Christian_Herrera_HW1

February 14, 2026

1 Homework 1. Basic Python (Questions)

2 Assignment Submission Guidelines

Please follow the guidelines below for submitting your assignment:

1. **Submission Deadline:**
   - All assignments must be submitted **no later than 23:59 PM next Tuesday**.
   - Late submissions will not be accepted unless prior arrangements have been made by the instructor.
2. **Submission Platform:**
   - Submit your assignment through **Canvas**. Ensure that you upload the files to the correct assignment link.
3. **Required Files:**
   - **Python Script (.ipynb file):** Submit the Python Jupyter notebook script you used to complete the assignment. The file should contain your well-commented code.
   - **PDF Version (.pdf file):** Additionally, submit a PDF version of your Python code. This can be a printout or export of your script, showing all the code with any necessary explanations or output results included.
4. **File Naming Convention:**
   - Please name your files as follows: `Lastname_Firstname_AssignmentName`
   - Example: `Alex_John_HW1.ipynb` and `Alex_John_HW1.pdf`
5. **Technical Issues:**
   - If you encounter any technical issues with Canvas or your submission, please contact the TAs immediately **before the deadline** to avoid penalties.

3 Questions

1. Show what Markdown is used for in Jupyter Notebooks. (5 Points)

[ ]:
```
# Markdown is used in Jupyter Notebooks to format text, create headings, lists,␣
↪and more.
# This helps create documents that are easy to read and comprehend.
```

3.0.1 This is a heading.

In Visual Studio Code, headings in the notebook appear in the outline which make it really easy to follow the flow of the notebook.

2. Show how to format text to be bold in Markdown? Provide an example. (5 Points)

[55]:
```
# Personally, I prefer using the HTML style markdown notation.
# <b> In a markdown cell, this text is bold </b>
```

This text is bold, and I executed the cell.

3. Create a Python variable named `a` and assign it the value 5. (5 Points)

[56]:
```
a = 5 # The given variable name and value.
```

4. Perform an addition operation with any two numbers and display the result. (5 Points)

[57]:
```
addition_ex = 4 + 3 # The addition operation.

print(addition_ex) # Printing the result.
```

```
7
```

5. Divide 15 by 3 and show the result. (5 Points)

[58]:
```
division_ex = 15/3 # The division operation.

print(division_ex) # Printing the result.
```

```
5.0
```

6. Use the `input()` function to ask the user for their name and then greet them. (5 Points)

[59]:
```
name = input('Enter your name: ') # Prompting user for their name.

print(f'Hello, {name}!') # Printing the message with their name.
```

```
Hello, Christian!
```

7. Calculate the square of 7 and display the result. (5 Points)

[60]:
```
square_of_7 = 7 ** 2 # The square operation.

print(square_of_7) # Printing the result.
```

```
49
```

8. Ask the user for a number, multiply it by 2, and display the result. (5 Points)

[61]:
```
user_numer = input('Enter a number: ') # Prompting user for a number.
user_numer_times_two = int(user_numer) * 2 # Performing the operation.

print(user_numer_times_two) # Printing the result.
```

```
2490
```

9. Write a program that converts time from hours to seconds. Ask the user to input the time in hours, and then convert that time into seconds. Remember, 1 hour equals 3600 seconds. (5 Points)

[62]:
```
# Its been a while since I've done error handling, so this is a simple example.
# If this isn't allowed, please advise.

try: # Ensuring that the user input is either a float or integer.
    user_hours = float(input('Enter a number of hours: ')) # Prompting user for␣
 ↪hours.
    convert_hours_to_seconds = user_hours* 3600 # Converting hours to seconds.
    print(f'Your input of {user_hours} hours is equal to␣
 ↪{convert_hours_to_seconds} seconds.') # Printing the result.
except ValueError: # The user input was a string.
    print("Invalid input. Try again.") # Apparently the user does not know how␣
 ↪to follow instructions.
```

```
Your input of 24.0 hours is equal to 86400.0 seconds.
```

10. Evaluate the expression `3 * (4 + 5) - 6 / 2` and display the result. (5 Points)

[63]:
```
expression = (3*(4 + 5) - 6)/(2) # The given expression.

print(expression) # Output the result of the expression.
```

```
10.5
```

11. Assign the string 'Python' to a variable and print it. (5 Points)

[64]:
```
given_string = 'Python' # The given string.

print(given_string) # Printing the string.
```

```
Python
```

12. Calculate the remainder when 22 is divided by 5. (5 Points)

[65]:
```
remainder = 22%5 # The operation.

print(remainder) # Printing the result of the operation.
```

```
2
```

13. Ask the user for two numbers, add them together, and display the sum. (5 Points)

[66]:
```
input_1 = input('Enter first number: ') # First user input.
input_2 = input('Enter second number: ') # Second user input.

sum_of_inputs = int(input_1) + int(input_2) # Sum of the two inputs.
```

```
print(f'The sum of {input_1} and {sum_of_inputs}.') # Output the␣
 ↪result.
```

```
The sum of 555 and 222 is 777.
```

14. Write a Python code snippet that takes user input and prints the type of the input variable. (5 Points)

[67]:
```
user_input_dtype = input('Enter whatever you want: ') # User input.

print(f'Your input of {user_input_dtype} has a data type of␣
 ↪{type(user_input_dtype)}.') # Output the data type of the input.
```

```
Your input of BRUH has a data type of <class 'str'>.
```

15. Calculate the product of 8 and 12, then subtract 4 and display the result. (5 Points)

[68]:
```
# Doing this all within the print statement, just to keep it short.

print(8 * 12 - 4)
```

```
92
```

16. Explain the difference between == and = in Python. (5 Points)

[69]:
```
# The = operator is used to assign values to variables, you use it when you␣
 ↪want to create a variable.
# The == operator is used for boolean comparison, to check for equality.
# I wasn't prompted to give an exmple, but here is one anyway:

x = 10 # Using the = operator to assign the value of 10 to the variable x.
if x == 10: # Using the == operator to check if the value of x is equal to 10.
    print("x is equal to 10.") # Output if the condition is true.
```

```
x is equal to 10.
```

17. Use Python to find the floor division of 100 by 11. (5 Points)

[70]:
```
floor_division_ex = 100//11 # The example.

print(floor_division_ex) # Output the result of the floor division.
```

```
9
```

18. Ask the user for a number, then calculate the square of that number. (5 Points)

[71]:
```
user_input_for_square = input('Enter a number to be squared: ') # User input.
squared_value = int(user_input_for_square) ** 2 # Squaring the input.

print(f'The square of {user_input_for_square} is {squared_value}.')
```

```
The square of 81 is 6561.
```

19. What does the % operator do in Python? Provide an example. (5 Points)

[72]:
```
# The % operator yields the remainder of a division operation.
# If we take 11 mod 5, we can multiply 5 by 2 to get 10, which shows the␣
 ↪remainder is 1.

mod_ex = 11%5 # The example.

print(mod_ex) # Output the result of the modulus operation.
```

```
1
```

20. Using variables, demonstrate how to calculate and print the area of a rectangle (length * width). (5 Points)

[73]:
```
rect_length = 10 # Length of the rectangle.
rect_width = 5 # Width of the rectangle.

rect_area = rect_length * rect_width # Area calculation.

print(f'The area of the rectangle with length {rect_length} and width␣
 ↪{rect_width} is {rect_area}.') # Output the area.
```

```
The area of the rectangle with length 10 and width 5 is 50.
```

21. Write a Python script that asks the user for their first name and last name separately, and then prints their full name. (5 Points)

[74]:
```
first_name = input('Enter your first name: ') # First name input.
last_name = input('Enter your last name: ') # Last name input.

print(f'Hello, {first_name} {last_name}!')
```

```
Hello, Christian Herrera!
```

22. Write a Python script that asks the user for their current age, and then prints what their age will be in 10 years. (5 Points)

[75]:
```
user_age = input('Enter your age: ') # User age input.
age_in_10_years = int(user_age) + 10 # Age in 10 years calculation.
if age_in_10_years > 90:
    print('Wow, your age in 10 years is to high to calculate.') # Output if age␣
 ↪exceeds 90.
else:
    print(f'In 10 years, you will be {age_in_10_years} years old.') # Output␣
 ↪the future age.
```

```
In 10 years, you will be 74 years old.
```

23. Write a Python script that asks the user for a number of minutes, and then converts it to hours and minutes. (10 Points)

[76]:
```
input_in_minutes = input('Enter a number of minutes: ') # User input in minutes.
convert_to_seconds = int(input_in_minutes) * 60 # Conversion to seconds.
print(f'{input_in_minutes} minutes is equal to {convert_to_seconds} seconds.')␣
  ↪# Output the result.
```

```
15 minutes is equal to 900 seconds.
```