Four collection data types

- **List** is a collection which is ordered and changeable. Allows duplicate members.
- **Tuple** is a collection which is ordered and unchangeable. Allows duplicate members.
- **Set** is a collection which is unordered, unchangeable, and unindexed. No duplicate members.
- **Dictionary** is a collection which is ordered and changeable. No duplicate members.

List, Tuple, Set, Dictionary

Program

```
myList = [10, 20, 30]
print (myList)
print (len(myList))

myTuple = (11, 21, 31)
print (myTuple)
print (len(myTuple))

mySet = {40, 50, 60}
print (mySet)
print (len(mySet))

myDictionary = {"name":"Hiren", "age":47}
print (myDictionary)
```

Output

```
[10, 20, 30]
3
(11, 21, 31)
3
{40, 50, 60}
3
{'name': 'Hiren', 'age': 47}
```

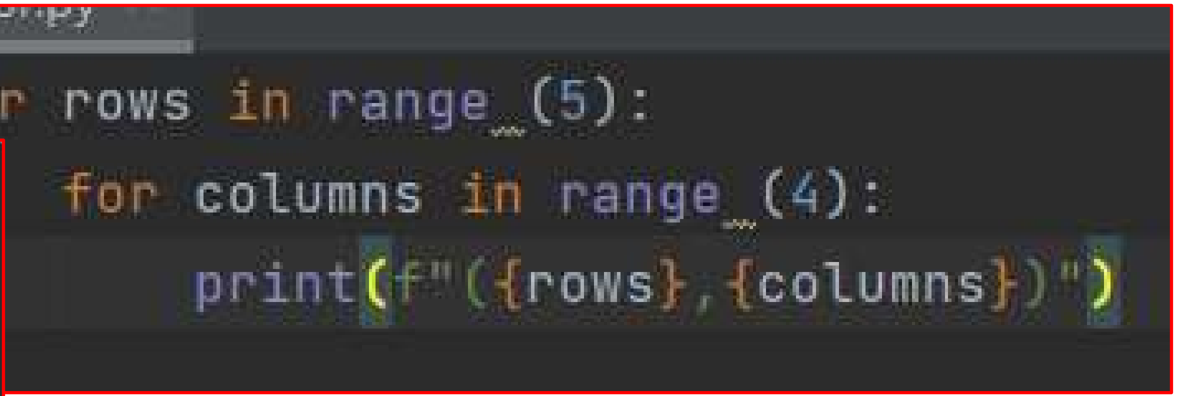
List

```
items = [10, 22, 34, 49, 56, 66]
print(items)
items.append(71)
print(items)
items.insert(2, 25)
print(items)
items.remove(49)
print(items)
items.pop()
print(items)
print(items.index(56))
print(56 in items)
print(items.count(22))
items.sort()
print(items)
items.reverse()
print(items)
dup = items.copy()
dup.append(99);
print(dup)
items.clear()
print(items)
```

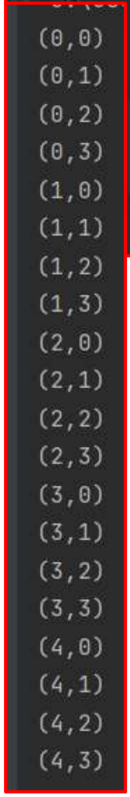
```
[10, 22, 34, 49, 56, 66]

[10, 22, 34, 49, 56, 66, 71]

[10, 22, 25, 34, 49, 56, 66, 71]

[10, 22, 25, 34, 56, 66, 71]

[10, 22, 25, 34, 56, 66]
4
True
1

[10, 22, 25, 34, 56, 66]

[66, 56, 34, 25, 22, 10]


[66, 56, 34, 25, 22, 10, 99]

[]
```

List

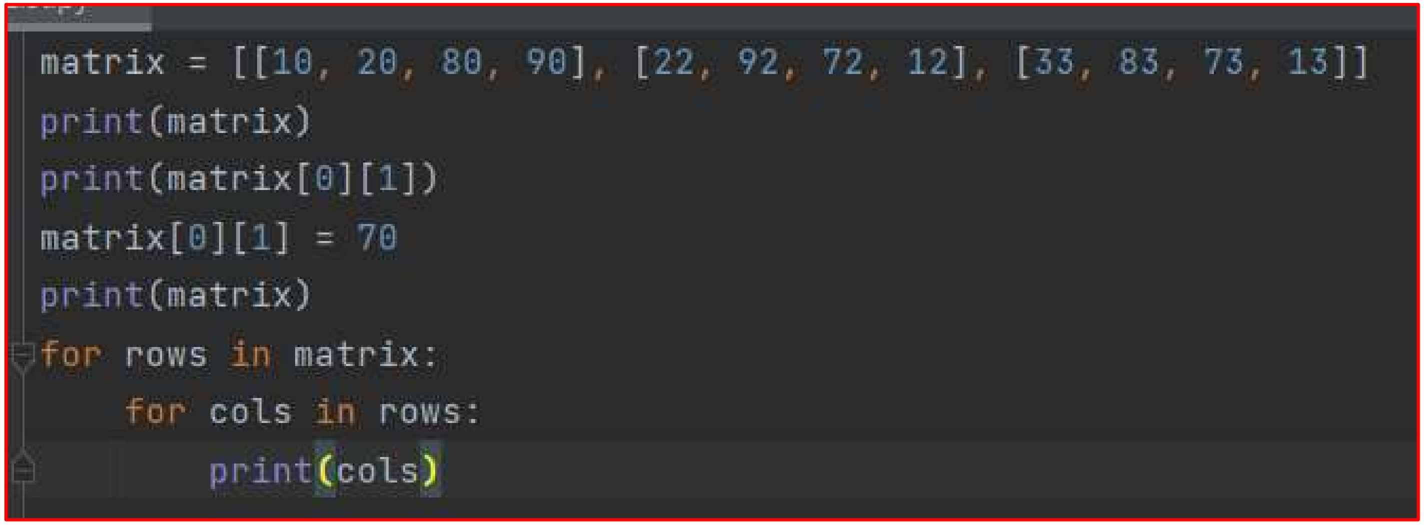
www.hbpatel.in



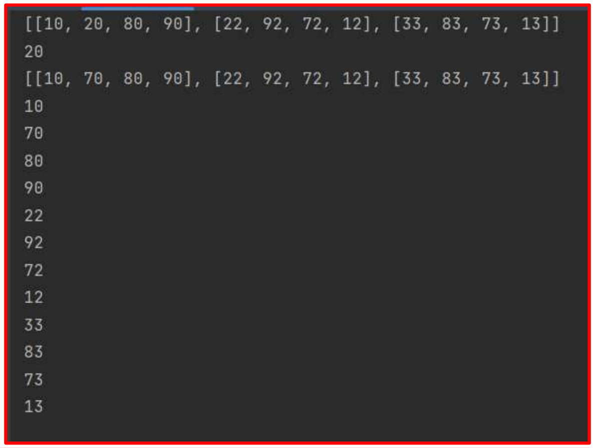
```
for rows in range (5):
    for columns in range (4):
        print(f"({rows},{columns})")
```



```
(0,0)
(0,1)
(0,2)
(0,3)
(1,0)
(1,1)
(1,2)
(1,3)
(2,0)
(2,1)
(2,2)
(2,3)
(3,0)
(3,1)
(3,2)
(3,3)
(4,0)
(4,1)
(4,2)
(4,3)
```



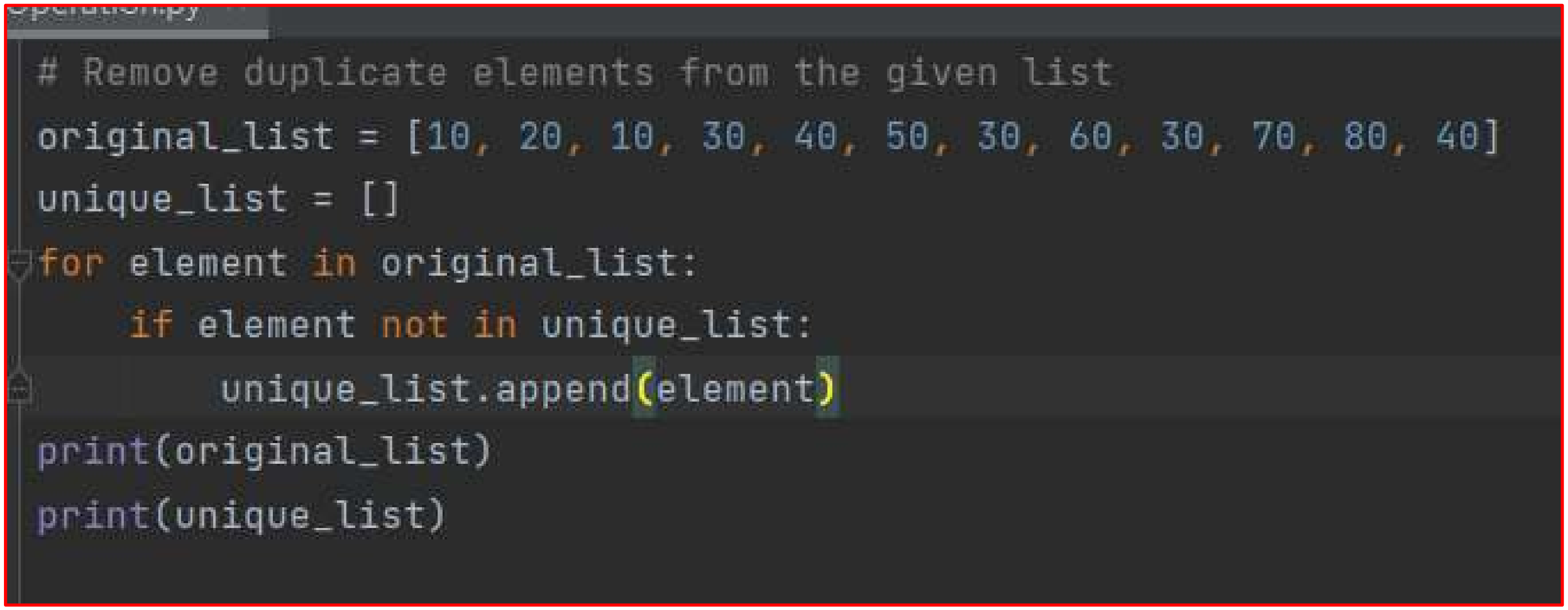
```
matrix = [[10, 20, 80, 90], [22, 92, 72, 12], [33, 83, 73, 13]]
print(matrix)
print(matrix[0][1])
matrix[0][1] = 70
print(matrix)
for rows in matrix:
    for cols in rows:
        print(cols)
```



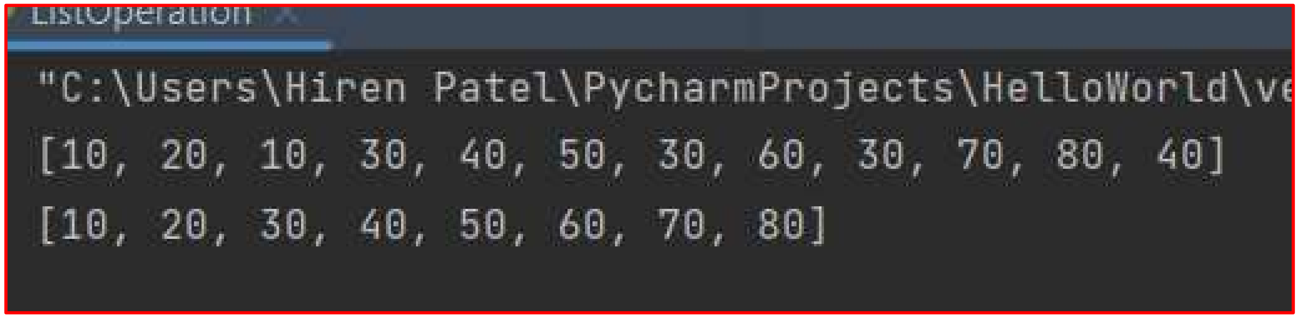
```
[[10, 20, 80, 90], [22, 92, 72, 12], [33, 83, 73, 13]]
20
[[10, 70, 80, 90], [22, 92, 72, 12], [33, 83, 73, 13]]
10
70
80
90
22
92
72
12
33
83
73
13
```

List

www.hbpatel.in



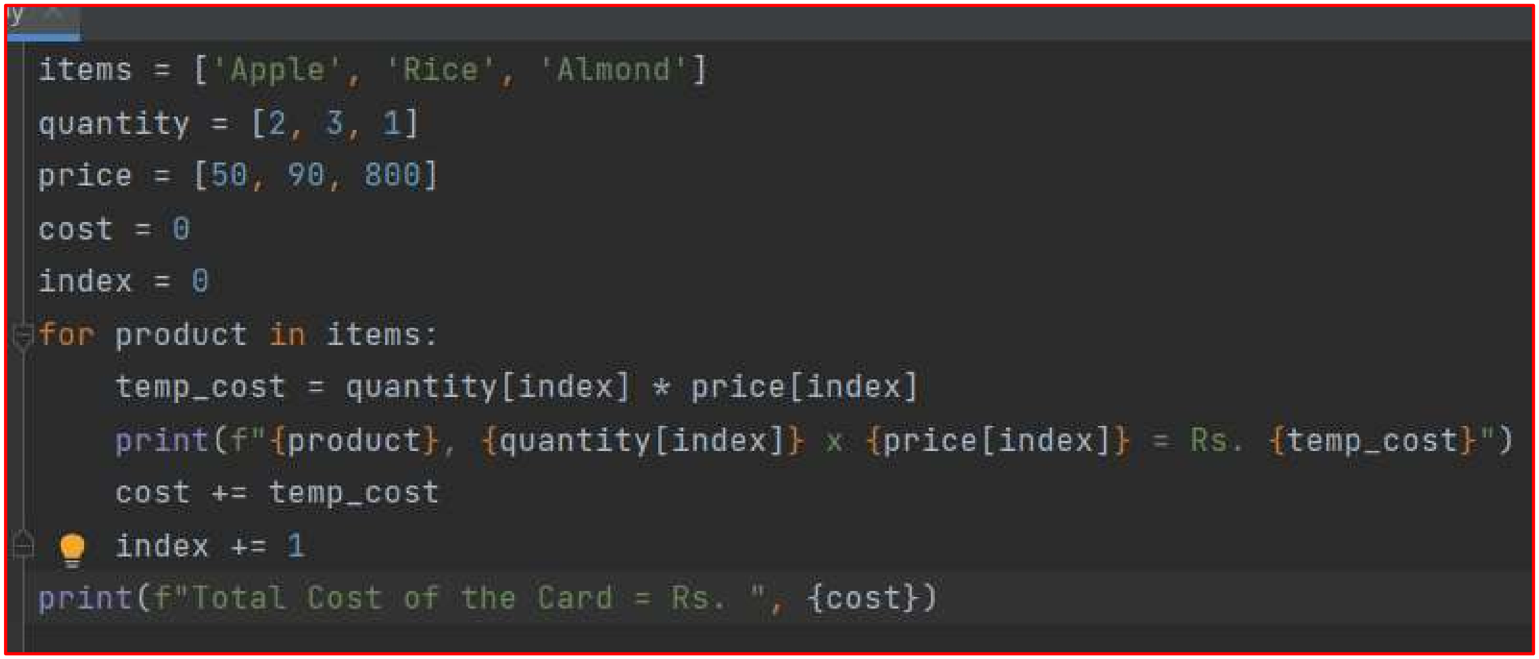
```python
# Remove duplicate elements from the given list
original_list = [10, 20, 10, 30, 40, 50, 30, 60, 30, 70, 80, 40]
unique_list = []
for element in original_list:
    if element not in unique_list:
        unique_list.append(element)
print(original_list)
print(unique_list)
```



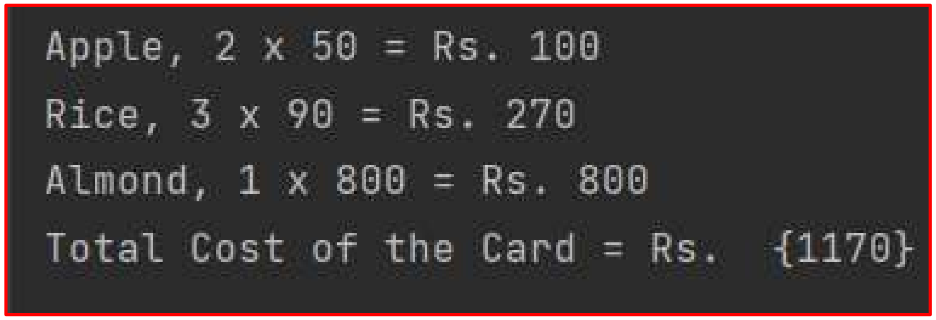
```
"C:\Users\Hiren Patel\PycharmProjects\HelloWorld\ve
[10, 20, 10, 30, 40, 50, 30, 60, 30, 70, 80, 40]
[10, 20, 30, 40, 50, 60, 70, 80]
```

List

www.hbpatel.in



```
items = ['Apple', 'Rice', 'Almond']
quantity = [2, 3, 1]
price = [50, 90, 800]
cost = 0
index = 0
for product in items:
    temp_cost = quantity[index] * price[index]
    print(f"{product}, {quantity[index]} x {price[index]} = Rs. {temp_cost}")
    cost += temp_cost
    index += 1
print(f"Total Cost of the Card = Rs. ", {cost})
```



```
Apple, 2 x 50 = Rs. 100
Rice, 3 x 90 = Rs. 270
Almond, 1 x 800 = Rs. 800
Total Cost of the Card = Rs.  {1170}
```

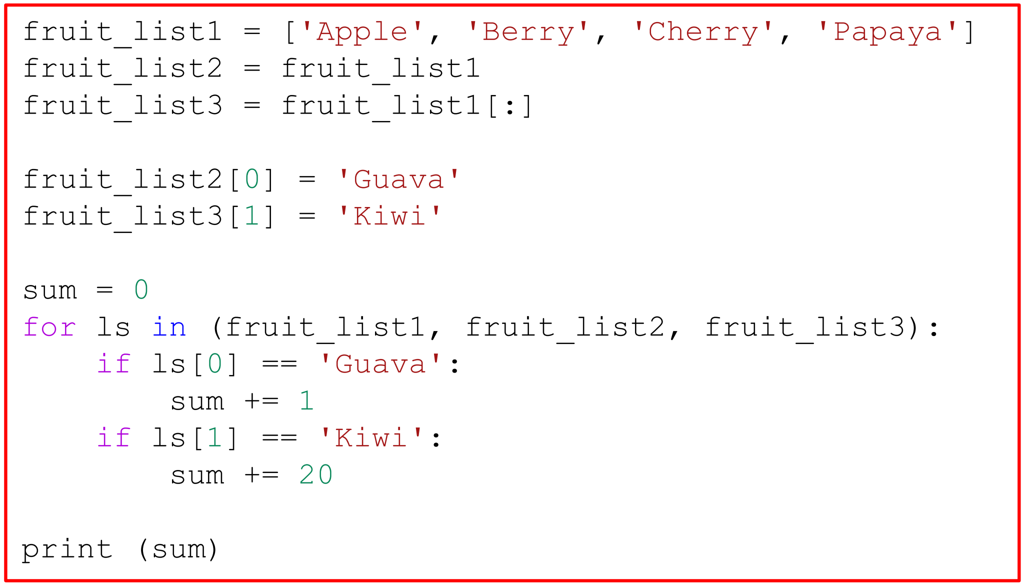
List (Predict the output)

```
List1 = ['Hiren','Sanjay','Vijay','Pradip']
List2 = List1
print (List1)
print (List2)

List2[0] = 'Parimal'
print (List1)
print (List2)

List1[1] = 'Pravin'
print (List1)
print (List2)
```

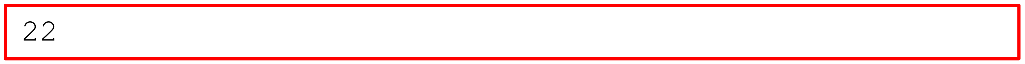
```
['Hiren', 'Sanjay', 'Vijay', 'Pradip']
['Hiren', 'Sanjay', 'Vijay', 'Pradip']
['Parimal', 'Sanjay', 'Vijay', 'Pradip']
['Parimal', 'Sanjay', 'Vijay', 'Pradip']
['Parimal', 'Pravin', 'Vijay', 'Pradip']
['Parimal', 'Pravin', 'Vijay', 'Pradip']
```

List (Predict the output)

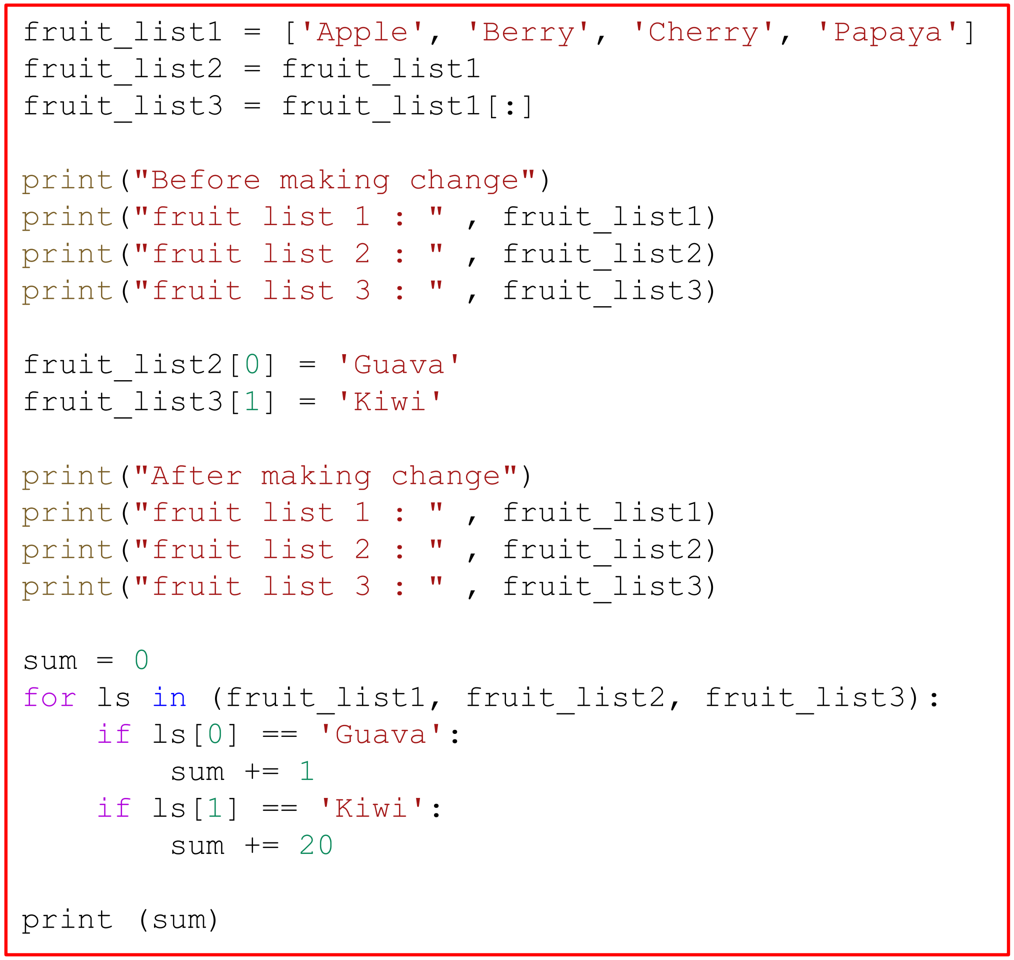
www.hbpatel.in



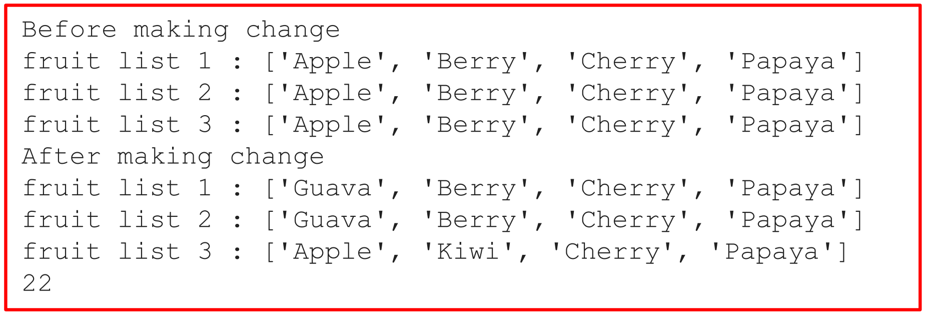
```python
fruit_list1 = ['Apple', 'Berry', 'Cherry', 'Papaya']
fruit_list2 = fruit_list1
fruit_list3 = fruit_list1[:]

fruit_list2[0] = 'Guava'
fruit_list3[1] = 'Kiwi'

sum = 0
for ls in (fruit_list1, fruit_list2, fruit_list3):
    if ls[0] == 'Guava':
        sum += 1
    if ls[1] == 'Kiwi':
        sum += 20

print (sum)
```



```
22
```

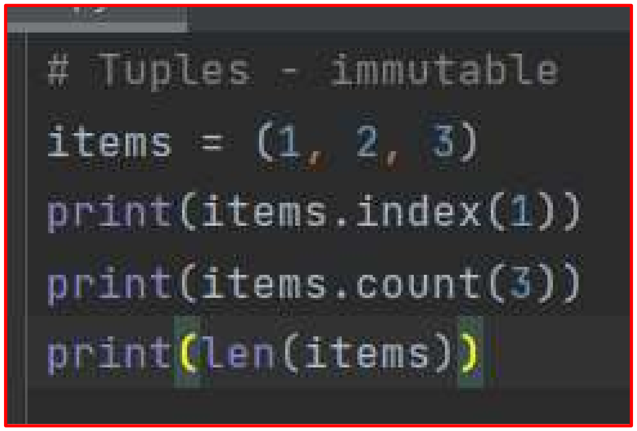


```python
fruit_list1 = ['Apple', 'Berry', 'Cherry', 'Papaya']
fruit_list2 = fruit_list1
fruit_list3 = fruit_list1[:]

print("Before making change")
print("fruit list 1 : " , fruit_list1)
print("fruit list 2 : " , fruit_list2)
print("fruit list 3 : " , fruit_list3)

fruit_list2[0] = 'Guava'
fruit_list3[1] = 'Kiwi'

print("After making change")
print("fruit list 1 : " , fruit_list1)
print("fruit list 2 : " , fruit_list2)
print("fruit list 3 : " , fruit_list3)

sum = 0
for ls in (fruit_list1, fruit_list2, fruit_list3):
    if ls[0] == 'Guava':
        sum += 1
    if ls[1] == 'Kiwi':
        sum += 20

print (sum)
```

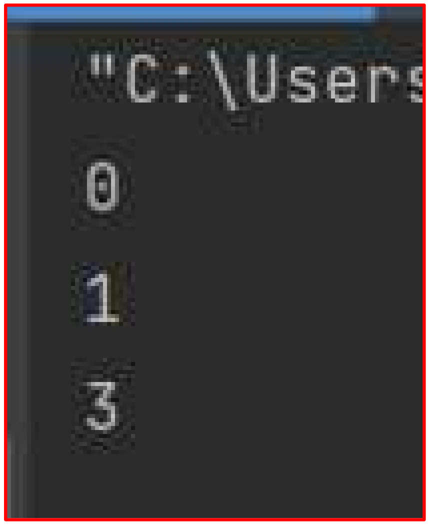


```
Before making change
fruit list 1 : ['Apple', 'Berry', 'Cherry', 'Papaya']
fruit list 2 : ['Apple', 'Berry', 'Cherry', 'Papaya']
fruit list 3 : ['Apple', 'Berry', 'Cherry', 'Papaya']
After making change
fruit list 1 : ['Guava', 'Berry', 'Cherry', 'Papaya']
fruit list 2 : ['Guava', 'Berry', 'Cherry', 'Papaya']
fruit list 3 : ['Apple', 'Kiwi', 'Cherry', 'Papaya']
22
```

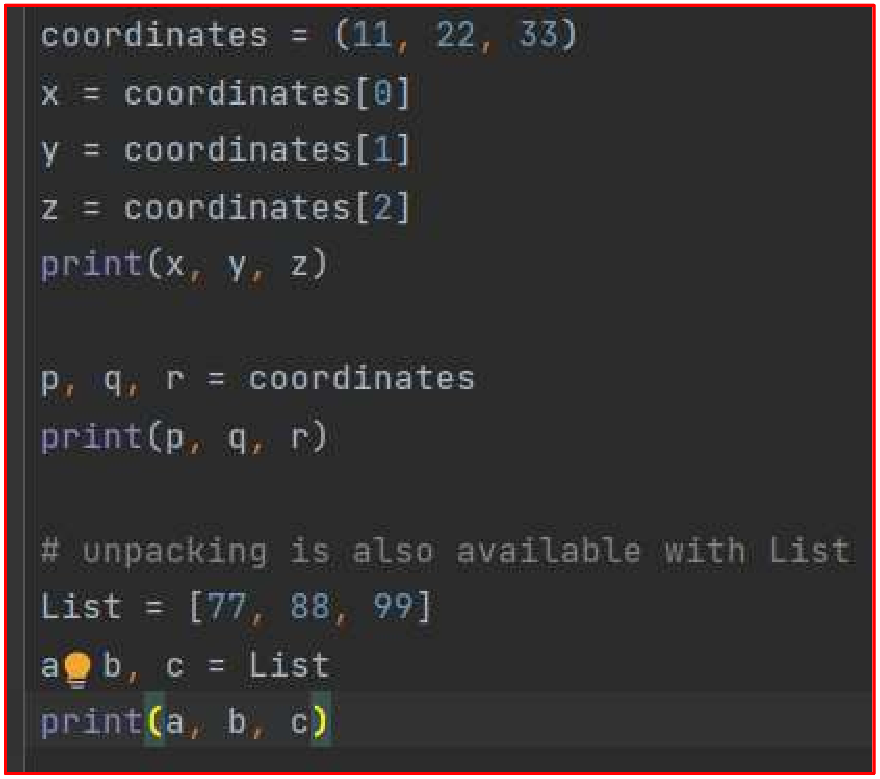
Tuples



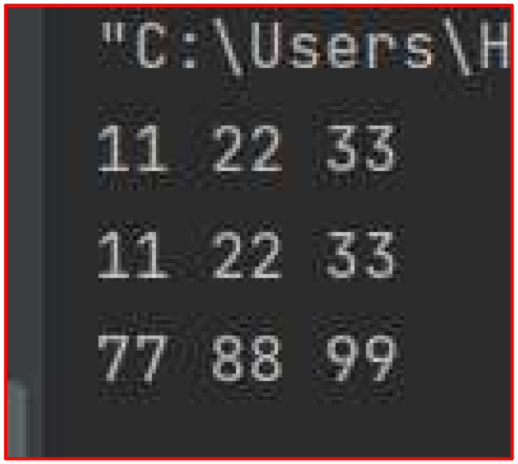
```python
# Tuples - immutable
items = (1, 2, 3)
print(items.index(1))
print(items.count(3))
print(len(items))
```



```
"C:\Users
0
1
3
```



```python
coordinates = (11, 22, 33)
x = coordinates[0]
y = coordinates[1]
z = coordinates[2]
print(x, y, z)

p, q, r = coordinates
print(p, q, r)

# unpacking is also available with List
List = [77, 88, 99]
a, b, c = List
print(a, b, c)
```



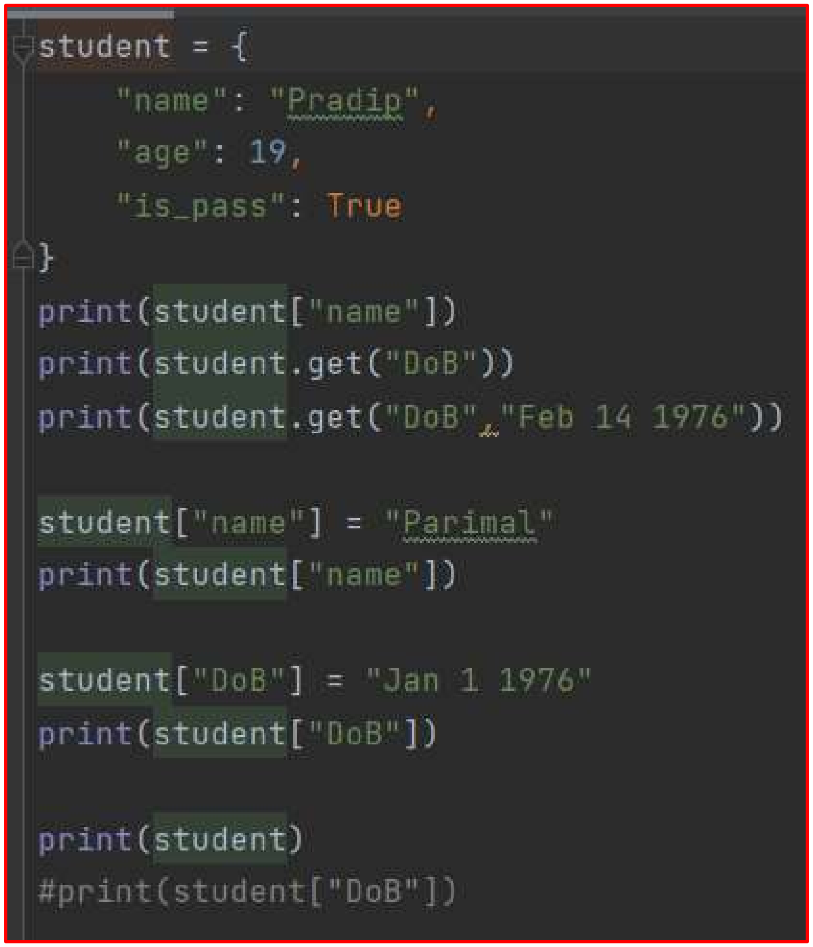
```
"C:\Users\H
11 22 33
11 22 33
77 88 99
```

Tuples (Predict the output)

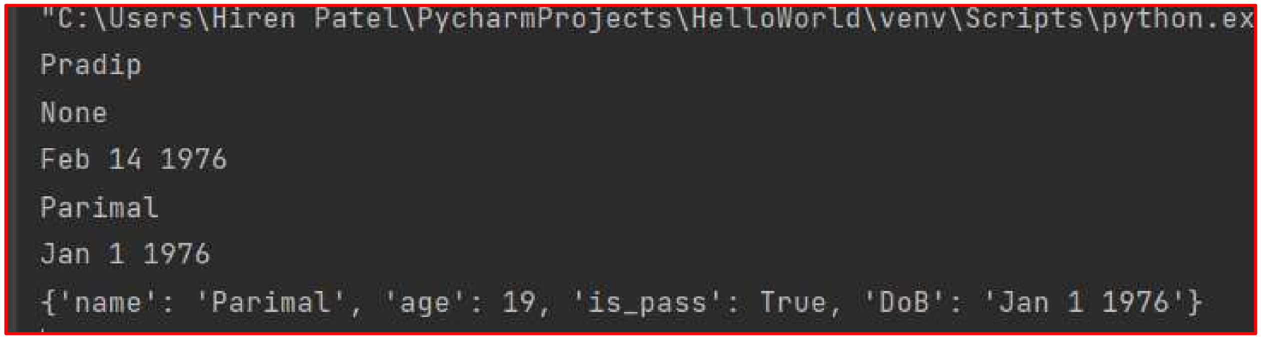
```
init_tuple_a = '1', '2'
init_tuple_b = ('3', '4')

print (init_tuple_a + init_tuple_b)

init_tuple_c = init_tuple_a + init_tuple_b
print (init_tuple_c)
```

```
('1', '2', '3', '4')
('1', '2', '3', '4')
```

Dictionaries



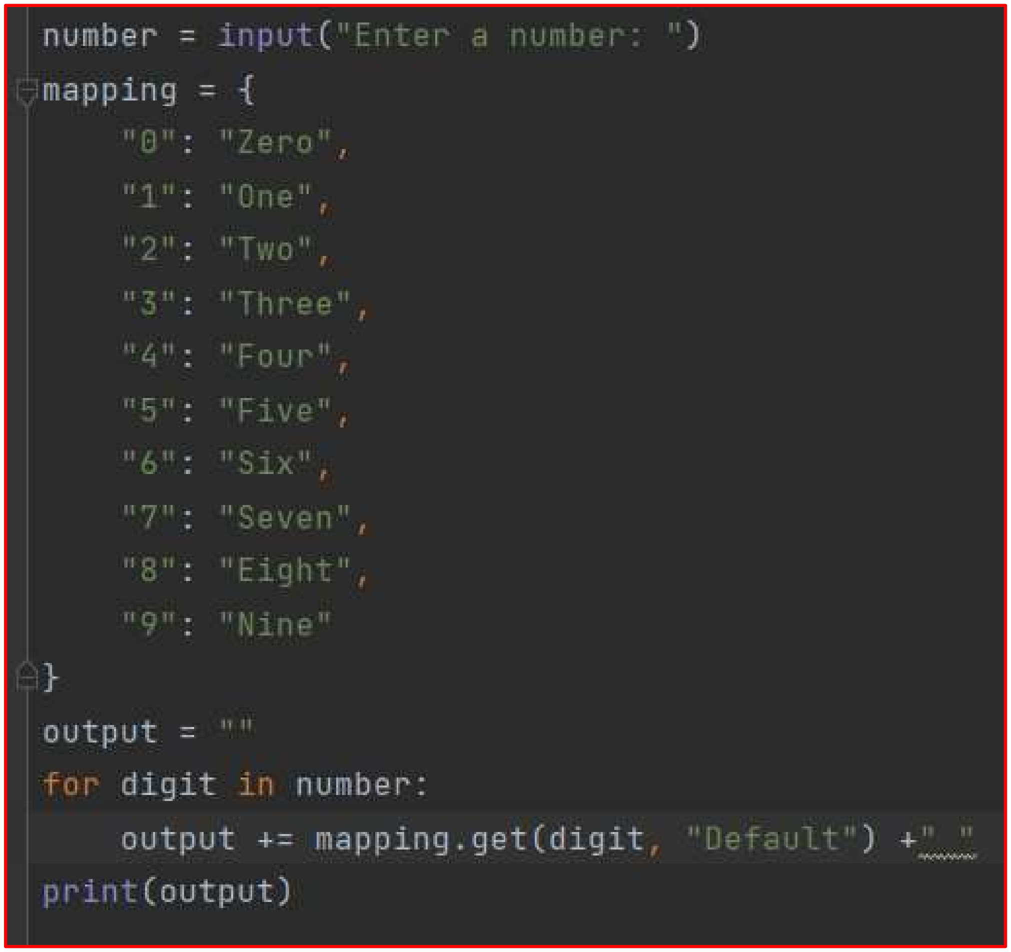
```
student = {
    "name": "Pradip",
    "age": 19,
    "is_pass": True
}
print(student["name"])
print(student.get("DoB"))
print(student.get("DoB","Feb 14 1976"))

student["name"] = "Parimal"
print(student["name"])

student["DoB"] = "Jan 1 1976"
print(student["DoB"])

print(student)
#print(student["DoB"])
```



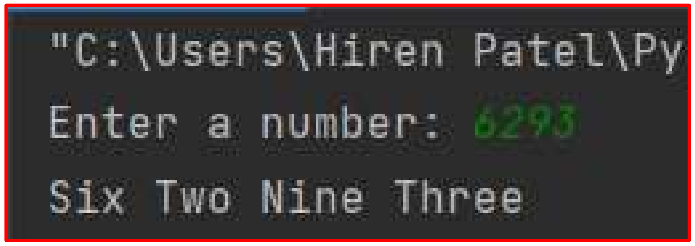
```
"C:\Users\Hiren Patel\PycharmProjects\HelloWorld\venv\Scripts\python.ex
Pradip
None
Feb 14 1976
Parimal
Jan 1 1976
{'name': 'Parimal', 'age': 19, 'is_pass': True, 'DoB': 'Jan 1 1976'}
```

Dictionaries

www.hbpatel.in



```python
number = input("Enter a number: ")
mapping = {
    "0": "Zero",
    "1": "One",
    "2": "Two",
    "3": "Three",
    "4": "Four",
    "5": "Five",
    "6": "Six",
    "7": "Seven",
    "8": "Eight",
    "9": "Nine"
}
output = ""
for digit in number:
    output += mapping.get(digit, "Default") +" "
print(output)
```



```
"C:\Users\Hiren Patel\Py
Enter a number: 6293
Six Two Nine Three
```

Dictionaries

```
student_marks = {"Hiren": 80, "Pradip": 90, "Parimal": 75, "Sanjay": 65, "Vijay": 50}
student_grades = {}

for student in student_marks:
  if(student_marks[student]>90):
    student_grades[student]="Outstanding"
  elif(student_marks[student]>80):
    student_grades[student]="Exceeds Expectations"
  elif(student_marks[student]>70):
    student_grades[student]="Acceptable"
  else:
    student_grades[student]="Fail"
  print(student, student_marks[student], student_grades[student])

print(student_marks)
print(student_grades)
```

```
Hiren 80 Acceptable
Pradip 90 Exceeds Expectations
Parimal 75 Acceptable
Sanjay 65 Fail
Vijay 50 Fail
{'Hiren': 80, 'Pradip': 90, 'Parimal': 75, 'Sanjay': 65, 'Vijay': 50}
{'Hiren': 'Acceptable', 'Pradip': 'Exceeds Expectations', 'Parimal': 'Acceptable', 'Sanjay': 'Fail', 'Vijay': 'Fail'}
```

Dictionaries

```
States = [
        {"name": "Uttar Pradesh",
        "cities": ["Agra", "Aligarh", "Allahabad"],
        "population": 199812341
        },
        {"name": "Maharashtra",
        "cities": ["Mumbai", "Pune", "Solapur"],
        "population": 112374333
        },
        {"name": "Gujarat",
        "cities": ["Ahmedabad", "Baroda", "Rajkot"],
        "population": 60439692
        },
]
def add_new_state(state_name, state_cities, state_population):
        new_state = {}
        new_state["name"] = state_name
        new_state["city"] = state_cities
        new_state["population"] = state_population
        States.append(new_state)

add_new_state("Rajasthan",["Udaipur","Jesalmer","Kumbhalgarh"], 68548437)
print(States)
print(States[2])
```

Dictionaries

www.hbpatel.in

OUTPUT

```
[
{'name': 'Uttar Pradesh', 'cities': ['Agra', 'Aligarh', 'Allahabad'], 'population': 199812341},
{'name': 'Maharashtra', 'cities': ['Mumbai', 'Pune', 'Solapur'], 'population': 112374333},
{'name': 'Gujarat', 'cities': ['Ahmedabad', 'Baroda', 'Rajkot'], 'population': 60439692},
{'name': 'Rajasthan', 'city': ['Udaipur', 'Jesalmer', 'Kumbhalgarh'], 'population': 68548437}
]
{'name': 'Gujarat', 'cities': ['Ahmedabad', 'Baroda', 'Rajkot'], 'population': 60439692}
```

Dictionaries (Predict the output)

www.hbpatel.in

```
arr = {}
arr[1] = 1
arr['1'] = 2
arr[1] += 1

sum = 0
for k in arr:
    sum += arr[k]

print(arr)
print (sum)
```

```
{1: 2, '1': 2}
4
```

```
fruit = {}

def addone(index):
    if index in fruit:
        fruit[index] += 1
    else:
        fruit[index] = 1

addone('Apple')
addone('Banana')
addone('apple')

print(fruit)
print (len(fruit))

addone('Apple')
print(fruit)
print (len(fruit))
```

```
{'Apple': 1, 'Banana': 1, 'apple': 1}
3
{'Apple': 2, 'Banana': 1, 'apple': 1}
3
```