# PRACTICAL – 1

**Aim :- Write a program to demonstrate various string method.**

**Code:**

name = "Rishi joshi"

length = len(name)

print(length)

num = 11

temp = str(num)

print(type(temp))

name = "rishi joshi"

name = name.upper()

print(name)

name = "RISHI JOSHI"

name = name.lower()

print(name)

name = "rISHI jOSHI"

name = name.capitalize()

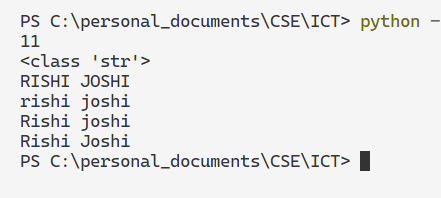
print(name)

name = "rishi joshi"

name = name.title()

print(name)

**Output:**

****

# PRACTICAL – 2

**Aim :- Write a program to demonstrate accessing method of string characters.**

**Code:**

name = "Rishi Joshi"

# Accessing String through the index

first\_char = name[0]

second\_char = name[1]

last\_char = name[10]

secLast\_char = name[9]

print(first\_char)

print(second\_char)

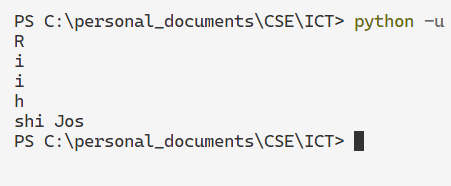
print(last\_char)

print(secLast\_char)

# Using slicing method

substring = name[2:9]

print(substring)

**Output:**

# PRACTICAL – 3

**Aim :- Write a program that create a list and perform various list method.**

**Code:**

cars = ['Lancer',600,'rx7',800]

cars.append('Eclipse')

print(cars)

length = cars.remove(600)

print(cars)

cars.insert(2,"skyline")

print(cars)

number = [5, 2, 1, 4, 3]

number.sort()

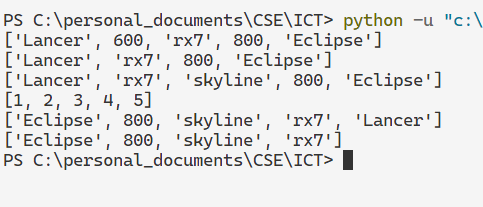
print(number)

cars.reverse()

print(cars)

cars.pop()

print(cars)

**Output:**

# PRACTICAL – 4

**Aim :- Write a program to demonstrate the method of accessing list elements.**

**Code:**

# Accessing through the index

languages = ["python", "c", "cpp", "java"]

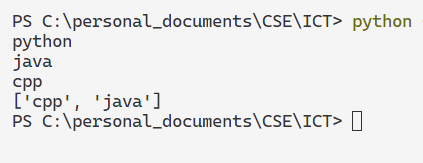
print(languages[0])

print(languages[-1])

print(languages[-2])

# Accessing through the slicing

print(languages[2:4])

**Output:**

# PRACTICAL – 5

**Aim :- Write a program to demonstrate the method of tuple elements.**

**Code:**

nums = (5, 3, 2, 4, 1, 2)

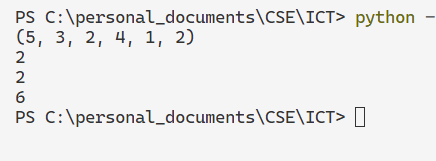
print(nums)

print(nums.count(2))

print(nums.index(2))

print(len(nums))

**Output:**



# PRACTICAL – 6

**Aim :- Write a program to create a dictionary and demonstrate various dictionary methods.**

**Code:**

people = {"name" : "Rishi", "rollNo" : 551}

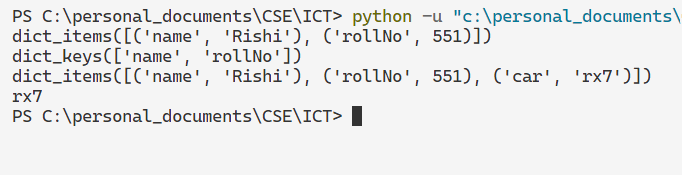
print(people.items())

print( people.keys())

people.update({"car" : "rx7"})

print(people.items())

print(people.get("car"))

**Output:**

# PRACTICAL – 7

**Aim :- Write a program to demonstrate various set methods.**

**Code:**

# method 1 to create set

dogs = {"pitbull", "husky"}

# method 2 to create set

cats = set({"tiger", "lion"})

dogs.add("golden")

print(dogs)

print(len(dogs))

print(dogs.pop())

newSet = dogs.union(dogs,cats)

print(newSet)

dogs.remove("golden")

print(dogs)

dogs.clear()

print(dogs)

**Output:**

# PRACTICAL – 8

**Aim :- Demonstrate difference between set and frozen set.**

**Code:**

dogs = {"pitbull", "husky", "golden"}

choosenDogs = frozenset(dogs)

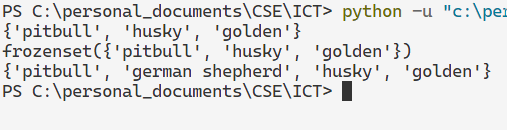
print(dogs)

print(choosenDogs)

dogs.add("german shepherd")

print(dogs)

# choosenDogs.add("german shepherd") it wil give you error

**Output:**

# PRACTICAL – 9

**Aim :- Give the difference between 1) list and tuple 2) set and dictionary.**

**Code:**

**Difference between List and Tuple**

# List

my\_list = [1, 2, 3]

my\_list[0] = 4 # Lists are mutable, you can change elements

print(my\_list) # Output: [4, 2, 3]

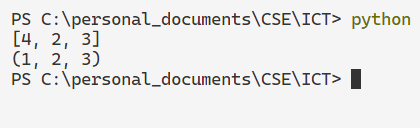
# Tuple

my\_tuple = (1, 2, 3)

# my\_tuple[0] = 4 # This would raise an error, tuples are immutable

print(my\_tuple)

**Output:**



**Difference between Set and Dictionary**

# Set

my\_set = {1, 2, 3}

my\_set.add(4) # Sets are mutable, you can add and remove elements

my\_set.remove(2)

print(my\_set) # Output: {1, 3, 4}

# Dictionary

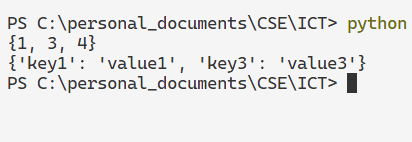
my\_dict = {'key1': 'value1', 'key2': 'value2'}

my\_dict['key3'] = 'value3' # Dictionaries are mutable, you can add, remove, and modify key-value pairs

del my\_dict['key2']

print(my\_dict) # Output: {'key1': 'value1', 'key3': 'value3'}

**Output:**

****