python programs using lists from basics to advanced along with explanation with output

Below is a **complete learning path of Python List programs** — starting from **absolute basics → intermediate → advanced**, with **clear explanation and sample output** for each. This is ideal for **beginners, lab exams, interviews, and practice**, and matches the way you've been learning Python step-by-step.

🔷 PART 1: BASIC LIST PROGRAMS

1️⃣ Create and Print a List

Program

python

```python
numbers = [10, 20, 30, 40, 50]
print(numbers)
```

Copy code

Explanation

- A list stores multiple values in a single variable.
- Elements are enclosed in **square brackets []**.

Output

csharp

```
[10, 20, 30, 40, 50]
```

Copy code

2️⃣ Access List Elements (Indexing)

Program

python

```
fruits = ["Apple", "Banana", "Mango"]
print(fruits[0])
print(fruits[2])
```

Copy code

Explanation

- Index starts from **0**
- `fruits[0]` → first element

Output

nginx

```
Apple
Mango
```

Copy code

3️⃣ Find Length of List

Program

python

```
marks = [75, 80, 90, 85]
print(len(marks))
```

Copy code

Explanation

- `len()` returns number of elements in the list.

Output

```
4
```

Copy code

4️⃣ Iterate Through a List (for loop)

Program

python

```python
colors = ["Red", "Green", "Blue"]

for color in colors:
    print(color)
```

Copy code

Output

mathematica

```
Red
Green
Blue
```

Copy code

5️⃣ Add Elements to List

Program

python

```python
numbers = [1, 2, 3]
numbers.append(4)
print(numbers)
```

Copy code

Explanation

- `append()` adds element at the end.

Output

csharp

```
[1, 2, 3, 4]
```

Copy code

6️⃣ Insert Element at Specific Position

Program

python

```
numbers = [10, 20, 40]
numbers.insert(2, 30)
print(numbers)
```

Copy code

Output

csharp

```
[10, 20, 30, 40]
```

Copy code

---

7 Remove Elements from List

Program

python

```
fruits = ["Apple", "Banana", "Mango"]
fruits.remove("Banana")
print(fruits)
```

Copy code

Output

css

```
['Apple', 'Mango']
```

Copy code

---

8 Delete Element Using Index

Program

python

```
numbers = [10, 20, 30]
del numbers[1]
print(numbers)
```

Copy code

Output

csharp

```
[10, 30]
```

Copy code

🔷 PART 2: INTERMEDIATE LIST PROGRAMS

9️⃣ Find Largest and Smallest Number

Program

python

```python
numbers = [45, 12, 89, 23]

print("Largest:", max(numbers))
print("Smallest:", min(numbers))
```

Copy code

Output

makefile

```makefile
Largest: 89
Smallest: 12
```

Copy code

🔟 Sum of List Elements

Program

python

```python
numbers = [10, 20, 30, 40]
print("Sum =", sum(numbers))
```

Copy code

Output

ini

```ini
Sum = 100
```

Copy code

1️⃣1️⃣ Count Occurrences of an Element

Program

python

```python
nums = [1, 2, 2, 3, 2]
print(nums.count(2))
```

Copy code

Output

```
3
```

Copy code

1️⃣2️⃣ Sort a List

Program

python

```python
numbers = [45, 12, 89, 23]
numbers.sort()
print(numbers)
```

Copy code

Output

csharp

```csharp
[12, 23, 45, 89]
```

Copy code

1️⃣3️⃣ Reverse a List

Program

python

```python
numbers = [1, 2, 3, 4]
numbers.reverse()
print(numbers)
```

Copy code

Output

csharp

```
[4, 3, 2, 1]
```

Copy code

1️⃣4️⃣ Copy a List

Program

python

```
a = [1, 2, 3]
b = a.copy()
print(b)
```

Copy code

Output

csharp

```
[1, 2, 3]
```

Copy code

1️⃣5️⃣ Merge Two Lists

Program

python

```
list1 = [1, 2]
list2 = [3, 4]
list3 = list1 + list2
print(list3)
```

Copy code

Output

csharp

```
[1, 2, 3, 4]
```

Copy code

🔷 PART 3: ADVANCED LIST PROGRAMS

1️⃣6️⃣ Remove Duplicate Elements

Program

python

```python
numbers = [1, 2, 2, 3, 4, 4]
unique = list(set(numbers))
print(unique)
```

Copy code

Output

csharp

```csharp
[1, 2, 3, 4]
```

Copy code

1️⃣7️⃣ List Comprehension (Square of Numbers)

Program

python

```python
squares = [x*x for x in range(1, 6)]
print(squares)
```

Copy code

Explanation

- Short and powerful way to create lists.

Output

csharp

```csharp
[1, 4, 9, 16, 25]
```

Copy code

1️⃣8️⃣ Find Even Numbers from List

Program

python

```python
numbers = [1, 2, 3, 4, 5, 6]
evens = [n for n in numbers if n % 2 == 0]
print(evens)
```

Copy code

Output

csharp

```
[2, 4, 6]
```

Copy code

1️⃣9️⃣ Nested List (2D List)

Program

python

```python
matrix = [
    [1, 2],
    [3, 4]
]

for row in matrix:
    print(row)
```

Copy code

Output

csharp

```
[1, 2]
```

2️⃣0️⃣ Flatten a Nested List

Program

python

Copy code

```
matrix = [[1, 2], [3, 4], [5, 6]]
flat = [num for row in matrix for num in row]
print(flat)
```

Output

csharp

Copy code

```
[1, 2, 3, 4, 5, 6]
```

2️⃣1️⃣ Check if List is Empty

Program

python

Copy code

```
lst = []

if not lst:
    print("List is empty")
```

Output

csharp

Copy code

```
List is empty
```

2️⃣2️⃣ List as Stack (LIFO)

Program

python

Copy code

```
stack = []
stack.append(10)
stack.append(20)
stack.pop()
print(stack)
```

Output

```csharp
[10]
```

2️⃣3️⃣ List as Queue (FIFO)

Program

```python
queue = []
queue.append(10)
queue.append(20)
queue.pop(0)
print(queue)
```

Output

```csharp
[20]
```

🔷 PART 4: EXAM & INTERVIEW PROGRAMS

2️⃣4️⃣ Find Second Largest Element

```python
nums = [10, 20, 4, 45, 99]
nums = list(set(nums))
nums.sort()
print(nums[-2])
```

Output

```
45
```

2️⃣5️⃣ Rotate List by One Position

python

Copy code

```python
lst = [1, 2, 3, 4]
lst = lst[1:] + lst[:1]
print(lst)
```

Output

csharp

Copy code

```
[2, 3, 4, 1]
```

2️⃣6️⃣ Check Palindrome List

python

Copy code

```python
lst = [1, 2, 3, 2, 1]

if lst == lst[::-1]:
    print("Palindrome")
else:
    print("Not Palindrome")
```

Output

nginx

Copy code

```
Palindrome
```

✅ SUMMARY

✔ Basics → Creation, Access, Add, Remove

✔ Intermediate → Sort, Search, Merge

✅ Advanced → List Comprehension, Nested Lists

✅ Exam-oriented & Interview Programs

If you want next:

- 🔥 **50 LIST LAB PROGRAMS**
- 🎯 **LIST MCQs with tricky options**
- 💡 **Time-complexity of list operations**
- 🧪 **College practical record format**
- 🚀 **Real-world projects using lists**

Just tell me 👍

Is this conversation helpful so far?