



PYTHON ASSIGNMENT SCREENSHOT FILE

Name- Aditya Kumar

SAP ID – 590015145

1. Python Installation & Execution

```
[1]: import sys

#Take two variable one for name and another for course
name = input("Enter your name:")
course_name = input("Enter your course:")

# Print the information
print("Name:", name)
print("Course Name:", course_name)
print("Python Version:", sys.version)

Enter your name: aditya
Enter your course: mca
Name: aditya
Course Name: mca
Python Version: 3.12.4 | packaged by Anaconda, Inc. | (main, Jun 18 2024, 15:03:56) [MSC v.1929 64 bit (AMD64)]
```

2. Using IDLE

```
[9]: a = "HELLO , PYTHON WORLD"
print(a)

HELLO , PYTHON WORLD
```

3. String Operation

```
[2]: #Take two variable first for first name and second for last name
first_name = input("Enter your first name: ")
last_name = input("Enter your last name: ")

#concatenate the last name and first name with space
reversed_name = last_name + " " + first_name

# print the reversed name
print(reversed_name[::-1])

Enter your first name: aditya
Enter your last name: kumar
aytida ramuk
```

4. Numeric Data Types and Conversion Function

```
[3]: num = input("Enter a number: ")

# Convert to integer
integer_num = int(num)
print("Integer:", integer_num)

# Explanation: Integers are whole numbers without decimal points.
# They can be positive, negative, or zero.

# Convert to float
float_num = float(num)
print("Float:", float_num)

# Explanation: Floats are numbers with decimal points. They can represent
# fractional values and have a wider range than integers.

# Convert to complex
complex_num = complex(num)
print("Complex:", complex_num)

# Explanation: Complex numbers are numbers that have a real part and an imaginary part.
# They are represented as a + bi, where a is the real part and b is the imaginary part.

Enter a number: 3
Integer: 3
Float: 3.0
Complex: (3+0j)
```

5. Simple Input and Output(Using the **format** Method)



```
[4]: # Prompt the user to enter the length and width of the rectangle
length = float(input("Enter the length of the rectangle(in cm): "))
width = float(input("Enter the width of the rectangle(in cm): "))

# Calculate the area of the rectangle
area = length * width

# Print the calculated area
print("The area of the rectangle is:", area)

Enter the length of the rectangle(in cm): 3
Enter the width of the rectangle(in cm): 44
The area of the rectangle is: 132.0
```

6. The **%** method & **print** method



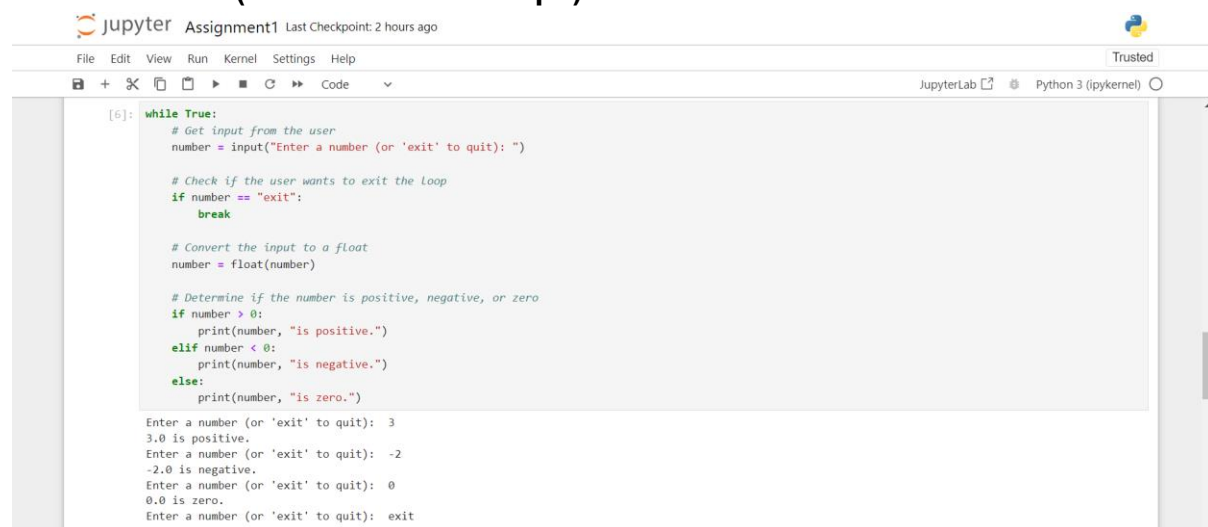
```
[5]: # Get three numbers from the user
num1 = float(input("Enter the first number: "))
num2 = float(input("Enter the second number: "))
num3 = float(input("Enter the third number: "))

# Calculate the average
average = (num1 + num2 + num3) / 3

# Print the average using string formatting
print("The average of the three numbers is: %.2f" % average)

Enter the first number: 3
Enter the second number: 2
Enter the third number: 4
The average of the three numbers is: 3.00
```

7.Control Flow(If Statements & loops)



```
[6]: while True:
    # Get input from the user
    number = input("Enter a number (or 'exit' to quit): ")

    # Check if the user wants to exit the loop
    if number == "exit":
        break

    # Convert the input to a float
    number = float(number)

    # Determine if the number is positive, negative, or zero
    if number > 0:
        print(number, "is positive.")
    elif number < 0:
        print(number, "is negative.")
    else:
        print(number, "is zero.")

Enter a number (or 'exit' to quit): 3
3.0 is positive.
Enter a number (or 'exit' to quit): -2
-2.0 is negative.
Enter a number (or 'exit' to quit): 0
0.0 is zero.
Enter a number (or 'exit' to quit): exit
```

8. Regional and logical Operators



```
[7]: # Get input from the user
num1 = int(input("Enter the first number: "))
num2 = int(input("Enter the second number: "))

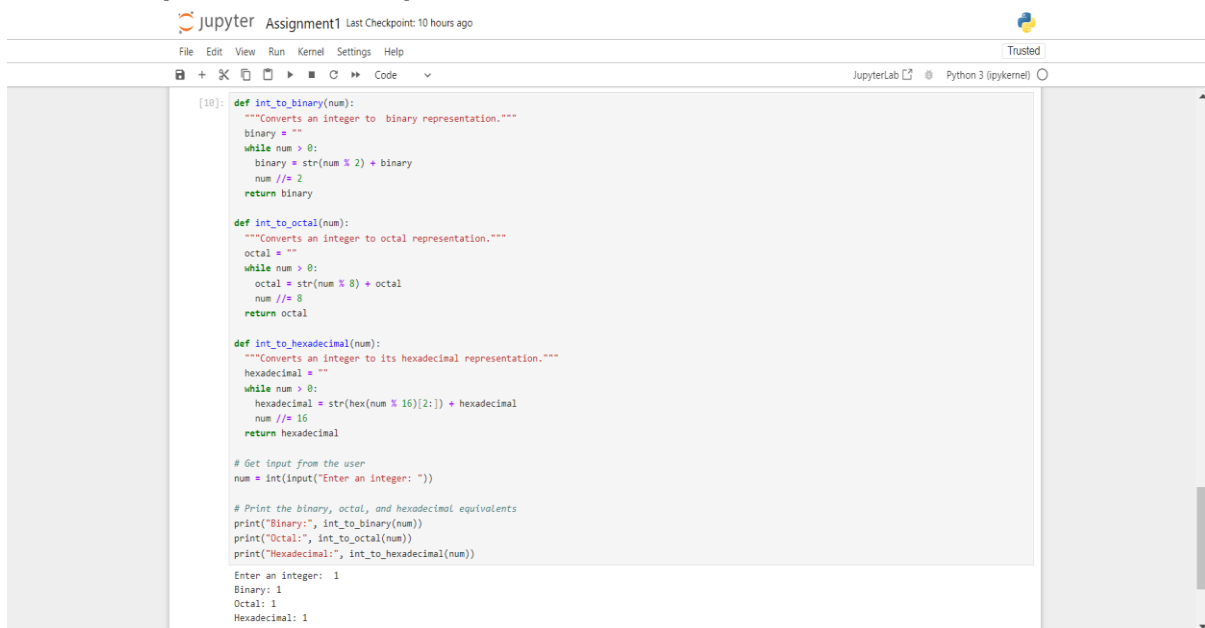
# Check if both numbers are even
if num1 % 2 == 0 and num2 % 2 == 0:
    print("Both numbers are even.")

# Check if both numbers are odd
elif num1 % 2 != 0 and num2 % 2 != 0:
    print("Both numbers are odd.")

# If neither of the above conditions is true, one number is even and the other is odd
else:
    print("One number is even and the other is odd.")

Enter the first number: 3
Enter the second number: 2
One number is even and the other is odd.
```

9. For Loop and Bitwise Operators



The image shows a JupyterLab interface with a Python 3 (ipykernel) environment. The code defines three functions: `int_to_binary`, `int_to_octal`, and `int_to_hexadecimal`. Each function uses a `while` loop to build the string representation of the number in the respective base. The `int_to_hexadecimal` function uses `hex(num % 16)[2:]` to get the hexadecimal digit. The code also includes a user input prompt and prints the results for the input value 1.

```
[10]: def int_to_binary(num):  
    """Converts an integer to binary representation."""  
    binary = ""  
    while num > 0:  
        binary = str(num % 2) + binary  
        num //= 2  
    return binary  
  
    def int_to_octal(num):  
        """Converts an integer to octal representation."""  
        octal = ""  
        while num > 0:  
            octal = str(num % 8) + octal  
            num //= 8  
        return octal  
  
    def int_to_hexadecimal(num):  
        """Converts an integer to its hexadecimal representation."""  
        hexadecimal = ""  
        while num > 0:  
            hexadecimal = str(hex(num % 16)[2:]) + hexadecimal  
            num //= 16  
        return hexadecimal  
  
    # Get input from the user  
    num = int(input("Enter an integer: "))  
  
    # Print the binary, octal, and hexadecimal equivalents  
    print("Binary:", int_to_binary(num))  
    print("Octal:", int_to_octal(num))  
    print("Hexadecimal:", int_to_hexadecimal(num))  
  
    Enter an integer: 1  
    Binary: 1  
    Octal: 1  
    Hexadecimal: 1
```