**Collections**

In Flutter, collections are used to store and manage data in different structures such as lists, maps, sets, and queues. These collections provide various operations to manipulate and retrieve data efficiently. Here's an overview of different collections and their usage along with read and write operations in Flutter:

### 1. Lists:

Lists are ordered collections that store elements sequentially. In Dart (the language used in Flutter), lists can be of fixed length or grow dynamically.

\*\*Usage:\*\*

```dart

// Creating a list

List<int> numbers = [1, 2, 3, 4, 5];

// Accessing elements

print(numbers[0]); // Output: 1

// Modifying elements

numbers[2] = 10;

// Adding elements

numbers.add(6); // Adds 6 to the end of the list

// Removing elements

numbers.removeAt(3); // Removes element at index 3

```

### 2. Maps:

Maps store data in key-value pairs where each key is unique. They are used when data needs to be accessed using a specific identifier (key).

\*\*Usage:\*\*

```dart

// Creating a map

Map<String, int> ages = {

'Alice': 25,

'Bob': 30,

'Charlie': 22,

};

// Accessing values

print(ages['Bob']); // Output: 30

// Modifying values

ages['Charlie'] = 23;

// Adding new key-value pairs

ages['David'] = 28;

// Removing a key-value pair

ages.remove('Alice');

```

### 3. Sets:

Sets are unordered collections of unique elements. They don't allow duplicate values.

\*\*Usage:\*\*

```dart

// Creating a set

Set<String> countries = {'USA', 'Canada', 'UK', 'USA'};

// Sets don't allow duplicates, so 'USA' will only be stored once

// Adding elements

countries.add('France');

// Removing elements

countries.remove('UK');

```

### Read and Write Operations:

#### Reading Data:

```dart

List<int> numbers = [1, 2, 3, 4, 5];

// Iterating through a list

for (int num in numbers) {

print(num); // Prints each number in the list

}

Map<String, int> ages = {

'Alice': 25,

'Bob': 30,

'Charlie': 22,

};

// Iterating through a map

ages.forEach((key, value) {

print('$key: $value'); // Prints each key-value pair in the map

});

Set<String> countries = {'USA', 'Canada', 'UK', 'USA'};

// Iterating through a set

for (String country in countries) {

print(country); // Prints each country in the set

}

```

#### Writing/Modifying Data:

```dart

List<int> numbers = [1, 2, 3, 4, 5];

// Modifying elements

numbers[2] = 10; // Modifies the value at index 2

Map<String, int> ages = {

'Alice': 25,

'Bob': 30,

'Charlie': 22,

};

// Adding new key-value pairs

ages['David'] = 28; // Adds a new entry for 'David'

Set<String> countries = {'USA', 'Canada', 'UK', 'USA'};

// Adding elements

countries.add('France'); // Adds 'France' to the set

```

These collections in Flutter provide a powerful way to organize and manage data, allowing developers to efficiently perform various operations like reading, writing, modifying, and removing elements based on specific requirements.