List in Dart – Detailed Teaching Guide with Real Examples

This document explains **List** in Dart in a detailed, beginner-friendly way. It includes **concepts, syntax, methods, real-life examples, Flutter use cases, and practice questions**.

---

1. What is a List?

A **List** is an ordered collection of values. It allows: - Duplicate values - Index-based access - Dynamic size (can grow or shrink)

Think of a List like: - A shopping list - A list of contacts - A playlist of songs - A to-do list

---

2. Creating a List

Example

```
List<int> numbers = [1, 2, 3, 4];
List<String> names = ["Alex", "Sam", "John"];
```

You can also use `var`:

```
var cities = ["Delhi", "Mumbai", "Pune"];
```

---

3. Accessing Elements (Indexing)

Lists use **zero-based indexing**.

```
List<String> fruits = ["Apple", "Banana", "Mango"];
print(fruits[0]); // Apple
print(fruits[2]); // Mango
```

---

4. Updating Elements

```
fruits[1] = "Orange";
print(fruits);
```

---

5. Common List Methods

```
fruits.add("Grapes");         // Add one item
fruits.addAll(["Pineapple"]); // Add multiple items
fruits.remove("Apple");       // Remove by value
fruits.removeAt(1);            // Remove by index
fruits.contains("Mango");     // Check existence
fruits.length;                 // Number of items
fruits.clear();                // Remove all items
fruits.isEmpty;                // Check if empty
fruits.isNotEmpty;             // Check if not empty
```

---

6. Looping Through a List

Using for loop

```
for (int i = 0; i < fruits.length; i++) {
  print(fruits[i]);
}
```

Using for-in loop

```
for (var fruit in fruits) {
  print(fruit);
}
```

---

7. Real-Life Examples

Example 1: Shopping Cart

```
List<String> cart = ["Apple", "Banana"];
cart.add("Mango");
```

Example 2: To-Do List

```
List<String> tasks = ["Wake up", "Study", "Exercise"];
tasks.remove("Study");
```

Example 3: Student Marks

```
List<int> marks = [85, 90, 78, 92];
```

---

8. List in Flutter (Real Usage)

Example: Showing a List of Items

```
List<String> products = ["Shoes", "Bag", "Watch"];
```

In Flutter, this list can be used in `ListView.builder`.

Example: Selected Items

```
List<String> selectedItems = [];
selectedItems.add("Apple");
```

---

9. Difference Between List, Set, and Map

Feature	List	Set	Map
Order	Maintained	Not guaranteed	No index
Duplicates	Allowed	Not allowed	Keys not allowed
Access	By index	By value	By key

10. Practical Questions

1. Why is a List better than a Set for a playlist?
2. Where would duplicates be useful in a List?
3. How would you store a user's browsing history using a List?
4. Why is indexing useful?

---

11. Guess the Output

5.

```
List<int> nums = [1, 2, 3];
nums.add(4);
nums.remove(2);
print(nums);
```

6.

```
List<String> names = ["A", "B", "C"];
names[1] = "D";
print(names);
```

7.

```
List<int> data = [];
print(data.isEmpty);
```

---

12. Coding Practice

1. Write a program to find the sum of all numbers in a List.
2. Write a program to find the largest number in a List.
3. Write a program to reverse a List.
4. Write a program to remove duplicates from a List.

---

13. Key Takeaway

- Use **List** when you need ordered data
- Lists allow duplicates
- Lists are widely used in Flutter UI
- Lists are dynamic and flexible

Understanding Lists deeply will help you manage data efficiently in Dart and Flutter apps.