

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:

```
PS C:\personal_documents\CSE\ICT> python -
11
<class 'str'>
RISHI JOSHI
rishi joshi
Rishi joshi
Rishi Joshi
PS C:\personal_documents\CSE\ICT> █
```

PRACTICAL – 2

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

Code:

```
name = "Rishi Joshi"
# Accessing String through the index

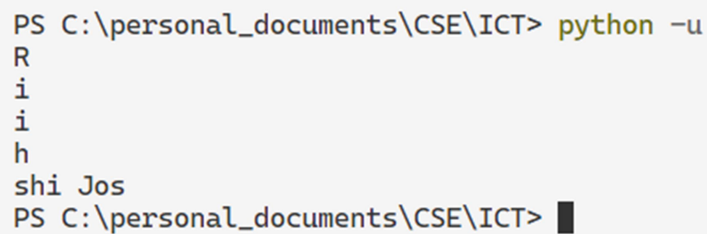
first_char = name[0]
second_char = name[1]
last_char = name[10]
secLast_char = name[9]

print(first_char)
print(second_char)
print(last_char)
print(secLast_char)

# Using slicing method

substring = name[2:9]
print(substring)
```

Output:



```
PS C:\personal_documents\CSE\ICT> python -u
R
i
i
h
shi Jos
PS C:\personal_documents\CSE\ICT> █
```

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)
```

Signature of Faculty: _____

Date: _____

Output:

```
PS C:\personal_documents\CSE\ICT> python -u "c:\
['Lancer', 600, 'rx7', 800, 'Eclipse']
['Lancer', 'rx7', 800, 'Eclipse']
['Lancer', 'rx7', 'skyline', 800, 'Eclipse']
[1, 2, 3, 4, 5]
['Eclipse', 800, 'skyline', 'rx7', 'Lancer']
['Eclipse', 800, 'skyline', 'rx7']
PS C:\personal_documents\CSE\ICT> █
```

PRACTICAL – 4

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

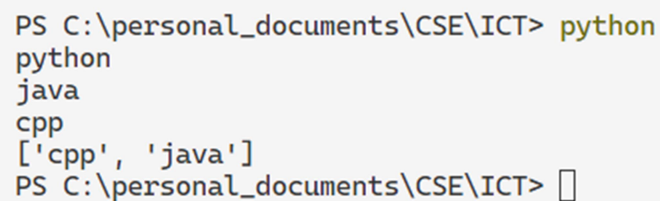
Code:

```
# 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:



```
PS C:\personal_documents\CSE\ICT> python ...
python
java
cpp
['cpp', 'java']
PS C:\personal_documents\CSE\ICT> █
```

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:

```
PS C:\personal_documents\CSE\ICT> python -
(5, 3, 2, 4, 1, 2)
2
2
6
PS C:\personal_documents\CSE\ICT> □
```

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:

```
PS C:\personal_documents\CSE\ICT> python -u "c:\personal_documents\  
dict_items([('name', 'Rishi'), ('rollNo', 551)])  
dict_keys(['name', 'rollNo'])  
dict_items([('name', 'Rishi'), ('rollNo', 551), ('car', 'rx7')])  
rx7  
PS C:\personal_documents\CSE\ICT> █
```

Signature of Faculty: _____

Date: _____

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:

```
PS C:\personal_documents\CSE\ICT> python -u
{'husky', 'pitbull', 'golden'}
3
husky
{'pitbull', 'lion', 'tiger', 'golden'}
{'pitbull'}
set()
PS C:\personal_documents\CSE\ICT> █
```

Signature of Faculty: _____

Date: _____

PRACTICAL – 8

Aim :- Demonstrate difference between set and frozen set.

Code:

```
dogs = {"pitbull", "husky", "golden"}
chosenDogs = frozenset(dogs)

print(dogs)
print(chosenDogs)

dogs.add("german shepherd")
print(dogs)
# chosenDogs.add("german shepherd") it wil give you error
```

Output:

```
PS C:\personal_documents\CSE\ICT> python -u "c:\per
{'pitbull', 'husky', 'golden'}
frozenset({'pitbull', 'husky', 'golden'})
{'pitbull', 'german shepherd', 'husky', 'golden'}
PS C:\personal_documents\CSE\ICT> █
```

Signature of Faculty: _____

Date: _____

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:

```
PS C:\personal_documents\CSE\ICT> python
[4, 2, 3]
(1, 2, 3)
PS C:\personal_documents\CSE\ICT> █
```

Signature of Faculty: _____

Date: _____

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:

```
PS C:\personal_documents\CSE\ICT> python
{1, 3, 4}
{'key1': 'value1', 'key3': 'value3'}
PS C:\personal_documents\CSE\ICT> █
```