Blossom (Intermediate)

December 6, 2024

0.0.1 Task 3 Class Practice Code: Functions and Data Structures

---

Class Plan

1. **Functions**
   - Defining and calling functions
   - Function parameters and return values
   - Default arguments and keyword arguments
2. **Data Structures**
   - Lists: operations and methods
   - Dictionaries: creating, accessing, and modifying
   - Tuples: immutable sequences
   - Sets: unique elements and set operations
3. **Combine Concepts**
   - Solve problems using functions and data structures.

0.0.2 Practice Code

[ ]:
```
```

[1]:
```
#### **Task 1: Functions**
# Function to calculate the square of a number
def square(num):
    return num ** 2

print("Square of 5:", square(5))
```

```
Square of 5: 25
```

[ ]:
```
```

[2]:
```
# Function with default arguments
def greet(name="Guest"):
    print(f"Hello, {name}!")

greet()  # Default name
```

```
greet("Mainflow")  # Custom name
```

```
Hello, Guest!
Hello, Mainflow!
```

[ ]:

[3]:
```
# Function to calculate factorial
def factorial(n):
    if n == 0 or n == 1:
        return 1
    else:
        return n * factorial(n - 1)

print("Factorial of 5:", factorial(5))
```

```
Factorial of 5: 120
```

[ ]:

[4]:
```
#### **Task 2: Lists**
# Create and manipulate a list
fruits = ["apple", "banana", "cherry"]
print("Original list:", fruits)

# Add an element
fruits.append("orange")
print("After adding an element:", fruits)

# Remove an element
fruits.remove("banana")
print("After removing an element:", fruits)

# Access elements
print("First fruit:", fruits[0])

# Iterate over the list
for fruit in fruits:
    print("Fruit:", fruit)
```

```
Original list: ['apple', 'banana', 'cherry']
After adding an element: ['apple', 'banana', 'cherry', 'orange']
After removing an element: ['apple', 'cherry', 'orange']
First fruit: apple
Fruit: apple
Fruit: cherry
Fruit: orange
```

[ ]:

[5]:
```
#### **Task 3: Dictionaries**
# Create and access a dictionary
student = {"name": "John", "age": 21, "grade": "A"}
print("Student dictionary:", student)

# Access a value
print("Name:", student["name"])

# Add a new key-value pair
student["major"] = "Computer Science"
print("After adding a new key-value pair:", student)

# Iterate through keys and values
for key, value in student.items():
    print(f"{key}: {value}")
```

```
Student dictionary: {'name': 'John', 'age': 21, 'grade': 'A'}
Name: John
After adding a new key-value pair: {'name': 'John', 'age': 21, 'grade': 'A',
'major': 'Computer Science'}
name: John
age: 21
grade: A
major: Computer Science
```

[ ]:
```

```

[6]:
```
#### **Task 4: Tuples**
# Create and access a tuple
colors = ("red", "green", "blue")
print("Colors tuple:", colors)
print("First color:", colors[0])
```

```
Colors tuple: ('red', 'green', 'blue')
First color: red
```

[7]:
```
# Tuples are immutable
# colors[0] = "yellow"  # This will cause an error
#### **Task 5: Sets**
# Create and manipulate a set
numbers = {1, 2, 3, 4}
print("Original set:", numbers)

# Add an element
numbers.add(5)
print("After adding an element:", numbers)

# Remove an element
```

```
numbers.remove(3)
print("After removing an element:", numbers)
```

```
Original set: {1, 2, 3, 4}
After adding an element: {1, 2, 3, 4, 5}
After removing an element: {1, 2, 4, 5}
```

[ ]:

[8]:
```
# Set operations
set1 = {1, 2, 3}
set2 = {3, 4, 5}
print("Union:", set1 | set2)
print("Intersection:", set1 & set2)
print("Difference:", set1 - set2)
```

```
Union: {1, 2, 3, 4, 5}
Intersection: {3}
Difference: {1, 2}
```

[ ]:

0.0.3 Practice

1. **Grade Calculator**:
   - Write a function to calculate the grade based on a list of scores.
   - Example: Input: `[80, 90, 75]`, Output: `"Average Score: 81.67, Grade: B"`
2. **Word Frequency Counter**:
   - Write a program to count the frequency of each word in a given sentence using a dictionary.
3. **Unique Elements**:
   - Write a function that takes a list as input and returns a list of unique elements using a set.
4. **Employee Management**:
   - Create a dictionary to store employee details (ID, Name, Salary).
   - Write functions to:
     - Add a new employee
     - Update an employee's salary
     - Display all employee details

[ ]: