South"` region.

[36]:
```
df_south = df[df["region"]=="South"]
df_south.head()
```

[36]:

	property_type	region	area_m2	price_usd	lat	lon	state
9304	apartment	South	127.0	296448.85	-25.455704	-49.292918	Paraná
9305	apartment	South	104.0	219996.25	-25.455704	-49.292918	Paraná
9306	apartment	South	100.0	194210.50	-25.460236	-49.293812	Paraná
9307	apartment	South	77.0	149252.94	-25.460236	-49.293812	Paraná
9308	apartment	South	73.0	144167.75	-25.460236	-49.293812	Paraná

[37]:
```
wqet_grader.grade("Project 1 Assessment", "Task 1.5.16", df_south)
```

```
---------------------------------------------------------------------------
Exception                                 Traceback (most recent call last)
Cell In [37], line 1
----> 1 wqet_grader.grade("Project 1 Assessment", "Task 1.5.16", df_south)

File /opt/conda/lib/python3.9/site-packages/wqet_grader/__init__.py:182, in grade(assessment_id, question_id, submission)
    177 def grade(assessment_id, question_id, submission):
    178     submission_object = {
    179         'type': 'simple',
    180         'argument': [submission]
    181     }
--> 182     return show_score(grade_submission(assessment_id, question_id, submission_object))
```

Simple | 0 | 7 | Python 3 (ipykernel) | Idle | Mode: Command | Ln 1, Col 1 | English (United States) | 015-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

012-data-wrangling-with-pai | 013-exploratory-data-analysi | 014-size-or-location.ipynb | 015-assignment.ipynb

Code | Python 3 (ipykernel)

Task 1.5.17: Use the `value_counts` method to create a Series `homes_by_state` that contains the number of properties in each state in `df_south`.

[38]:
```
homes_by_state = df_south["state"].value_counts()
homes_by_state
```

[38]:
```
Rio Grande do Sul    2643
Santa Catarina       2634
Paraná               2544
Name: state, dtype: int64
```

[39]:
```
wqet_grader.grade("Project 1 Assessment", "Task 1.5.17", homes_by_state)
```

```
---------------------------------------------------------------------------
Exception                                 Traceback (most recent call last)
Cell In [39], line 1
----> 1 wqet_grader.grade("Project 1 Assessment", "Task 1.5.17", homes_by_state)

File /opt/conda/lib/python3.9/site-packages/wqet_grader/__init__.py:182, in grade(assessment_id, question_id, submission)
    177 def grade(assessment_id, question_id, submission):
    178     submission_object = {
    179         'type': 'simple',
    180         'argument': [submission]
    181     }
--> 182     return show_score(grade_submission(assessment_id, question_id, submission_object))

File /opt/conda/lib/python3.9/site-packages/wqet_grader/transport.py:146, in grade_submission(assessment_id, question_id, submission_object)
    144         raise Exception('Grader raised error: {}'.format(error['message']))
    145     else:
--> 146         raise Exception('Could not grade submission: {}'.format(error['message']))
    147 result = envelope['data']['result']
    149 # Used only in testing

Exception: Could not grade submission: Could not verify access to this assessment: Received error from WQET submission API: You have already passed this course!
```

Simple | 0 | 7 | Python 3 (ipykernel) | Idle | Mode: Command | Ln 1, Col 1 | English (United States) | 015-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

012-data-wrangling-with-pai | 013-exploratory-data-analysi | 014-size-or-location.ipynb | 015-assignment.ipynb

Code | Python 3 (ipykernel)

Task 1.5.18: Create a scatter plot showing price vs. area for the state in `df_south` that has the largest number of properties. Be sure to label the x-axis `"Area [sq meters]"` and the y-axis `"Price [USD]"`; and use the title `"<name of state>: Price vs. Area"`. Use Matplotlib (`plt`).

Tip: You should replace `<name of state>` with the name of the state that has the largest number of properties.

[40]:
```
# Subset data
#df_south_rgs =  df[df["state"]=="Rio Grande do Sul"]

# Build scatter plot
#plt.scatter(x=homes_by_state["area_m2"], y=homes_by_state["price_usd"])


# Label axes
#plt.xlabel("Area [sq meters]")
#plt.ylabel("Price [USD]")

# Add title
#plt.title("Rio Grande do Sul: Price vs. Area")

# Don't change the code below 👇
#plt.savefig("images/1-5-18.png", dpi=150)



df_south_rgs =  df[df["state"]=="Rio Grande do Sul"]

# Create a scatter plot price vs area
plt.scatter(x=df_south_rgs["area_m2"], y=df_south_rgs["price_usd"])
plt.xlabel("Area [sq meters]")
plt.ylabel("Price [USD]")
plt.title("Rio Grande do Sul: Price vs. Area");
plt.savefig("images/1-5-18.png", dpi=150)
```

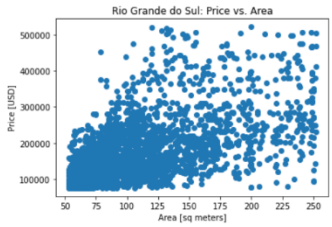
Simple | 0 | 7 | Python 3 (ipykernel) | Idle | Mode: Command | Ln 1, Col 1 | English (United States) | 015-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

012-data-wrangling-with-pai | 013-exploratory-data-analysi | 014-size-or-location.ipynb | 015-assignment.ipynb

Code — Python 3 (ipykernel)

```
# Don't change the code below 👇
#plt.savefig("images/1-5-18.png", dpi=150)



df_south_rgs =  df[df["state"]=="Rio Grande do Sul"]

# Create a scatter plot price vs area
plt.scatter(x=df_south_rgs["area_m2"], y=df_south_rgs["price_usd"])
plt.xlabel("Area [sq meters]")
plt.ylabel("Price [USD]")
plt.title("Rio Grande do Sul: Price vs. Area");
plt.savefig("images/1-5-18.png", dpi=150)
```



Simple | 0 | 7 | Python 3 (ipykernel) | Idle — Mode: Command — Ln 1, Col 1 — English (United States) — 015-assignment.ipynb

File Edit View Run Kernel Tabs Settings Help

012-data-wrangling-with-pai | 013-exploratory-data-analysi | 014-size-or-location.ipynb | 015-assignment.ipynb

Code — Python 3 (ipykernel)

Task 1.5.19: Create a dictionary `south_states_corr`, where the keys are the names of the three states in the `"south"` region of Brazil, and their associated values are the correlation coefficient between `"area_m2"` and `"price_usd"` in that state.

As an example, here's a dictionary with the states and correlation coefficients for the Southeast region. Since you're looking at a different region, the states and coefficients will be different, but the structure of the dictionary will be the same.

```
{'Espírito Santo': 0.631133255473303,
 'Minas Gerais': 0.5830029036378931,
 'Rio de Janeiro': 0.4554077103515366,
 'São Paulo': 0.45882050624839366}
```

[41]:
```
south_states_corr = df_south_rgs["area_m2"].corr(df_south_rgs["price_usd"])

south_states_corr
```

[41]:
```
0.5773267433717683
```

[42]:
```
wget_grader.grade("Project 1 Assessment", "Task 1.5.19", south_states_corr)
```

```
---------------------------------------------------------------------------
Exception                                 Traceback (most recent call last)
File /opt/conda/lib/python3.9/site-packages/wget_grader/transport.py:132, in grade_submission(assessment_id, question_id, submission_object)
    131 try:
--> 132   encoded_submission_object = encode_submission(submission_object)
    133 except Exception as e:

File /opt/conda/lib/python3.9/site-packages/wget_grader/transport.py:96, in encode_submission(object)
     95   for value in object['argument']:
---> 96     encoded_submission['argument'].append(encode_value(value))
     97 return encoded_submission

File /opt/conda/lib/python3.9/site-packages/wget_grader/transport.py:57, in encode_value(value, value_type)
     56     return {"type": "file", "format": "binary", "data": base64.b64encode(value.read()).decode()}
```

Simple | 0 | 7 | Python 3 (ipykernel) | Idle — Mode: Command — Ln 1, Col 1 — English (United States) — 015-assignment.ipynb