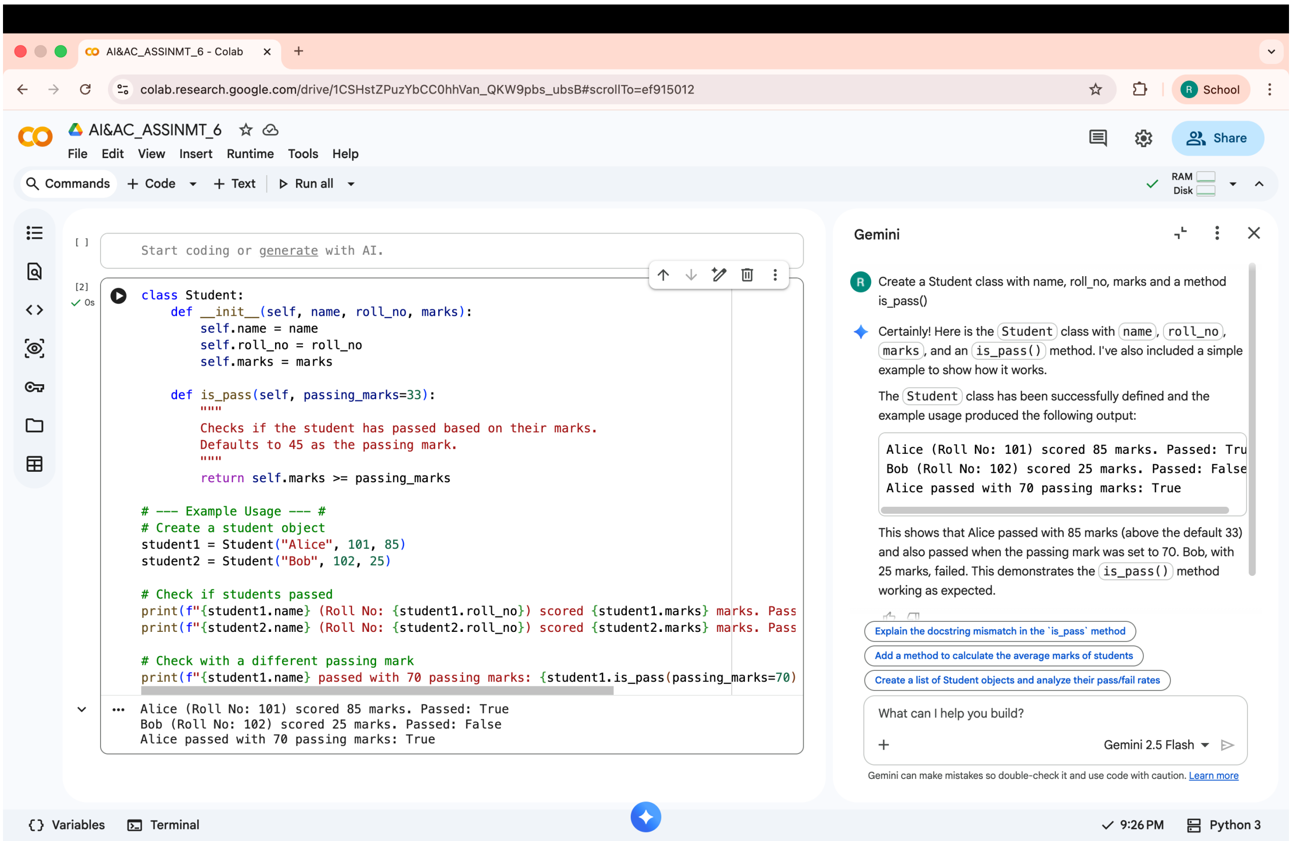
AI ASSISTED CODING - ASSIGNMENT - 6

NAME : R.RONITH REDDY
HTNO : 2303A52280

TASK - 1 : CLASSES – DATA VALIDATION

PROMPT : CREATE A STUDENT CLASS WITH NAME, ROLL_NO, MARKS AND A METHOD IS_PASS()



AI&AC_ASSINMT_6 - Colab
colab.research.google.com/drive/1CSHstZPuzYbCC0hhVan_QKW9pbs_ubsB#scrollTo=ef915012
School
AI&AC_ASSINMT_6
File Edit View Insert Runtime Tools Help
Share
Commands
+ Code
+ Text
Run all
RAM
Disk
Start coding or generate with AI.
[2]
0s
Variables
Terminal
9:26 PM
Python 3

```
class Student:
    def __init__(self, name, roll_no, marks):
        self.name = name
        self.roll_no = roll_no
        self.marks = marks

    def is_pass(self, passing_marks=33):
        """
        Checks if the student has passed based on their marks.
        Defaults to 45 as the passing mark.
        """
        return self.marks >= passing_marks

# --- Example Usage --- #
# Create a student object
student1 = Student("Alice", 101, 85)
student2 = Student("Bob", 102, 25)

# Check if students passed
print(f"{student1.name} (Roll No: {student1.roll_no}) scored {student1.marks} marks. Pass
print(f"{student2.name} (Roll No: {student2.roll_no}) scored {student2.marks} marks. Pass

# Check with a different passing mark
print(f"{student1.name} passed with 70 passing marks: {student1.is_pass(passing_marks=70)
```

```
Alice (Roll No: 101) scored 85 marks. Passed: True
Bob (Roll No: 102) scored 25 marks. Passed: False
Alice passed with 70 passing marks: True
```

Gemini

Create a Student class with name, roll_no, marks and a method is_pass()

Certainly! Here is the `Student` class with `name`, `roll_no`, `marks`, and an `is_pass()` method. I've also included a simple example to show how it works.

The `Student` class has been successfully defined and the example usage produced the following output:

```
Alice (Roll No: 101) scored 85 marks. Passed: Tru
Bob (Roll No: 102) scored 25 marks. Passed: False
Alice passed with 70 passing marks: True
```

This shows that Alice passed with 85 marks (above the default 33) and also passed when the passing mark was set to 70. Bob, with 25 marks, failed. This demonstrates the `is_pass()` method working as expected.

Explain the docstring mismatch in the `is_pass` method
Add a method to calculate the average marks of students
Create a list of Student objects and analyze their pass/fail rates
What can I help you build?
Gemini 2.5 Flash

Gemini can make mistakes so double-check it and use code with caution. Learn more

EXPLANATION :

THE `STUDENT` CLASS STORES STUDENT DETAILS USING A CONSTRUCTOR. THE `IS_PASS()` METHOD CHECKS WHETHER THE STUDENT'S MARKS ARE GREATER THAN OR EQUAL TO 40 AND THEN RETURNS THE RESULT AS PASS OR FAIL STATUS.

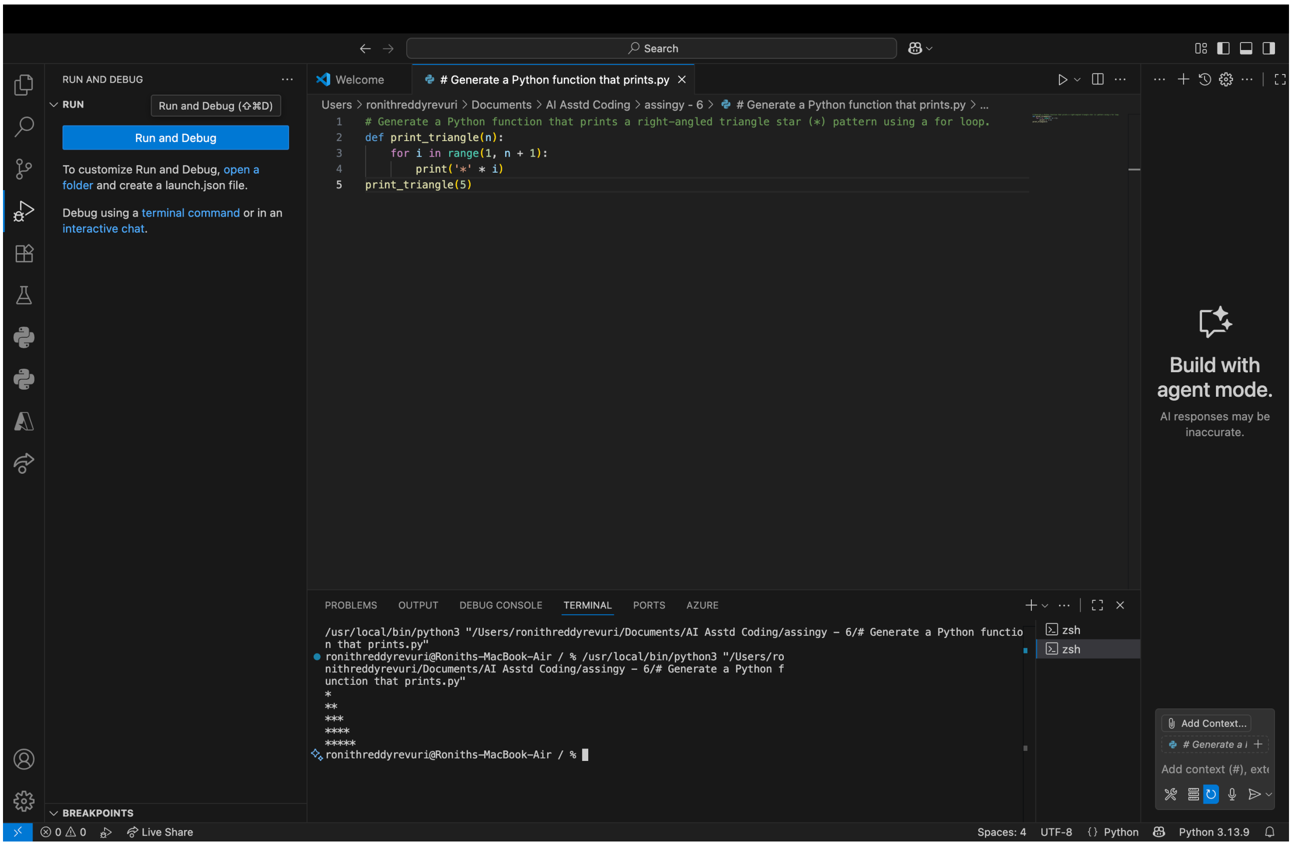
OUTPUT :

ALICE (ROLL NO: 101) SCORED 85 MARKS. PASSED: TRUE
BOB (ROLL NO: 102) SCORED 25 MARKS. PASSED: FALSE
ALICE PASSED WITH 70 PASSING MARKS: TRUE

TASK - 2 : LOOPS – PATTERN GENERATION

PROMPT :

GENERATE A PYTHON FUNCTION THAT PRINTS A RIGHT-ANGLED TRIANGLE STAR (*) PATTERN USING A FOR LOOP.



To customize Run and Debug, open a folder and create a launch.json file.

Debug using a terminal command or in an interactive chat.

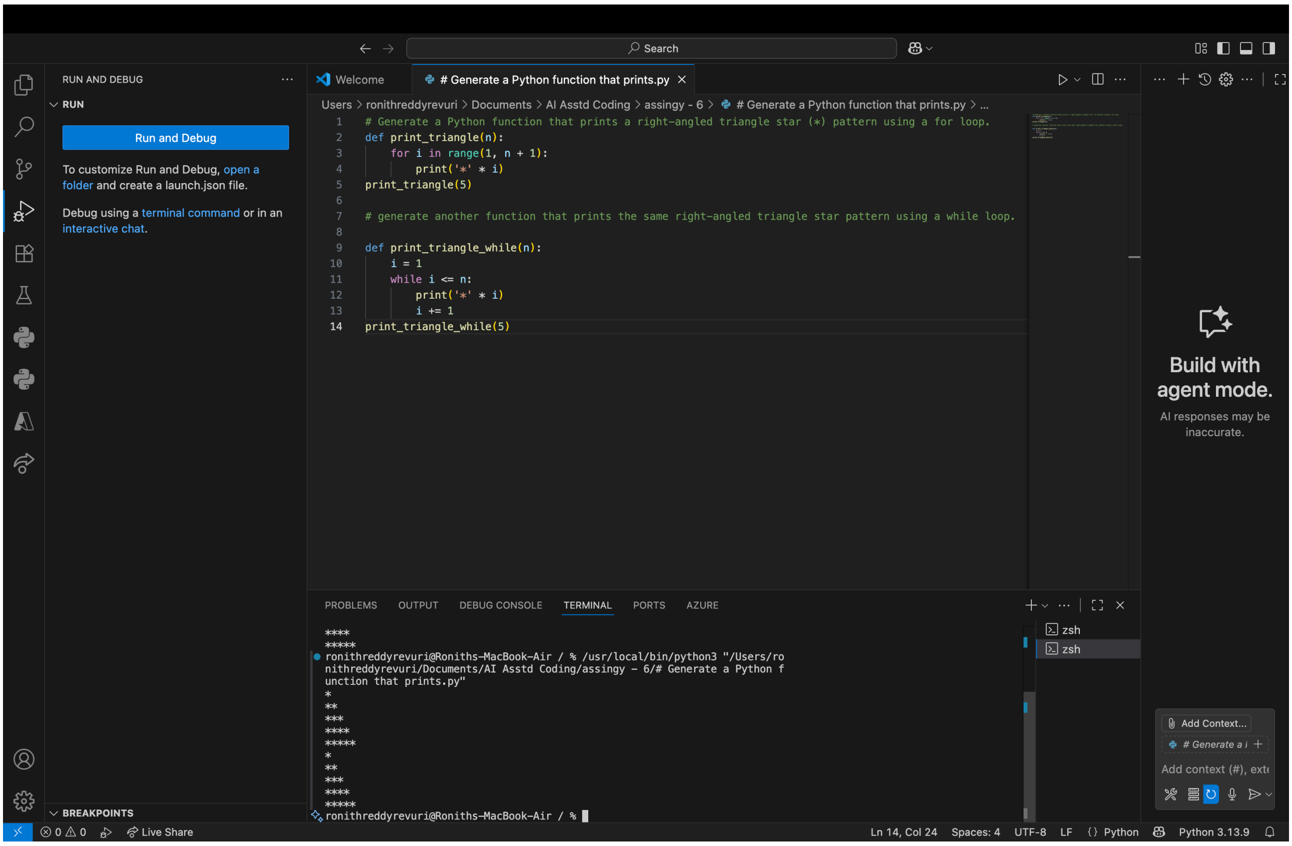
AI responses may be inaccurate.

```python
# Generate a Python function that prints a right-angled triangle star (*) pattern using a for loop.
def print_triangle(n):
    for i in range(1, n + 1):
        print('*' * i)
print_triangle(5)
```

```
/usr/local/bin/python3 "/Users/ronithreddyrevuri/Documents/AI Asstd Coding/assingy - 6/# Generate a Python functio
n that prints.py"
ronithreddyrevuri@Roniths-MacBook-Air / % /usr/local/bin/python3 "/Users/ro
nithreddyrevuri/Documents/AI Asstd Coding/assingy - 6/# Generate a Python f
unction that prints.py"
*
**
***
****
*****
ronithreddyrevuri@Roniths-MacBook-Air / % 
```

PROMPT - 2 :

GENERATE ANOTHER FUNCTION THAT PRINTS THE SAME RIGHT-ANGLED TRIANGLE STAR PATTERN USING A WHILE LOOP.



To customize Run and Debug, open a folder and create a launch.json file.

Debug using a terminal command or in an interactive chat.

AI responses may be inaccurate.

```python
# Generate a Python function that prints a right-angled triangle star (*) pattern using a for loop.
def print_triangle(n):
    for i in range(1, n + 1):
        print('*' * i)
print_triangle(5)

# generate another function that prints the same right-angled triangle star pattern using a while loop.

def print_triangle_while(n):
    i = 1
    while i <= n:
        print('*' * i)
        i += 1
print_triangle_while(5)
```

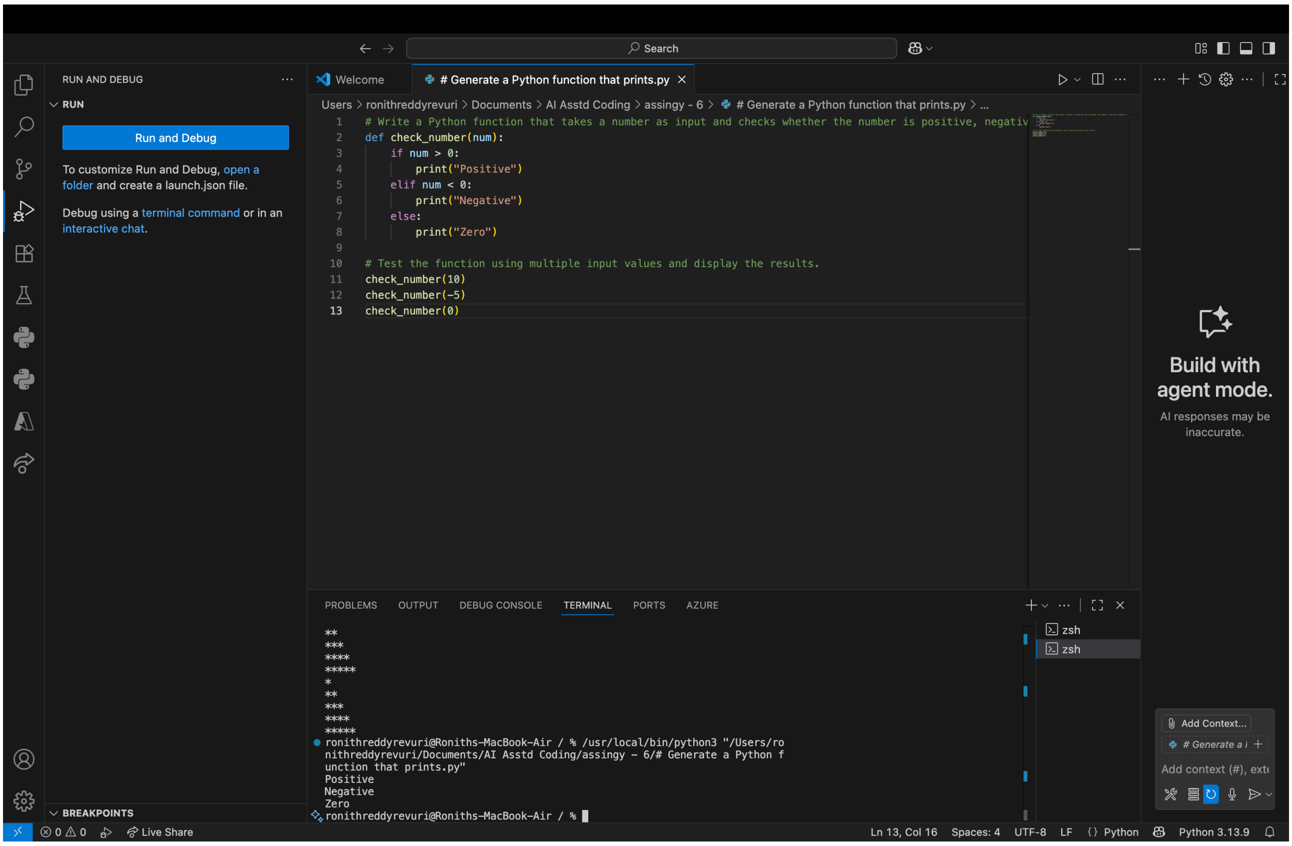
```
****
*****
ronithreddyrevuri@Roniths-MacBook-Air / % /usr/local/bin/python3 "/Users/ro
nithreddyrevuri/Documents/AI Asstd Coding/assingy - 6/# Generate a Python f
unction that prints.py"
*
**
***
****
*****
*
**
***
****
*****
ronithreddyrevuri@Roniths-MacBook-Air / % 
```

EXPLANATION :

THIS PROGRAM PRINTS A RIGHT-ANGLED TRIANGLE USING STAR SYMBOLS.
FIRST, THE PATTERN IS CREATED USING A FOR LOOP, AND THEN THE SAME PATTERN IS CREATED USING A WHILE LOOP.
BOTH LOOPS INCREASE THE NUMBER OF STARS IN EACH ROW.
THIS SHOWS THAT DIFFERENT LOOP TYPES CAN PRODUCE THE SAME RESULT.

TASK - 3 : CONDITIONAL STATEMENTS – NUMBER ANALYSIS

PROMPT :
WRITE A PYTHON FUNCTION THAT TAKES A NUMBER AS INPUT AND CHECKS WHETHER THE NUMBER IS POSITIVE, NEGATIVE, OR ZERO USING IF-ELIF-ELSE CONDITIONAL STATEMENTS. TEST THE FUNCTION USING MULTIPLE INPUT VALUES AND DISPLAY THE RESULTS.



To customize Run and Debug, open a folder and create a launch.json file.

Debug using a terminal command or in an interactive chat.

```python
# Write a Python function that takes a number as input and checks whether the number is positive, negativ
def check_number(num):
    if num > 0:
        print("Positive")
    elif num < 0:
        print("Negative")
    else:
        print("Zero")

# Test the function using multiple input values and display the results.
check_number(10)
check_number(-5)
check_number(0)
```

```
**
***
****
*****
*
**
***
****
*****
ronithreddyrevuri@Roniths-MacBook-Air / % /usr/local/bin/python3 "/Users/ro
nithreddyrevuri/Documents/AI Asstd Coding/assingy - 6/# Generate a Python f
unction that prints.py"
Positive
Negative
Zero
ronithreddyrevuri@Roniths-MacBook-Air / % 
```

EXPLANATION :

THIS TASK USES **IF-ELIF-ELSE CONDITIONAL STATEMENTS** TO ANALYZE A GIVEN NUMBER.

- **IF THE NUMBER IS GREATER THAN ZERO, IT IS CLASSIFIED AS POSITIVE.**
- **IF THE NUMBER IS LESS THAN ZERO, IT IS CLASSIFIED AS NEGATIVE.**
- **IF THE NUMBER IS EQUAL TO ZERO, IT IS CLASSIFIED AS ZERO.**

THE FUNCTION IS TESTED WITH MULTIPLE INPUT VALUES TO ENSURE THAT ALL CONDITIONS ARE HANDLED CORRECTLY.
THIS TASK DEMONSTRATES PROPER DECISION-MAKING LOGIC USING CONDITIONAL STATEMENTS IN PYTHON.

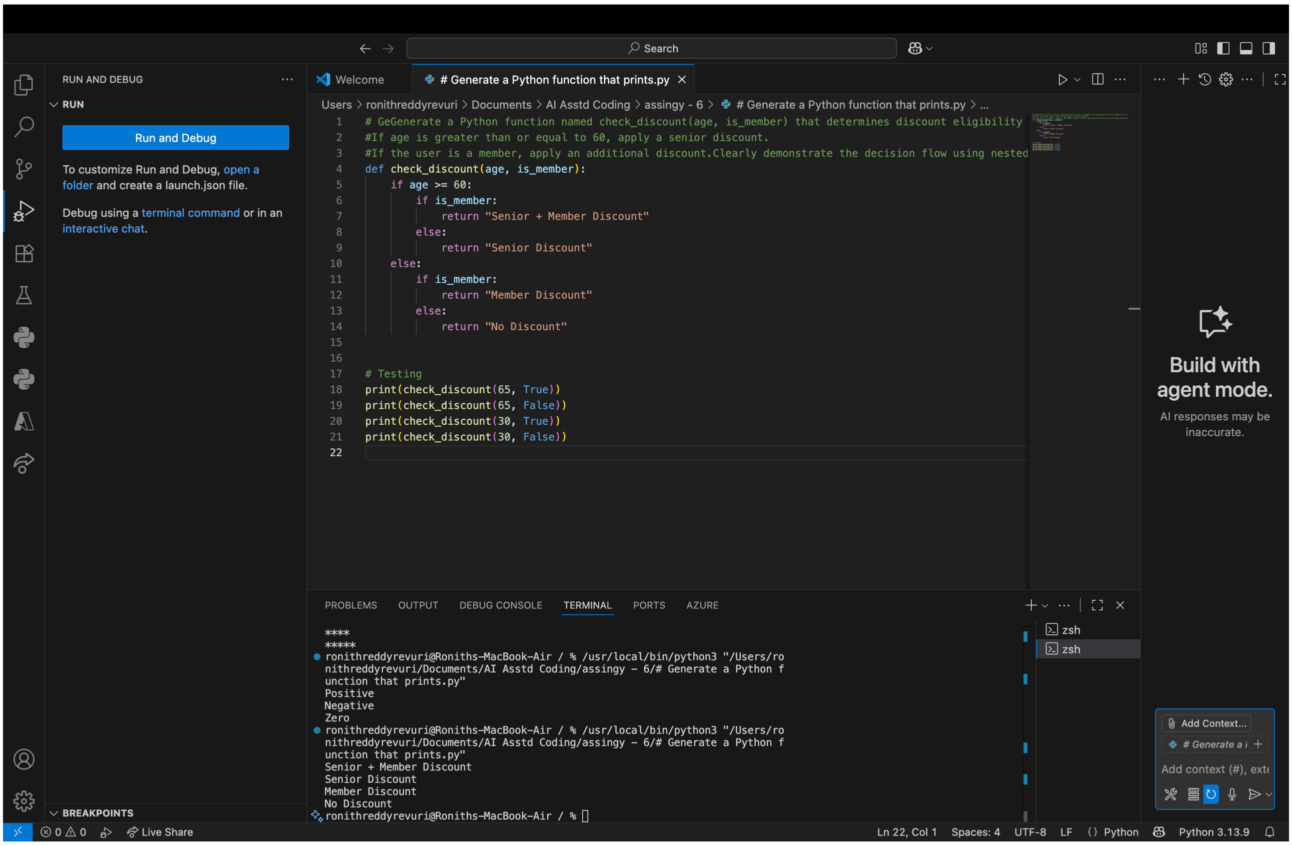
TASK - 4 : NESTED CONDITIONAL STATEMENTS

PROMPT :

GENERATE A PYTHON FUNCTION NAMED `CHECK_DISCOUNT(AGE, IS_MEMBER)` THAT DETERMINES DISCOUNT ELIGIBILITY USING NESTED IF STATEMENTS BASED ON THE FOLLOWING CONDITIONS:

- **IF AGE IS GREATER THAN OR EQUAL TO 60, APPLY A SENIOR DISCOUNT.**
- **IF THE USER IS A MEMBER, APPLY AN ADDITIONAL DISCOUNT.**
  CLEARLY DEMONSTRATE THE DECISION FLOW USING NESTED CONDITIONALS.

EXPLANATION:
THIS PROGRAM CHECKS DISCOUNT ELIGIBILITY USING NESTED IF STATEMENTS.
IT FIRST CHECKS THE AGE FOR A SENIOR DISCOUNT AND THEN CHECKS MEMBERSHIP FOR AN ADDITIONAL DISCOUNT.
THIS HELPS IN MAKING CLEAR AND STRUCTURED DECISIONS.

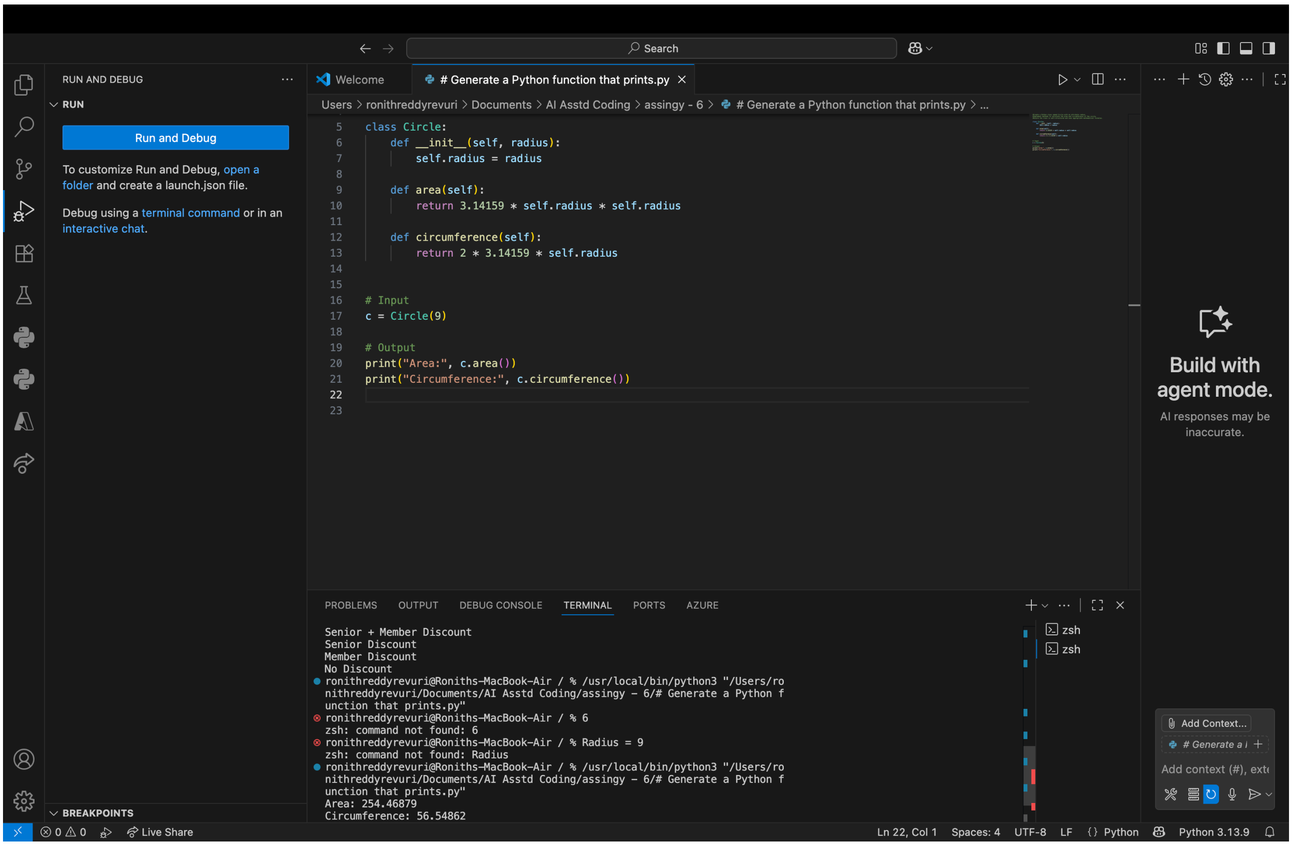

Search
RUN AND DEBUG
RUN
Run and Debug
To customize Run and Debug, open a folder and create a launch.json file.
Debug using a terminal command or in an interactive chat.
BREAKPOINTS
Welcome
# Generate a Python function that prints.py
Users > ronithreddyrevuri > Documents > AI Asstd Coding > assingy - 6 > # Generate a Python function that prints.py > ...
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
AZURE
zsh
zsh
Build with agent mode.
AI responses may be inaccurate.
Add Context...
# Generate a i
Add context (#), exte
0 ⚠ 0
Live Share
Ln 22, Col 1
Spaces: 4
UTF-8
LF
{} Python
Python 3.13.9

```python
# GeGenerate a Python function named check_discount(age, is_member) that determines discount eligibility
#If age is greater than or equal to 60, apply a senior discount.
#If the user is a member, apply an additional discount.Clearly demonstrate the decision flow using nested
def check_discount(age, is_member):
    if age >= 60:
        if is_member:
            return "Senior + Member Discount"
        else:
            return "Senior Discount"
    else:
        if is_member:
            return "Member Discount"
        else:
            return "No Discount"


# Testing
print(check_discount(65, True))
print(check_discount(65, False))
print(check_discount(30, True))
print(check_discount(30, False))

```

```
****
*****
ronithreddyrevuri@Roniths-MacBook-Air / % /usr/local/bin/python3 "/Users/ro
nithreddyrevuri/Documents/AI Asstd Coding/assingy - 6/# Generate a Python f
unction that prints.py"
Positive
Negative
Zero
ronithreddyrevuri@Roniths-MacBook-Air / % /usr/local/bin/python3 "/Users/ro
nithreddyrevuri/Documents/AI Asstd Coding/assingy - 6/# Generate a Python f
unction that prints.py"
Senior + Member Discount
Senior Discount
Member Discount
No Discount
ronithreddyrevuri@Roniths-MacBook-Air / % 
```

TASK - 5 : CLASS – MATHEMATICAL OPERA

PROMPT : CREATE A PYTHON CLASS NAMED `CIRCLE` WITH AN ATTRIBUTE `RADIUS`. IMPLEMENT METHODS TO CALCULATE THE AREA AND CIRCUMFERENCE OF THE CIRCLE. ENSURE THE CLASS IS WELL-STRUCTURED AND USES APPROPRIATE MATHEMATICAL FORMULAS.


RUN AND DEBUG
RUN
Run and Debug
Welcome
# Generate a Python function that prints.py
Users > ronithreddyrevuri > Documents > AI Asstd Coding > assingy - 6 > # Generate a Python function that prints.py > ...
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
AZURE
zsh
zsh
Build with agent mode.
AI responses may be inaccurate.
Add Context...
BREAKPOINTS
Live Share
Ln 22, Col 1
Spaces: 4
UTF-8
LF
Python
Python 3.13.9

To customize Run and Debug, open a folder and create a launch.json file.

Debug using a terminal command or in an interactive chat.

```python
class Circle:
    def __init__(self, radius):
        self.radius = radius

    def area(self):
        return 3.14159 * self.radius * self.radius

    def circumference(self):
        return 2 * 3.14159 * self.radius


# Input
c = Circle(9)

# Output
print("Area:", c.area())
print("Circumference:", c.circumference())
```

```
Senior + Member Discount
Senior Discount
Member Discount
No Discount
ronithreddyrevuri@Roniths-MacBook-Air / % /usr/local/bin/python3 "/Users/ro
nithreddyrevuri/Documents/AI Asstd Coding/assingy - 6/# Generate a Python f
unction that prints.py"
ronithreddyrevuri@Roniths-MacBook-Air / % 6
zsh: command not found: 6
ronithreddyrevuri@Roniths-MacBook-Air / % Radius = 9
zsh: command not found: Radius
ronithreddyrevuri@Roniths-MacBook-Air / % /usr/local/bin/python3 "/Users/ro
nithreddyrevuri/Documents/AI Asstd Coding/assingy - 6/# Generate a Python f
unction that prints.py"
Area: 254.46879
Circumference: 56.54862
```

EXPLANATION :

THIS PROGRAM USES A CIRCLE CLASS TO CALCULATE AREA AND CIRCUMFERENCE.
THE RADIUS IS GIVEN AS INPUT, AND STANDARD FORMULAS ARE USED FOR CALCULATIONS.
IT SHOWS HOW CLASSES AND METHODS WORK IN PYTHON.