MAIN FLOW SERVICES AND TECHNOLOGIES

DATA ANALYTICS

INTERSHIP BATCH 2 AUG - 5 OCT 2024

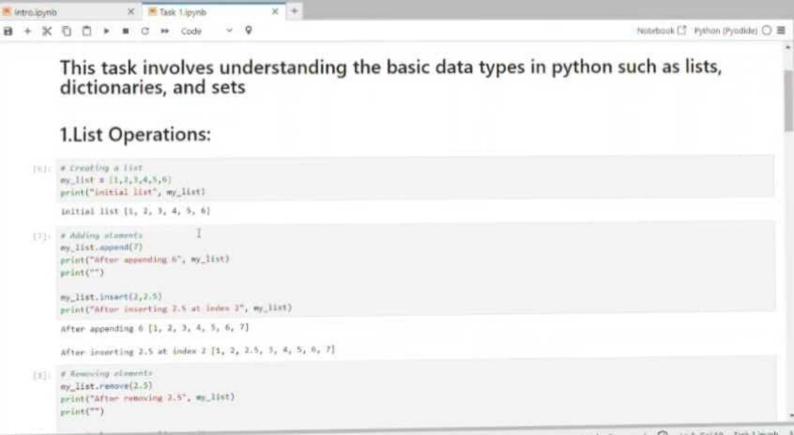   NAME : VAISHNAVI DEVI
    EMAIL : vaishnavidevi2322@gmail.com

This task involves understanding the basic data types in python such as lists, dictionaries, and sets

1.List Operations:

```
# Creating a list
my_list = [1,2,3,4,5,6]
print("initial list", my_list)

initial list [1, 2, 3, 4, 5, 6]
```

```
# Adding elements
my_list.append(7)
print("After appending 6", my_list)
print("")

my_list.insert(2,2.5)
print("After inserting 2.5 at index 2", my_list)
```

This task involves understanding the basic data types in python such as lists, dictionaries, and sets

1.List Operations:

```
[6]: # Creating a list
     my_list = [1,2,3,4,5,6]
     print("initial list", my_list)

     initial list [1, 2, 3, 4, 5, 6]
```

```
[7]: # Adding elements
     my_list.append(7)
     print("After appending 6", my_list)
     print("")

     my_list.insert(2,2.5)
     print("After inserting 2.5 at index 2", my_list)

     After appending 6 [1, 2, 3, 4, 5, 6, 7]

     After inserting 2.5 at index 2 [1, 2, 2.5, 3, 4, 5, 6, 7]
```

```
[8]: # Removing elements
     my_list.remove(2.5)
     print("After removing 2.5", my_list)
     print("")
```

```python
# Removing elements
my_list.remove(2.5)
print("After removing 2.5", my_list)
print("")

popped_element = my_list.pop()
print("After popping an element", my_list)
print("popped element",popped_element)
```

```
After removing 2.5 [1, 2, 3, 4, 5, 6, 7]

After popping an element [1, 2, 3, 4, 5, 6]
popped element 7
```

```python
# Modifying elements
my_list[1] = 1.5
print("After modifying the element at index 1", my_list)
```

```
After modifying the element at index 1 [1, 1.5, 3, 4, 5, 6]
```

```python
# displaying the element index value
position = my_list.index(4)
print("displaying the element(4) index value", position)
print("")

#getting to know the datatype used
print("datatype used here is", type(my_list))
```

```
displaying the element(4) index value 3

datatype used here is <class 'list'>
```

2.Dictionary Operations

```python
[11]: # Creating a dictionary (STUDENT RECORD)
      my_dict = {501:"Arjun",
                 502:"Franklin",
                 503:"Vijay",
                 504:"Disha",
                 505:"Shiva"}
      print("Student list", my_dict)
```

```
Student list {501: 'Arjun', 502: 'Franklin', 503: 'Vijay', 504: 'Disha', 505: 'Shiva'}
```

```python
[12]: # adding elements
      my_dict["506"] = "Marini"
      print("After adding new student 'Marini'", my_dict)
      print("")
```

```
After adding new student 'Marini' {501: 'Arjun', 502: 'Franklin', 503: 'Vijay', 504: 'Disha', 505: 'Shiva', '506': 'Marini'}
```

```python
[ ]: # Removing elements
     my_dict = {501: "Arjuin",
                502: "Franklin",
                503: "Vijay",
                504: "Disha",
                505: "Shiva",
                506: "Marini"}
     del my_dict[504]
     print("After deleting student at roll.no 504['Disha']", my_dict)
     print("")
```

▶   ■   C   ⏩   Code   ⌄   ⚲

```python
# Removing elements
my_dict = {501: 'Arjuin',
           502: 'Franklin',
           503: 'Vijay',
           504: 'Disha',
           505: 'Shiva',
           506: 'Harini'}
del my_dict[504]
print("After deleting student at roll.no 504['Disha']", my_dict)
print("")

removed_value = my_dict.pop(502)
print("After popping student 'Franklin'", my_dict)
print("popped value", removed_value)
```

After deleting student at roll.no 504['Disha'] {501: 'Arjuin', 502: 'Franklin', 503: 'Vijay', 505: 'Shiva', 506: 'Harini'}

After popping student 'Franklin' {501: 'Arjuin', 503: 'Vijay', 505: 'Shiva', 506: 'Harini'}
popped value Franklin

```python
# Modifying elements
my_dict[504] = 'Ramesh'
print("After modifying the value", my_dict)
```

After modifying the value {501: 'Arjuin', 503: 'Vijay', 505: 'Shiva', 506: 'Harini', 504: 'Ramesh'}

```python
# getting to know the datatype used
print("datatype used here is", type(my_dict))
```

datatype used here is <class 'dict'>

3. Set Operations

```
[5]: # Creating a set
     my_set = {1,2,3,4,5}
     print("inital setr", my_set)

     inital setr {1, 2, 3, 4, 5}
```

```
[6]: # Adding elements
     my_set.add(6)
     print("After Adding 6", my_set)

     After Adding 6 {1, 2, 3, 4, 5, 6}
```

```
[7]: # Removing elements
     my_set.remove(3)
     print("After removing 3", my_set)
     print("")

     popped_element = my_set.pop()
     print("After popping an element", my_set)
     print("popped element", popped_element)

     After removing 3 {1, 2, 4, 5, 6}

     After popping an element {2, 4, 5, 6}
     popped element 1
```

```
[8]: # Set do not support indexing, so we can't modify elements directly
     #but we can remove and ten add a new element to simulate modification
     my_set.remove(2)
     my_set.add(20)
```

🖫 + ✂ ▢ ◻ ▸ ■ C ↔ Code   ∨ ⚲

```
[7]  # Removing elements
     my_set.remove(3)
     print("After removing 3", my_set)
     print("")

     popped_element = my_set.pop()
     print("After popping an element", my_set)
     print("popped element", popped_element)
```

After removing 3 (1, 2, 4, 5, 6)

After popping an element (2, 4, 5, 6)
popped element 1

```
[8]  # Set do not support indexing, so we can't modify elements directly
     #but we can remove and fun add a new element to simulate modification
     my_set.remove(2)
     my_set.add(20)
     print("After modifying (removing 2 and adding 20)",my_set)
```

After modifying (removing 2 and adding 20) (4, 5, 6, 20)

```
[ ]
```