Code Gemini

+91 9373610956
www.codegemini.in
codegeminiedtech@gmail.com

Assignment 1: Python Basics and Data Types

1. **Variable Manipulation:**
   - **Question**: Define variables to store a student's name, age, GPA, and list of subjects. Convert age from years to months and display the result. Concatenate and print a sentence using all the defined variables.
   - **Answer**:

```
# Define variables
name = "John Doe"
age = 20  # in years
gpa = 3.8
subjects = ["Math", "Science", "English"]

# Convert age to months
age_in_months = age * 12
print(f"{name} is {age_in_months} months old, has a GPA of {gpa},
and studies {', '.join(subjects)}.")
```

2. **Type Conversion Practice:**
   - **Question**: Convert a float to an integer, and explain what happens to the decimal part.
   - **Answer**:

```
float_number = 5.7
int_number = int(float_number)
print(int_number)  # Output: 5
# Explanation: The decimal part (.7) is truncated, not rounded.
```

Assignment 2: Data Structures - List, Tuple, Set, and Dictionary

1. **Working with Lists:**
   - **Question**: Create a list of integers, append a new number, insert a number at a specific index, remove duplicates, and print the final list.
   - **Answer**:

```
numbers = [1, 2, 3, 2, 4]
numbers.append(5)
numbers.insert(2, 10)
unique_numbers = list(set(numbers))  # Remove duplicates
print(unique_numbers)  # Output may vary since set order is not
guaranteed
```

2. **Tuple Manipulation:**
   - **Question**: Define a tuple of 5 colors, try adding a new color (explain why it fails), convert it to a list, add the color, and convert it back to a tuple.

+91 9373610956
www.codegemini.in
codegeminiedtech@gmail.com

- **Answer**:

```
colors = ("red", "blue", "green", "yellow", "purple")

# Trying to add a color (will cause an error since tuples are
immutable)
# colors.append("orange")  # This would raise an AttributeError

# Convert tuple to list, add a color, and convert back to tuple
color_list = list(colors)
color_list.append("orange")
colors = tuple(color_list)
print(colors)  # Output: ('red', 'blue', 'green', 'yellow',
'purple', 'orange')
```

3. **Set Operations:**
   - **Answer**:

```
python
Copy code
set1 = {1, 2, 3, 4}
set2 = {3, 4, 5, 6}

# Union
print(set1.union(set2))  # Output: {1, 2, 3, 4, 5, 6}

# Intersection
print(set1.intersection(set2))  # Output: {3, 4}

# Difference
print(set1.difference(set2))  # Output: {1, 2}

# Remove an element and add a new one
set1.discard(1)
set1.add(7)
print(set1)  # Output: {2, 3, 4, 7}
```

4. **Dictionary Practice:**
   - **Answer**:

```
library = {
    "The Great Gatsby": 5,
    "To Kill a Mockingbird": 2,
    "1984": 4
}

# Update the count of a book
library["1984"] -= 1  # One copy borrowed
print(library)

# Add a new book
```

+91 9373610956
www.codegemini.in
codegeminiedtech@gmail.com

```
library["Brave New World"] = 3
print(library)
```

Mini Project: Personal Library Management System

1. **Implementation:**

```
python
Copy code
library = {
    "The Great Gatsby": ["F. Scott Fitzgerald", 1925, "Available"],
    "To Kill a Mockingbird": ["Harper Lee", 1960, "Available"],
}

print(library)
```

Practice Questions - Answers

1. **Data Types and Variables:**
   - **Difference between `int` and `float`**: `int` represents whole numbers, while `float` includes decimal points.
   - **Mutable vs. Immutable Data Types**: Mutable types (like lists) can be modified in place, while immutable types (like tuples) cannot be altered after creation.
2. **Lists and Tuples:**
   - **Difference between Lists and Tuples**: Lists are mutable and allow changes, while tuples are immutable.
   - **Code to remove even numbers from a list**:

```
numbers = [1, 2, 3, 4, 5, 6]
odd_numbers = [num for num in numbers if num % 2 != 0]
print(odd_numbers)  # Output: [1, 3, 5]
```

3. **Dictionaries:**
   - **Keys vs. Values in a Dictionary**: Keys are unique identifiers for values, while values are the data associated with keys.
   - **Code to merge two dictionaries**:

```
dict1 = {"a": 1, "b": 2}
dict2 = {"b": 3, "c": 4}
merged_dict = {**dict1, **dict2}
print(merged_dict)  # Output: {'a': 1, 'b': 3, 'c': 4}
```