

## WEEK – 7

1. Write a python program to perform different set operations

### Union of Two Sets

#### Source code:-

```
# first set
A = {1, 3, 5}

# second set
B = {0, 2, 4}

# perform union operation using |
print('Union using |: ', A | B)

# perform union operation using union()
print('Union using union(): ', A.union(B))
```

#### Output:-

```
Union using |: {0, 1, 2, 3, 4, 5}
Union using union(): {0, 1, 2, 3, 4, 5}
```

#### Description:-

We use the `|` operator or the `union()` method to perform the set union operation.

### Set Intersection

```
# first set
A = {1, 3, 5}

# second set
B = {1, 2, 3}

# perform intersection operation using &
print('Intersection using &:', A & B)

# perform intersection operation using intersection()
print('Intersection using intersection(): ', A.intersection(B))
```

## Output

```
Intersection using &:amp; {1, 3}  
Intersection using intersection(): {1, 3}
```

### Description:-

In Python, we use the `&` operator or the `intersection()` method to perform the set intersection operation.

## Difference between Two Sets

```
# first set  
A = {2, 3, 5}  
  
# second set  
B = {1, 2, 6}  
  
# perform difference operation using &  
print('Difference using &:', A - B)  
  
# perform difference operation using difference()  
print('Difference using difference():', A.difference(B))
```

## Output

```
Difference using &: {3, 5}  
Difference using difference(): {3, 5}
```

### Description:-

We use the `-` operator or the `difference()` method to perform the difference between two sets.

## Set Symmetric Difference

```
# first set
A = {2, 3, 5}

# second set
B = {1, 2, 6}

# perform difference operation using &
print('using ^:', A ^ B)

# using symmetric_difference()
print('using symmetric_difference():', A.symmetric_difference(B))
```

Run Code >>

### Output

```
using ^: {1, 3, 5, 6}
using symmetric_difference(): {1, 3, 5, 6}
```

### Description:-

In Python, we use the `^` operator or the `symmetric_difference()` method to perform symmetric difference between two sets.

2. write a python script that prints prime numbers less than 20

### Source code:-

```
lower_value = int(input ("Please, Enter the Lowest Range Value: "))
upper_value = int(input ("Please, Enter the Upper Range Value: "))

print ("The Prime Numbers in the range are: ")
for number in range (lower_value, upper_value + 1):
    if number > 1:
        for i in range (2, number):
            if (number % i) == 0:
                break
        else:
            print (number)
```

### **Output:-**

```
Please, Enter the Lowest Range Value: 1
Please, Enter the Upper Range Value: 20
The Prime Numbers in the range are:
2
3
5
7
11
13
17
19
|
```

### **Description:-**

- Loop through all the elements in the given range.
- Check for each number if it has any factor between 1 and itself.
- If yes, then the number is not prime, and it will move to the next number.
- If no, it is the prime number, and the program will print it and check for the next number.
- The loop will break when it is reached to the upper value.

## **3. Write a Python program to Make a Simple Calculator**

### **Source code:-**

```
# This function adds two numbers
def add(x, y):
    return x + y

# This function subtracts two numbers
def subtract(x, y):
    return x - y

# This function multiplies two numbers
def multiply(x, y):
    return x * y

# This function divides two numbers
def divide(x, y):
    return x / y

print("Select operation.")
print("1.Add")
print("2.Subtract")
print("3.Multiply")
print("4.Divide")
```

```
while True:
    # take input from the user
    choice = input("Enter choice(1/2/3/4): ")

    # check if choice is one of the four options
    if choice in ('1', '2', '3', '4'):
        try:
            num1 = float(input("Enter first number: "))
            num2 = float(input("Enter second number: "))
        except ValueError:
            print("Invalid input. Please enter a number.")
            continue

        if choice == '1':
            print(num1, "+", num2, "=", add(num1, num2))

        elif choice == '2':
            print(num1, "-", num2, "=", subtract(num1, num2))

        elif choice == '3':
            print(num1, "*", num2, "=", multiply(num1, num2))

        elif choice == '4':
            print(num1, "/", num2, "=", divide(num1, num2))
```

```
next_calculation = input("Let's do next calculation? (yes/no): ")
if next_calculation == "no":
    break
else:
    print("Invalid Input")
```

### **Output:-**

```
Select operation.
1.Add
2.Subtract
3.Multiply
4.Divide
Enter choice(1/2/3/4): 3
Enter first number: 15
Enter second number: 14
15.0 * 14.0 = 210.0
Let's do next calculation? (yes/no): no
```

### **Description:-**

- In this program, we ask the user to choose an operation. Options 1, 2, 3, and 4 are valid. If any other input is given, `Invalid Input` is displayed and the loop continues until a valid option is selected.
- Two numbers are taken and an `if...elif...else` branching is used to execute a particular section. User-defined functions `add()`, `subtract()`, `multiply()` and `divide()` evaluate respective operations and display the output.