Programming with Python April 04 2025

Note: Brief Summary of contents discussed.

Set and Dictionary

Set: A set is an unordered collection of unique elements.

```
# Creating a set
fruits = {"apple", "banana", "cherry"}
print(fruits)  # Output is unordered and unique

# Adding elements
fruits.add("orange")
print(fruits)

# Removing elements
fruits.remove("banana")
print(fruits)

A = {1, 2, 3, 4}
B = {3, 4, 5, 6}

print(A | B)  # Union: {1, 2, 3, 4, 5, 6}
union_set = A.union(B)


print(A & B)  # Intersection: {3, 4}
intersection_set = A.intersection(B)


print(A - B)  # Difference: {1, 2}
difference_set = A.difference(B)


print(A ^ B)  # Symmetric Difference: {1, 2, 5, 6}
sym_diff_set = A.symmetric_difference(B)
```

Example: Students who attended only one workshop:

```
python_workshop = {"Alice", "Bob", "Charlie", "David"}
data_science_workshop = {"Charlie", "David", "Eve",
"Frank"}

# Students who attended only the Python workshop
only_python = python_workshop - data_science_workshop
print("Only Python Workshop:", only_python)  # Output:
{'Alice', 'Bob'}

# Students who attended only the Data Science workshop
only_data_science = data_science_workshop -
python_workshop
print("Only Data Science Workshop:",
only_data_science)  # Output: {'Eve', 'Frank'}
```

Dictionaries: A dictionary is a collection of key-value pairs, similar to a phone book where names(keys) are linked to numbers (values)

```
# Creating a dictionary
student = {"name": "John", "age": 20, "course":
"Math"}
print(student["name"])  # Output: John

# Adding a new key-value pair
student["grade"] = "A"

# Updating an existing value
student["age"] = 21

# Removing a key-value pair
del student["course"]

print(student)
```

#Looping through a dictionary

```
for key, value in student.items():
    print(key, ":", value)
```

Exercise:

1. Create a dictionary for a **bookstore** with book titles as keys and prices as values.
2. Update the price of a book.
3. Remove a book from the dictionary.
4. Print all books with their prices.

Applications of Sets and Dictionaries (20 minutes)

```
numbers = [1, 2, 2, 3, 4, 4, 5]
unique_numbers = set(numbers)
print(unique_numbers)  # Output: {1, 2, 3, 4, 5}
```

Word Frequency Counter Using Dictionary

```
text = "apple banana apple orange banana apple"
word_count = {}

for word in text.split():
    word_count[word] = word_count.get(word, 0) + 1

print(word_count)  # {'apple': 3, 'banana': 2,
'orange': 1}
```

Exercies

1. Create a dictionary storing **student names as keys** and **marks as values**.
2. Find the student with the **highest marks**.
3. Given a string, count the frequency of each character using a dictionary.
4. Given a sentence, count the frequency of each word. Ignore case and punctuation.

5. Merge Two Dictionaries and Sum Common Keys

```
dict1 = {'a': 5, 'b': 10, 'c': 15}
dict2 = {'b': 20, 'c': 25, 'd': 30}

Output: {'a': 5, 'b': 30, 'c': 40, 'd': 30}
```

Problem: Given two lists, one containing keys and another containing values, create a dictionary.

Problem: Given a dictionary, find the key corresponding to the highest value.

Problem: Given a nested dictionary of students and their subject scores, calculate their total scores.

```
students = {
    "Alice": {"math": 90, "science": 85,
"english": 88},
    "Bob": {"math": 75, "science": 80, "english":
78},
}

Output:
{'Alice': 263, 'Bob': 233}
```

Reverse a Dictionary (Keys as Values, Values as Keys)

Input: {'a': 1, 'b': 2, 'c': 3}
Output: {1: 'a', 2: 'b', 3: 'c'}