02 Team Activity: Calling Functions

Instructions

Arrange a one hour synchronous meeting with your team for this activity. Online students should coordinate a video-sharing meeting. Campus students will use class time for this meeting. You should prepare for this meeting by completing the preparation material and the individual checkpoint assignment beforehand.

Purpose

Improve your understanding of calling built-in functions and functions and methods that are in a standard Python module.

Problem Statement

You work for a retail store that wants to increase sales on Tuesday and Wednesday, which are the store's slowest sales days. On Tuesday and Wednesday, if a customer's subtotal is $50 or greater, the store will discount the customer's subtotal by 10%.

Assignment

Write a Python program named `discount.py` that gets a customer's subtotal as input and gets the current day of the week from your computer's clock. Your program must ***not*** ask the user to enter the day of the week. Instead, it must get the day of the week from your computer's clock.

If the subtotal is $50 or greater and today is Tuesday or Wednesday, your program must subtract 10% from the subtotal. Your program must then compute the total amount due by adding sales tax of 6% to the subtotal. Your program must print the discount amount if applicable, the sales tax amount, and the total amount due.

Core Requirements

1. Your program asks the user for the subtotal but does ***not*** ask the user for the day of the week. Your program gets the day of the week from your computer's clock.
2. Your program correctly computes and prints the discount amount if applicable.
3. Your program correctly computes and prints the sales tax amount and the total amount due.

Stretch Challenges

If your team finishes the core requirements in less than an hour, complete one or more of these stretch challenges. Note that the stretch challenges are optional.

1. Add code to your program that the computer will execute if today is Tuesday or Wednesday and the customer is not purchasing enough to receive the discount. This added code should compute and print the difference between $50 and the subtotal which is the additional amount the customer would need to purchase in order to receive the discount.

2. Near the beginning of your program replace the code that asks the user for the subtotal with a loop that repeatedly asks the user for a price and a quantity and computes the subtotal. This loop should repeat until the user enters the word "done" for the price.

Helpful Documentation

- The prepare content for this lesson explains how to call a function and a method.
- The `datetime.now()` method from the standard Python `datetime` module will get the current date and time from your computer's operating system. Here is an excerpt from the official documentation for the `datetime.now` method:

> `datetime.now(tz=None)`
> Return the current local date and time.
>
> *tz* is optional, but if it not `None`, it must be `tzinfo` (time zone information) object

The `weekday()` method will get an integer that represents the day of the week from a `datetime` object. Here is an excerpt from the official documentation for the `weekday` method:

> `dt.weekday()`
> Return the day of the week as an integer, where Monday is 0 and Sunday is 6.

These two Microsoft videos explain how to use methods from the standard `datetime` module.

Date data types (8 minutes)

Demonstration: Dates (9 minutes)

The following Python code imports the `datetime` class from the `datetime` module, calls the `datetime.now` method to get the current date and time from a computer's operating system, and then calls the weekday method get the day of the week as an integer.

```
 1  # Import the datetime class from the datetime
 2  # module so that it can be used in this program.
 3  from datetime import datetime
 4
 5  # Call the now() method to get the current date and
 6  # time as a datetime object from the computer's clock.
 7  current_date_and_time = datetime.now()
 8
 9  # Call the weekday() method to get the day of the
10  # week from the current_date_and_time object.
11  day_of_week = current_date_and_time.weekday()
12
13  # Print the day of the week for the user to see.
14  print(day_of_week)
```

```
> python day_of_week.py
4
```

After the computer executes line 7 in the above code, the variable *current_date_and_time* will hold the current date and time. After the computer executes line 11, the day_of_week variable will hold 0 if today is Monday, 1 if today is Tuesday, and so on to 6 if today is Sunday.

Testing Procedure

Verify that your program works correctly by following each step in this testing procedure:

1. If today is any day other than Tuesday or Wednesday, run your program using the input shown in the "Run on a Monday" section below. Ensure that your program's output matches the sample run output. If today is Tuesday or Wednesday, run your program using the input in the "Run on a Tuesday" section.

2. Is there a simple way to test your program for all the days of the week without testing on seven consecutive days and without changing your computer's system date? Hint: In your program immediately after these three lines of code:

```
# Call the weekday() method to get the day of the
# week from the current_date_and_time object.
day_of_week = current_date_and_time.weekday()
```

temporarily add a line of code like this one:

```
weekday = 2
```

Sample Runs

Run on a Monday

```
> python discount.py
Please enter the subtotal: 42.75
Sales tax amount: 2.56
Total: 45.31

> python discount.py
Please enter the subtotal: 55.20
Sales tax amount: 3.31
Total: 58.51
```

Run on a Tuesday

```
> python discount.py
Please enter the subtotal: 42.75
Sales tax amount: 2.56
Total: 45.31

> python discount.py
Please enter the subtotal: 55.20
Discount amount: 5.52
Sales tax amount: 2.98
Total: 52.66
```

Sample Solution

Please work diligently with your team for the one hour meeting. After the meeting is over, please compare your approach to the sample solution [↓] or the stretch solution [↓]. Please ***do not look at the sample solutions*** until you have either finished the program or diligently worked for at least one hour. At the end of the hour, if you are still struggling to complete the assignment, you may use the sample solution to help you finish.

Submission

When you have finished the activity, please report your progress via the associated I-Learn quiz. When asked about which of the requirements you completed, feel free to include any work done during the team meeting or after the meeting, including work done with the help of the sample solution, if necessary. In short, report on what you were able to accomplish, regardless of when you completed it or if you needed help from the sample solution.