**EXP 5:**

#1)Create empty set ‘Engineers’ and ‘Managers’.

engineers=set()

managers=set()

#2)Using input method add elements in ‘Engineers’ and ‘Managers’:

engineers.add("Jane")

engineers.add( "John")

engineers.add("Janice")

engineers.add( "Jack")

managers.add("Jane")

managers.add("Jack")

managers.add("Susan")

managers.add("Zack")

print("engineers are ",engineers)

print("managers are " ,managers)

print("\n")

#3)Display all engineers in this format: “Name of Engineer is ---“Jane

for x in engineers:

    print(f"Name of Engineer is {x}")

    print("\n")

#4)Copy all managers and construct a tuple myManagers = ('Jane', 'Jack', 'Susan', 'Zack')

mymanager=tuple(managers)

print("tuple of my manager is ", mymanager)

print("\n")

#5)Copy all engineers and construct a list myEngineers = {'Jane', 'John', 'Janice', 'Jack'}

mylist=list[engineers]

print("list of engineers", engineers)

print("\n")

#6)Add new manager ‘Jenifer’

managers.add("jenifer")

print("new managers is added",managers)

print("\n")

#7)Create a third set Engineer\_Manager by merging both Engineers and Managers sets.

engineer\_manager=engineers.union(managers)

print(engineer\_manager)

print("\n")

#8)Display the name of engineers who are not managers

notmanagers=engineers.difference(managers)

print("the name of engineers who are not managers ",notmanagers)

print("\n")

#9)Display the name of engineers who also serving as managers.

alsomanager=engineers.intersection(managers)

print("the name of engineers who also serving as managers.",alsomanager)

print("\n")

#10)Display the name of person who is either engineer or manager only but not performing both jobs

notabothjob= engineers.symmetric\_difference(managers)

print("the name of person who is either engineer or manager only but not performing both jobs",notabothjob)

print("\n")

**Output :**

engineers are {'Jane', 'Janice', 'John', 'Jack'}

managers are {'Jane', 'Jack', 'Susan', 'Zack'}

Name of Engineer is Jane

Name of Engineer is Janice

Name of Engineer is John

Name of Engineer is Jack

tuple of my manager is ('Jane', 'Jack', 'Susan', 'Zack')

list of engineers {'Jane', 'Janice', 'John', 'Jack'}

new managers is added {'Jane', 'Jack', 'jenifer', 'Zack', 'Susan'}

{'Jane', 'Jack', 'John', 'jenifer', 'Zack', 'Susan', 'Janice'}

the name of engineers who are not managers {'Janice', 'John'}

the name of engineers who also serving as managers. {'Jane', 'Jack'}

the name of person who is either engineer or manager only but not performing both jobs {'Zack', 'Susan', 'Janice', 'John', 'jenifer'}