

Assignment 1 - Python Basics Practice

This assignment is a part of the course "[Data Analysis with Python: Zero to Pandas](#)"

In this assignment, you'll get to practice some of the concepts and skills covered in the following notebooks:

1. [First Steps with Python and Jupyter](#)
2. [A Quick Tour of Variables and Data Types](#)
3. [Branching using Conditional Statements and Loops](#)

As you go through this notebook, you will find the symbol **???** in certain places. To complete this assignment, you must replace all the **???** with appropriate values, expressions or statements to ensure that the notebook runs properly end-to-end.

Guidelines

1. Make sure to run all the code cells, otherwise you may get errors like `NameError` for undefined variables.
2. Do not change variable names, delete cells or disturb other existing code. It may cause problems during evaluation.
3. In some cases, you may need to add some code cells or new statements before or after the line of code containing the **???**.
4. Since you'll be using a free online service for code execution, save your work by running `jovian.commit` at regular intervals.
5. Questions marked **(Optional)** will not be considered for evaluation, and can be skipped. They are for your learning.
6. If you are stuck, you can ask for help on the course forum. Post errors, ask for hints and help others, but **please don't share the full solution answer code** to give others a chance to write the code themselves.
7. After submission your code will be tested with some hidden test cases. Make sure to test your code exhaustively to cover all edge cases.

Important Links:

- Make submissions here: <https://jovian.ml/learn/data-analysis-with-python-zero-to-pandas/assignment/assignment-1-python-basics-practice>
- Get help on the community forum: <https://jovian.ml/forum/t/assignment-1-python-practice/7761> . You can get help with errors or ask for hints, but **please don't ask for or share the full working answer code** on the forum.

How to Run the Code and Save Your Work

Option 1: Running using free online resources (1-click, recommended): Click the **Run** button at the top of this page and select **Run on Binder**. You can also select "Run on Colab" or "Run on Kaggle", but you'll need to create an account on [Google Colab](#) or [Kaggle](#) to use these platforms.

Option 2: Running on your computer locally: To run the code on your computer locally, you'll need to set up [Python](#) & [Conda](#), download the notebook and install the required libraries. Click the **Run** button at the top of this page, select the **Run Locally** option, and follow the instructions.

Saving your work: You can save a snapshot of the assignment to your [Jovian](#) profile, so that you can access it later and continue your work. Keep saving your work by running `jovian.commit` from time to time.

```
In [1]: # Install the library
!pip install jovian --upgrade --quiet
```

```
In [2]: # Import it
import jovian
```

```
In [3]: project_name='python-practice-assignment'
```

```
In [4]: # Capture and upload a snapshot
jovian.commit(project=project_name, privacy='secret', environment=None)
```

```
[jovian] Updating notebook "shahnawazmohammad446/python-practice-assignment" on https://jovian.ai
```

```
[jovian] Committed successfully! https://jovian.ai/shahnawazmohammad446/python-practice-assignment
```

```
Out[4]: 'https://jovian.ai/shahnawazmohammad446/python-practice-assignment'
```

You'll be asked to provide an API Key, to securely upload the notebook to your Jovian.ml account. You can get the API key from your Jovian.ml profile page after logging in / signing up. See the docs for details: <https://jovian.ml/docs/user-guide/upload.html> . The privacy of your assignment notebook is set to *Secret*, so that you can the evlauators can access it, but it will not shown on your public profile to other users.

Problem 1 - Variables and Data Types

Q1: Assign your name to the variable `name` .

```
In [5]: name = "Mohammad Shahnawaz"
```

Q2: Assign your age (real or fake) to the variable `age` .

```
In [7]: age = 23
```

Q3: Assign a boolean value to the variable `has_android_phone` .

```
In [8]: has_android_phone = True
```

You can check the values of these variables by running the next cell.

```
In [9]:
```

```
name, age, has_android_phone
```

```
Out[9]: ('Mohammad Shahnawaz', 23, True)
```

Q4: Create a dictionary person with keys "Name", "Age", "HasAndroidPhone" and values using the variables defined above.

```
In [10]: person = {"Name": "Mohammad Shahnawaz",
                  "Age": 23,
                  "HasAndroidPhone": True}
```

Let's use the person dictionary to print a nice message.

```
In [11]: print("{} is aged {}, and owns an {}".format(
          person["Name"],
          person["Age"],
          "Android phone" if person["HasAndroidPhone"] else "iPhone"
        ))
```

Mohammad Shahnawaz is aged 23, and owns an Android phone.

Q4b (Optional): Use a for loop to display the type of each value stored against each key in person .

Here's the expected output for the key "Name" :

The key "Name" has the value "Derek" of the type "<class 'str'>"

```
In [17]: # this is optional
        for p in person:
            print(type(p))
```

```
<class 'str'>
<class 'str'>
<class 'str'>
```

Now that you've solved one problem, it would be a good idea to record a snapshot of your notebook.

```
In [18]: jovian.commit(project=project_name, environment=None)
```

```
[jovian] Updating notebook "shahnawazmohammad446/python-practice-assignment" on http
s://jovian.ai
[jovian] Committed successfully! https://jovian.ai/shahnawazmohammad446/python-pract
ice-assignment
```

```
Out[18]: 'https://jovian.ai/shahnawazmohammad446/python-practice-assignment'
```

Problem 2 - Working with Lists

Q5: Create a list containing the following 3 elements:

- your favorite color
- the number of pets you have
- a boolean value describing whether you have previous programming experience

```
In [19]: my_list = ["black", 1, False]
```

Let's see what the list looks like:

```
In [20]: my_list
```

```
Out[20]: ['black', 1, False]
```

Q6: Complete the following print and if statements by accessing the appropriate elements from my_list .

Hint: Use the list indexing notation `[]` .

```
In [21]: print('My favorite color is', my_list[0])
```

My favorite color is black

```
In [23]: print('I have {} pet(s)'.format(my_list[1]))
```

I have 1 pet(s).

```
In [25]: if my_list[2]==True:
          print("I have previous programming experience")
        else:
          print("I do not have previous programming experience")
```

I do not have previous programming experience

Q7: Add your favorite single digit number to the end of the list using the appropriate list method.

```
In [26]: my_list.append(7)
```

Let's see if the number shows up in the list.

```
In [27]: my_list
```

```
Out[27]: ['black', 1, False, 7]
```

Q8: Remove the first element of the list, using the appropriate list method.

Hint: Check out methods of list here: https://www.w3schools.com/python/python_ref_list.asp

```
In [28]: my_list.remove("black")
```

```
In [29]: my_list
```

```
Out[29]: [1, False, 7]
```

Q9: Complete the print statement below to display the number of elements in my_list .

```
In [30]: print("The list has {} elements.".format(len(my_list)))
```

The list has 3 elements.

Well done, you're making good progress! Save your work before continuing

```
In [31]: jovian.commit(project=project_name, environment=None)
```

[jovian] Updating notebook "shahnawazmohammad446/python-practice-assignment" on <https://jovian.ai>

[jovian] Committed successfully! <https://jovian.ai/shahnawazmohammad446/python-practice-assignment>

```
Out[31]: 'https://jovian.ai/shahnawazmohammad446/python-practice-assignment'
```

Problem 3 - Conditions and loops

Q10: Calculate and display the sum of all the numbers divisible by 7 between 18 and 534 i.e. $21+28+35+\dots+525+532$.

Hint: One way to do this is to loop over a range using for and use an if statement inside it.

```
In [39]: # store the final answer in this variable
sum_of_numbers = 0
n = 18

# perform the calculation here
while n<534:
    if not n%7:
        sum_of_numbers+=n
    n+=1
```

```
In [40]: print('The sum of all the numbers divisible by 7 between 18 and 534 is', sum_of_numbers)
```

The sum of all the numbers divisible by 7 between 18 and 534 is 20461

If you are not able to figure out the solution to this problem, you can ask for hints on the community forum: <https://jovian.ml/forum/t/assignment-1-python-practice/7761> . Remember to save your work before moving forward.

```
In [41]: jovian.commit(project=project_name, environment=None)
```

[jovian] Updating notebook "shahnawazmohammad446/python-practice-assignment" on <https://jovian.ai>

[jovian] Committed successfully! <https://jovian.ai/shahnawazmohammad446/python-practice-assignment>

```
Out[41]: 'https://jovian.ai/shahnawazmohammad446/python-practice-assignment'
```

Problem 4 - Flying to the Bahamas

Q11: A travel company wants to fly a plane to the Bahamas. Flying the plane costs 5000 dollars. So far, 29 people have signed up for the trip. If the company charges 200 dollars per ticket, what is the profit made by the company?

Fill in values or arithmetic expressions for the variables below.

```
In [44]: cost_of_flying_plane = 5000
```

```
In [48]: number_of_passengers = 29
```

```
In [46]: price_of_ticket = 200
```

```
In [50]: profit = (number_of_passengers*price_of_ticket)-cost_of_flying_plane
```

```
In [51]: print('The company makes of a profit of {} dollars'.format(profit))
```

The company makes of a profit of 800 dollars

Q11b (Optional): Out of the 29 people who took the flight, only 12 buy tickets to return from the Bahamas on the same plane. If the flying the plane back also costs 5000 dollars, and does the company make an overall profit or loss? The company charges the same fee of 200 dollars per ticket for the return flight.

Use an `if` statement to display the result.

```
In [ ]: # this is optional
      ???
```

```
In [ ]: # this is optional
      if ???:
          print("The company makes an overall profit of {} dollars".format(???))
      else:
          print("The company makes an overall loss of {} dollars".format(???))
```

Great work so far! Want to take a break? Remember to save and upload your notebook to record your progress.

```
In [52]: jovian.commit(project=project_name,environment=None)
```

[jovian] Updating notebook "shahnawazmohammad446/python-practice-assignment" on <https://jovian.ai>

[jovian] Committed successfully! <https://jovian.ai/shahnawazmohammad446/python-practice-assignment>

```
Out[52]: 'https://jovian.ai/shahnawazmohammad446/python-practice-assignment'
```

Problem 5 - Twitter Sentiment Analysis

Are you ready to perform some *Data Analysis with Python*? In this problem, we'll analyze some fictional tweets and find out whether the overall sentiment of Twitter users is happy or sad. This is a simplified version of an important real world problem called *sentiment analysis*.

Before we begin, we need a list of tweets to analyze. We're picking a small number of tweets here, but the exact same analysis can also be done for thousands, or even millions of tweets. The

collection of data that we perform analysis on is often called a *dataset*.

```
In [54]: tweets = [
    "Wow, what a great day today!! #sunshine",
    "I feel sad about the things going on around us. #covid19",
    "I'm really excited to learn Python with @JovianML #zerotopandas",
    "This is a really nice song. #linkinpark",
    "The python programming language is useful for data science",
    "Why do bad things happen to me?",
    "Apple announces the release of the new iPhone 12. Fans are excited.",
    "Spent my day with family!! #happy",
    "Check out my blog post on common string operations in Python. #zerotopandas",
    "Freecodecamp has great coding tutorials. #skillup"
]
```

Let's begin by answering a very simple but important question about our dataset.

Q12: How many tweets does the dataset contain?

```
In [55]: number_of_tweets = len(tweets)
```

Let's create two lists of words: `happy_words` and `sad_words`. We will use these to check if a tweet is happy or sad.

```
In [56]: happy_words = ['great', 'excited', 'happy', 'nice', 'wonderful', 'amazing', 'good',
```

```
In [57]: sad_words = ['sad', 'bad', 'tragic', 'unhappy', 'worst']
```

To identify whether a tweet is happy, we can simply check if contains any of the words from `happy_words`. Here's an example:

```
In [58]: sample_tweet = tweets[0]
```

```
In [59]: sample_tweet
```

```
Out[59]: 'Wow, what a great day today!! #sunshine'
```

```
In [60]: is_tweet_happy = False

# Get a word from happy_words
for word in happy_words:
    # Check if the tweet contains the word
    if word in sample_tweet:
        # Word found! Mark the tweet as happy
        is_tweet_happy = True
```

Do you understand what we're doing above?

For each word in the list of happy words, we check if it is a part of the selected tweet. If the word is indeed a part of the tweet, we set the variable `is_tweet_happy` to `True`.

```
In [61]: is_tweet_happy
```

```
Out[61]: True
```

Q13: Determine the number of tweets in the dataset that can be classified as happy.

Hint: You'll need to use a loop inside another loop to do this. Use the code from the example shown above.

```
In [70]: # store the final answer in this variable
         number_of_happy_tweets = 0

         # perform the calculations here
         for i in range(0, len(tweets)):
             for word in happy_words:
                 if word in tweets[i]:
                     is_tweet_happy = True
                     number_of_happy_tweets += 1
```

```
In [71]: print("Number of happy tweets:", number_of_happy_tweets)
```

Number of happy tweets: 6

If you are not able to figure out the solution to this problem, you can ask for hints on the community forum: <https://jovian.ml/forum/t/assignment-1-python-practice/7761> . Also try adding `print` statements inside your loops to inspect variables and make sure your logic is correct.

Q14: What fraction of the total number of tweets are happy?

For example, if 2 out of 10 tweets are happy, then the answer is $2/10$ i.e. 0.2 .

```
In [72]: happy_fraction = number_of_happy_tweets / len(tweets)
```

```
In [73]: print("The fraction of happy tweets is:", happy_fraction)
```

The fraction of happy tweets is: 0.6

To identify whether a tweet is sad, we can simply check if contains any of the words from `sad_words` .

Q15: Determine the number of tweets in the dataset that can be classified as sad.

```
In [74]: # store the final answer in this variable
         number_of_sad_tweets = 0

         # perform the calculations here
         for i in range(0, len(tweets)):
             for word in sad_words:
                 if word in tweets[i]:
                     is_tweet_sad = True
                     number_of_sad_tweets += 1
```

```
In [75]: print("Number of sad tweets:", number_of_sad_tweets)
```


Number of sad tweets: 2

Q16: What fraction of the total number of tweets are sad?

```
In [76]: sad_fraction = number_of_sad_tweets/len(tweets)
```

```
In [77]: print("The fraction of sad tweets is:", sad_fraction)
```

The fraction of sad tweets is: 0.2

The rest of this problem is optional. Let's save your work before continuing.

```
In [78]: jovian.commit(project=project_name,environment=None)
```

[jovian] Updating notebook "shahnawazmohammad446/python-practice-assignment" on <https://jovian.ai>

[jovian] Committed successfully! <https://jovian.ai/shahnawazmohammad446/python-practice-assignment>

```
Out[78]: 'https://jovian.ai/shahnawazmohammad446/python-practice-assignment'
```

Great work, even with some basic analysis, we already know a lot about the sentiment of the tweets given to us. Let us now define a metric called "sentiment score", to summarize the overall sentiment of the tweets.

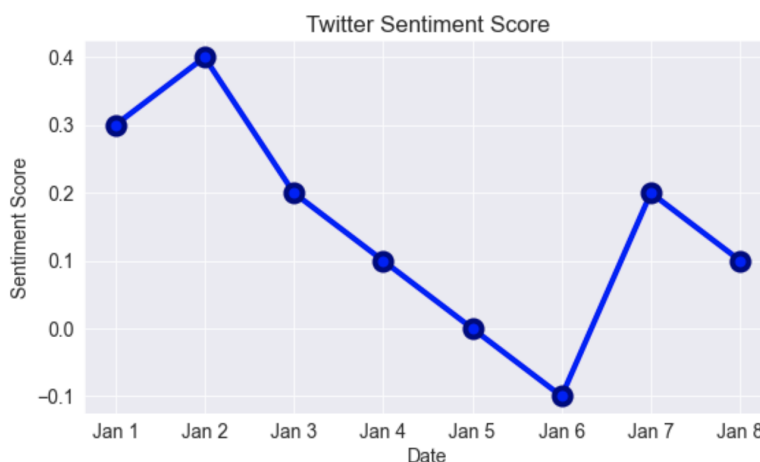
Q16b (Optional): Calculate the sentiment score, which is defined as the difference between the fraction of happy tweets and the fraction of sad tweets.

```
In [79]: sentiment_score = happy_fraction-sad_fraction
```

```
In [80]: print("The sentiment score for the given tweets is", sentiment_score)
```

The sentiment score for the given tweets is 0.39999999999999997

In a real world scenario, we could calculate & record the sentiment score for all the tweets sent out every day. This information can be used to plot a graph and study the trends in the changing sentiment of the world. The following graph was creating using the Python data visualization library `matplotlib`, which we'll cover later in the course.



What does the sentiment score represent? Based on the value of the sentiment score, can you identify if the overall sentiment of the dataset is happy or sad?

Q16c (Optional): Display whether the overall sentiment of the given dataset of tweets is happy or sad, using the sentiment score.

```
In [82]: if sentiment_score > 0:
          print("The overall sentiment is happy")
        else:
          print("The overall sentiment is sad")
```

The overall sentiment is happy

Finally, it's also important to track how many tweets are neutral i.e. neither happy nor sad. If a large fraction of tweets are marked neutral, maybe we need to improve our lists of happy and sad words.

Q16d (Optional): What is the fraction of tweets that are neutral i.e. neither happy nor sad.

```
In [83]: # store the final answer in this variable
          number_of_neutral_tweets = 0

          # perform the calculation here
          number_of_neutral_tweets = len(tweets) - number_of_happy_tweets - number_of_sad_tweets
```

```
In [84]: neutral_fraction = number_of_neutral_tweets/len(tweets)
```

```
In [85]: print('The fraction of neutral tweets is', neutral_fraction)
```

The fraction of neutral tweets is 0.2

Ponder upon these questions and try some experiments to hone your skills further:

- What are the limitations of our approach? When will it go wrong or give incorrect results?
- How can we improve our approach to address the limitations?
- What are some other questions you would like to ask, given a list of tweets?
- Try collecting some real tweets from your Twitter timeline and repeat this analysis. Do the results make sense?

IMPORTANT NOTE: If you want to try out these experiments, please create a new notebook using the "New Notebook" button on your Jovian.ml profile, to avoid making unintended changes to your assignment submission notebook.

Submission

Congratulations on making it this far! You've reached the end of this assignment, and you just completed your first data analysis problem. It's time to record one final version of your notebook for submission.

Make a submission here by filling the submission form: <https://jovian.ml/learn/data-analysis-with-python-zero-to-pandas/assignment/assignment-1-python-basics-practice>

```
In [ ]: jovian.commit(project=project_name, environment=None)
```

In []: