

1) Write a program to swap two values using tuple assignment.

```
1 x=int(input("ENTER FIRST NUMBER : "))
2 y=int(input("ENTER SECOND NUMBER : "))
3 print("\nBEFORE SWAPPING : X=",x," Y=",y)
4 (x,y)=(y,x)
5 print("\nAFTER SWAPPING : X=",x," Y=",y)
```

Console

```
BEFORE SWAPPING : X= 20 Y= 15
```

```
AFTER SWAPPING : X= 15 Y= 20
```

2. Write a program that scans an email address and forms a tuple of user name and domain.

main.py

```
1 Email=input("ENTER EMAIL ID : ")
2 Usernm=()
3 Domain=()
4 if '@' in Email:
5     Usernm=Email[0:Email.index('@')]
6     Domain=Email[Email.index('@')+1:]
7     print("UserName = ",Usernm," And Domain = ",Domain)
8 else:
9     print("Invalid Email")
10
11
```

OUTPUT

```
ENTER EMAIL ID : Pkr.bca@gmail.com
UserName = Pkr.bca And Domain = gmail.com
> |
```

3. write a program that has a list of numbers (both positive and negative). Make a new tuple that has only positive values from the list.

```
main.py
1 X=[1,9,5,-5,-13,39,105,0,-4,36,24]
2 tup1=()
3 for i in X:
4     if i>0:
5         tup1=tup1+(i,)
6 print("POSITIVE NUMBERS : ",tup1)
```

```
Shell
POSITIVE NUMBERS : (1, 9, 5, 39, 105, 36, 24)
> |
```

4. Write a program using nested tuple to store book details like “Title of book”, “Author name”, “Publisher name”, “Year of publication” and “Price”. Get the book title as input from user to search and display the details of the book.

```
Record=(( "C", "Kishor", "Arrow", 2020, 190),
        ("C++", "Ramesh", "Bing", 2021, 195),
        ("JAVA", "Mined", "Lotus", 2024, 200),
        ("PYTHON", "Suresh", "zix", 2021, 195))
n= input ("ENTER TITLE TO SEARCH : ")
for i in Record:
    if (i.count(n)):
        print(i)
        break
else:
    print("Record Not Found")
```

```
ENTER TITLE TO SEARCH : JAVA
('JAVA', 'Mined', 'Lotus', 2024, 200)

...Program finished with exit code 0
Press ENTER to exit console.
```

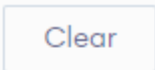
SET

- 1) Write a program that generates a set of prime numbers and another set of odd numbers. Demonstrate the result of union, intersection, difference and symmetric difference operations on these sets.

```
main.py   

1 Odd=set()
2 Prime=set()
3 N=int(input("RANGE FOR ODD NUMBER : "))
4 S=int(input("RANGE FOR PRIME NUMBER : "))
5 for i in range(1,N+1):
6     if i%2==1:
7         Odd.add(i)
8 for i in range(2,S+1):
9     for j in range(2,int(i/2)+1):
10        if(i%j==0):
11            break
12    else:
13        Prime.add(i)
14 print("Odd : ",Odd)
15 print("Prime : ",Prime)
16 print("UNION = ",Odd.union(Prime))
17 print("Intersection = ",Prime.intersection(Odd))
18 print("Difference = ",Odd.symmetric_difference(Prime))
19 print("UNION = ",Prime.difference(Odd))
20
```

OUTPUT

```
Shell 

RANGE FOR ODD NUMBER : 25
RANGE FOR PRIME NUMBER : 40
Odd :  {1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25}
Prime :  {2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31}
UNION =  {1, 2, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 29, 31, 37}
Intersection =  {3, 5, 7, 11, 13, 17, 19, 23}
Difference =  {1, 2, 37, 9, 15, 21, 25, 29, 31}
UNION =  {2, 29, 37, 31}
> |
```

2) Write a program that creates two sets. One of even numbers in range 1-10 and the other has all composite numbers in range 1-20. Demonstrate the use of all(), issuperset(), len(), and sum() functions on the sets.

main.py



Run

```
1 Even=set()
2 Composite=set()
3 for i in range(1,11):
4     if i%2==0:
5         Even.add(i)
6 for i in range(1,21):
7     for j in range(2,int(i/2)+1):
8         if(i%j==0):
9             Composite.add(i)
10 print("EVEN NUMBER : ",Even)
11 print("COMPOSITE : ",Composite)
12 print("ISUPPERSET = ",Composite.issuperset(Even))
13 print("LENGTH (Even) = ",len(Even)," <=> LENGTH (Composite) = ",len
    (Composite))
14 print("SUM (Even) = ",sum(Even)," <=> SUM (Composite) = ",sum
    (Composite))
15 print("ALL (Even) = ",all(Even)," <=> ALL (Composite) = ",all
    (Composite))
```

OUTPUT

Shell

```
EVEN NUMBER : {2, 4, 6, 8, 10}
COMPOSITE : {4, 6, 8, 9, 10, 12, 14, 15, 16, 18, 20}
ISUPPERSET = False
LENGTH (Even) = 5 <=> LENGTH (Composite) = 11
SUM (Even) = 30 <=> SUM (Composite) = 132
ALL (Even) = True <=> ALL (Composite) = True
> |
```

NAME – PURUSHOTTAM KUMAR (2041)