

1. Create a list of student names and print the second and last student.

Input:



```
1 list = [ " Apple" , "Mango" ,"Jerry" , "Orange"]
2 for list in list:
3     print(list)
```

Output:

```
PS D:\python-clg> python -u "d:\python-clg\Assignment-5\prog1.py"
Apple
Mango
Jerry
Orange
PS D:\python-clg>
```

2. Create a list of 5 fruits and print all using a loop.

Input:



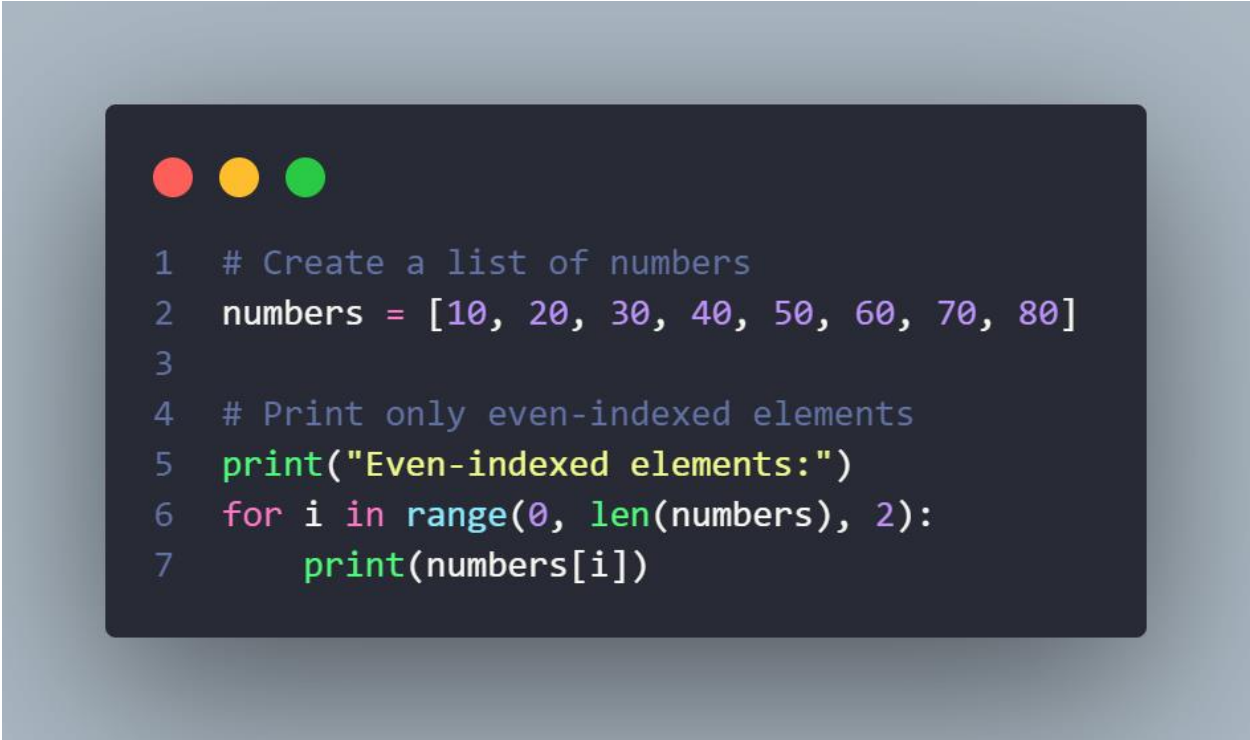
```
1 list = ["alex" , "jane" , "lisa" , "hinal"]
2 print(list[2])
3 print(list[-1])
```

Output:

```
PS D:\python-clg> python -u "d:\python-clg\Assignment-5\prog2.py"
lisa
hinal
PS D:\python-clg>
```

3. Create a list of numbers and print only even-indexed elements

Input:



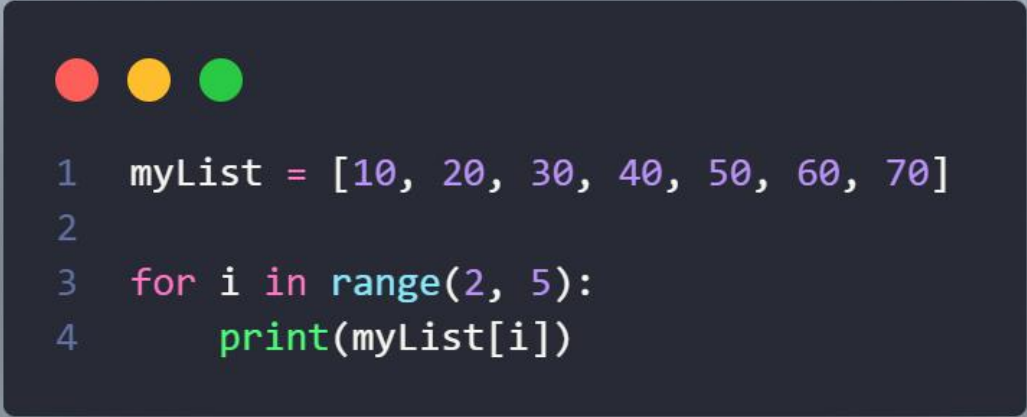
```
1 # Create a list of numbers
2 numbers = [10, 20, 30, 40, 50, 60, 70, 80]
3
4 # Print only even-indexed elements
5 print("Even-indexed elements:")
6 for i in range(0, len(numbers), 2):
7     print(numbers[i])
```

Output:

```
PS D:\python-clg> python -u "d:\python-clg\Assignment-5\prog3.py"
Even-indexed elements:
10
30
50
70
PS D:\python-clg>
```

4. Access and print a slice of a list using range (slicing).

Input:

A screenshot of a code editor window with a dark background and three colored window control buttons (red, yellow, green) in the top left corner. The code is written in a syntax-highlighted font. It defines a list 'myList' with values [10, 20, 30, 40, 50, 60, 70] and then uses a 'for' loop with 'range(2, 5)' to iterate over indices 2, 3, and 4, printing the corresponding elements from the list.


```
1 myList = [10, 20, 30, 40, 50, 60, 70]
2
3 for i in range(2, 5):
4     print(myList[i])
```

Output:

```
PS D:\python-clg> python -u "d:\python-clg\Assignment-5\prog4.py"
30
40
50
PS D:\python-clg>
```

5. Create a list with mixed data types and access each type

Input:



```
1 list = ["hello", 5, False, 78.14]
2
3 print("Original List: ",list)
4 print(" ")
5 print("List's First Index Type: ",type(list[0]))
6 print("List's First Index Type: ",type(list[1]))
7 print("List's First Index Type: ",type(list[2]))
8 print("List's First Index Type: ",type(list[3]))
```


Output:

```
PS D:\python-clg> python -u "d:\python-clg\Assignment-5\prog4.py"
Original List: ['hello', 5, False, 78.14]

List's First Index Type: <class 'str'>
List's First Index Type: <class 'int'>
List's First Index Type: <class 'bool'>
List's First Index Type: <class 'float'>
PS D:\python-clg>
```

6. Append elements to a list dynamically using user input.

Input:



```

1  list = []
2
3  num = int(input("Enter Number of element you want to add in list :"))
4
5  i = 0
6
7  while (i < num):
8      myList = input("Enter your element :")
9      list.append(myList)
10     i += 1
11
12  print(list)

```

Output:


```

PS D:\python-clg> python -u "d:\python-clg\Assignment-5\prog6.py"
Enter Number of element you want to add in list :4
Enter your element :ninja
Enter your element :jake
Enter your element :alex
Enter your element :lisa
['ninja', 'jake', 'alex', 'lisa']
PS D:\python-clg> 

```

7. Copy a list and show the difference between original and copied list.

Input:



```

1  list = ["apple","cherry","banana","watermelon","pappaya","mango"]
2  print("Original List: ",list)
3  myList = list.copy()
4
5  print("Copied List: ",list)

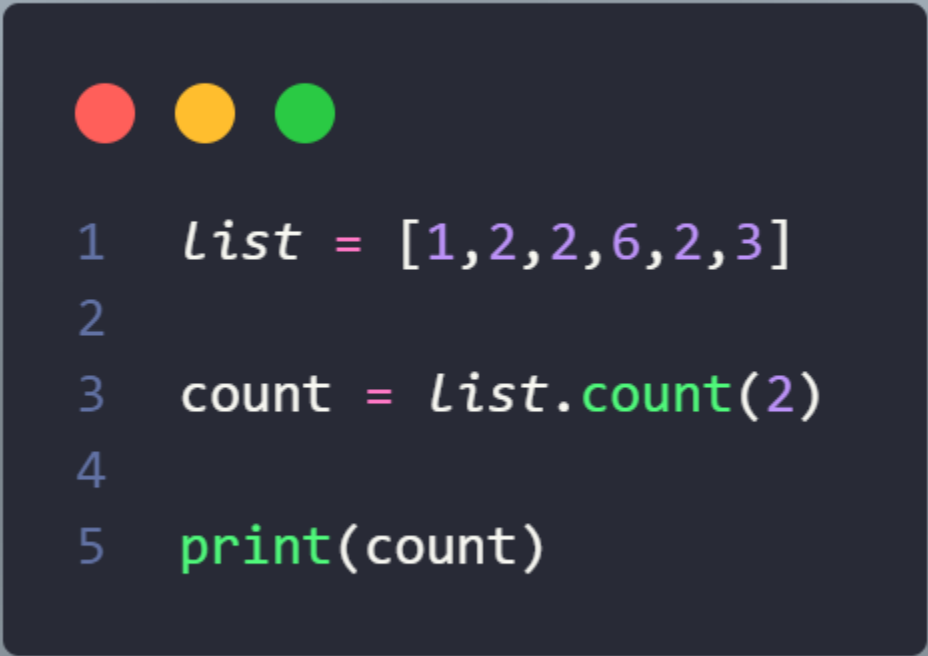
```

Output:

```
PS D:\python-clg> python -u "d:\python-clg\Assignment-5\prog7.py"
Original List: ['apple', 'cherry', 'banana', 'watermelon', 'pappaya', 'mango']
Copied List: ['apple', 'cherry', 'banana', 'watermelon', 'pappaya', 'mango']
PS D:\python-clg>
```

8. Count how many times a number appears in a list.

Input:




```
1 list = [1,2,2,6,2,3]
2
3 count = list.count(2)
4
5 print(count)
```

Output:

```
PS D:\python-clg> pyth
3
PS D:\python-clg>
```

9. Use insert() to add a value at a specific index.

Input:



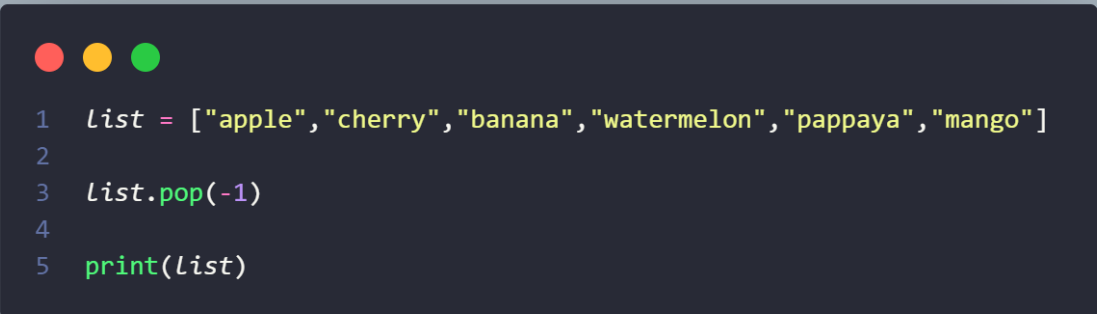
```
1 list = ["apple", "cherry", "banana", "watermelon", "pappaya", "mango"]
2
3 list.insert(3, "dragon fruit")
4
5 print(list)
```

Output:

```
PS D:\python-clg> python -u "d:\python-clg\Assignment-5\prog9.py"
['apple', 'cherry', 'banana', 'dragon fruit', 'watermelon', 'pappaya', 'mango']
PS D:\python-clg>
```

10. Use pop() to remove and display the last item.

Input:



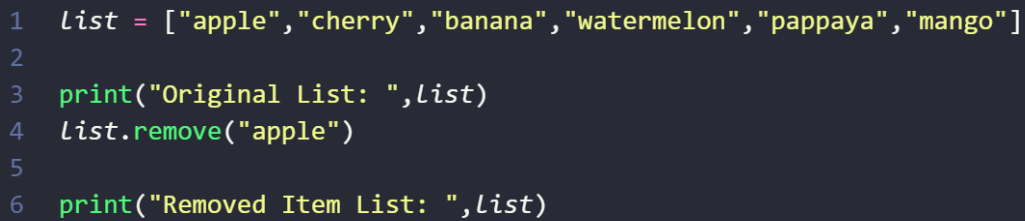
```
1 list = ["apple", "cherry", "banana", "watermelon", "pappaya", "mango"]
2
3 list.pop(-1)
4
5 print(list)
```

Output:

```
PS D:\python-clg> python -u "d:\python-clg\Assignment-5\prog10.py"
['apple', 'cherry', 'banana', 'watermelon', 'pappaya']
PS D:\python-clg>
```

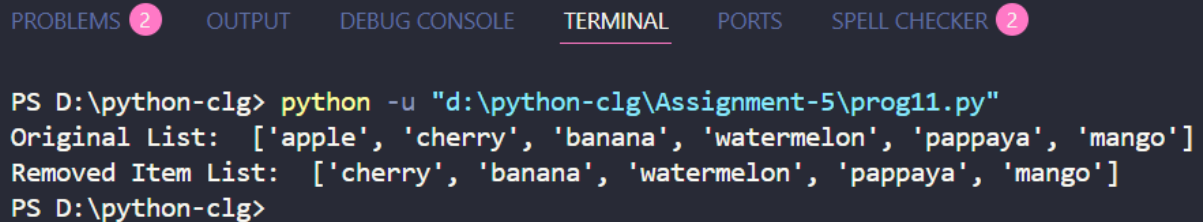
11. Use remove() to delete a specific item by value

Input:

A code editor window with a dark background and three colored window control buttons (red, yellow, green) in the top left corner. The code is written in a light green monospace font. It defines a list 'List' with six fruit names, prints the original list, removes the first element 'apple', and prints the updated list.

```
1 List = ["apple","cherry","banana","watermelon","pappaya","mango"]
2
3 print("Original List: ",List)
4 List.remove("apple")
5
6 print("Removed Item List: ",List)
```


Output:

A terminal window with a dark background. At the top, there are tabs for 'PROBLEMS' (with a pink circle containing '2'), 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL' (which is underlined), 'PORTS', and 'SPELL CHECKER' (with a pink circle containing '2'). The terminal shows the execution of a Python script, displaying the original list and the list after removing 'apple'.

```
PS D:\python-clg> python -u "d:\python-clg\Assignment-5\prog11.py"
Original List: ['apple', 'cherry', 'banana', 'watermelon', 'pappaya', 'mango']
Removed Item List: ['cherry', 'banana', 'watermelon', 'pappaya', 'mango']
PS D:\python-clg>
```

12. Use clear() to empty the entire list and print it

Input:



```
1 list = ["apple","cherry","banana","watermelon","pappaya","mango"]
2 print("Original List: ",list)
3 list.clear()
4 print("Cleared List: ",list)
```

Output:

```
PS D:\python-clg> python -u "d:\python-clg\Assignment-5\prog12.py"
Original List: ['apple', 'cherry', 'banana', 'watermelon', 'pappaya', 'mango']
Cleared List: []
PS D:\python-clg>
```

13. Create a list of numbers and sort them in ascending and descending order

Input:




```
1 list = [1,3,2,4,5,6]
2
3 list.sort()
4 print("Ascending Order: ",list)
5 list.sort(reverse = True)
6 print("Descending Order: ",list)
```

Output:

```
PS D:\python-clg> python -u "d:\python-
Ascending Order:  [1, 2, 3, 4, 5, 6]
Descending Order: [6, 5, 4, 3, 2, 1]
```

14. Reverse a list using reverse() method and print both original and reversed list.

Input:




```
1 list = [1,3,2,4,5,6]
2
3 list.sort()
4 print("Ascending Order: ",list)
5 list.sort(reverse = True)
6 print("Descending Order: ",list)
```

Output:

```
PS D:\python-clg> python -u "d:\python-clg\Assignment-5\prog14.py"
Original List: ['apple', 'cherry', 'banana', 'watermelon', 'pappaya', 'mango']
Reversed List: ['mango', 'pappaya', 'watermelon', 'banana', 'cherry', 'apple']
```

15. Find the index of a given element using index() and print it.

Input:



```
1 list = ["apple","cherry","banana","watermelon","pappaya","mango"]
2 print(list.index("watermelon"))
```

Output:

```
PS D:\python-clg> python -u python-clg\Assignment-5\prog15.py
3
PS D:\python-clg>
```