Seedling (Basic)

December 6, 2024

0.0.1 Task 1 Class Practice Code: Introduction to Python Basics

Focusing on foundational Python concepts. The Jupyter Notebook will cover **variables**, **data types**, and **basic operations**.

CoursePlan

1. **Introduction to Python Syntax**
   - Print statements
   - Variables and data types
2. **Perform Arithmetic Operations**
   - Addition, subtraction, multiplication, and division
   - Using variables in expressions
3. **Manipulate Strings**
   - String concatenation and repetition
   - String methods (like `.upper()`, `.lower()`, etc.)

0.0.2 Practice Code

[1]:
```
#### **Task 1: Print Statements and Variables**
```

[2]:
```
# Print a welcome message
print("Welcome to Python Basics!")
```

```
Welcome to Python Basics!
```

[3]:
```
# Define variables
name = "Satyanarayana"
age = 28
is_intern = True

# Display variable values
print("Name:", name)
print("Age:", age)
print("Is Intern:", is_intern)
```

```
Name: Satyanarayana
Age: 28
Is Intern: True
```

[ ]:

[4]:
```python
#### **Task 2: Arithmetic Operations**
# Define two numbers
num1 = 10
num2 = 5

# Perform arithmetic operations
addition = num1 + num2
subtraction = num1 - num2
multiplication = num1 * num2
division = num1 / num2

# Print the results
print("Addition:", addition)
print("Subtraction:", subtraction)
print("Multiplication:", multiplication)
print("Division:", division)
```

```
Addition: 15
Subtraction: 5
Multiplication: 50
Division: 2.0
```

[ ]:

[5]:
```python
#### **Task 3: String Manipulation**
# Define strings
greeting = "Hello"
name = "Mainflow"

# Concatenate strings
message = greeting + ", " + name + "!"
print(message)
```

```
Hello, Mainflow!
```

[ ]:

[6]:
```python
# String repetition
repeated = name * 3
print("Repeated Name:", repeated)

# Use string methods
print("Uppercase:", name.upper())
```

```
print("Lowercase:", name.lower())
```

```
Repeated Name: MainflowMainflowMainflow
Uppercase: MAINFLOW
Lowercase: mainflow
```

[ ]:

Exercises

1. **Simple Calculator**:
   - Write a script to take two numbers as input and perform all arithmetic operations.
2. **String Playground**:
   - Write a program to input a string and display:
     - Its length
     - Its first and last characters
     - The string reversed ##

[7]:
```
############################################################
```

[8]:
```
## Exercise 1: Simple Calculator
#A script that takes two numbers as input and performs all arithmetic␣
 ↪operations.
# Input two numbers from the user

num1 = float(input("Enter the first number: "))
num2 = float(input("Enter the second number: "))

# Perform arithmetic operations
addition = num1 + num2
subtraction = num1 - num2
multiplication = num1 * num2
division = num1 / num2 if num2 != 0 else "Division by zero not allowed"

# Display the results
print("\nResults:")
print(f"Addition: {num1} + {num2} = {addition}")
print(f"Subtraction: {num1} - {num2} = {subtraction}")
print(f"Multiplication: {num1} * {num2} = {multiplication}")
print(f"Division: {num1} / {num2} = {division}")
```

```
Enter the first number:  8
Enter the second number:  7


Results:
Addition: 8.0 + 7.0 = 15.0
Subtraction: 8.0 - 7.0 = 1.0
```

```
Multiplication: 8.0 * 7.0 = 56.0
Division: 8.0 / 7.0 = 1.1428571428571428
```

[ ]:

[7]:
```python
###Exercise 2: String Playground

# Input a string from the user
user_string = input("Enter a string: ")

# Display string details
print("\nString Analysis:")
print(f"Length of the string: {len(user_string)}")
print(f"First character: {user_string[0]}")
print(f"Last character: {user_string[-1]}")
print(f"Reversed string: {user_string[::-1]}")
```

```
Enter a string:  Main Flow Services And Technologies - Task 1


String Analysis:
Length of the string: 44
First character: M
Last character: 1
Reversed string: 1 ksaT - seigolonhceT dnA secivreS wolF niaM
```

[ ]:

[ ]:

0.0.3 Practice

1. **Enhance the Simple Calculator**:
   - Add modulus (%) and exponentiation (**) operations.
   - Handle cases where both numbers are zero.
2. **Enhance the String Playground**:
   - Check if the string is a palindrome (reads the same backward as forward).
   - Count the number of vowels in the string.

[ ]: