SET:

There are different ways to implement Set data structures depending on the behavior you want, similar to HashSet, LinkedHashSet, and SplayTreeSet in Java.

Dart's dart: collection library provides equivalents.

Here's how to **create and use** each:

1. HashSet

Unordered set based on hash codes. Fast lookup and insertion.

```
import 'dart:collection';

void main() {
  HashSet<String> hashSet = HashSet();

  hashSet.add("apple");
  hashSet.add("banana");
  hashSet.add("cherry");

  print("HashSet: $hashSet"); // Order is not guaranteed
}
```

2. LinkedHashSet

Maintains insertion order (like LinkedHashSet in Java).

```
import 'dart:collection';

void main() {
  LinkedHashSet<String> linkedHashSet = LinkedHashSet();

  linkedHashSet.add("apple");
  linkedHashSet.add("banana");
  linkedHashSet.add("cherry");
```

```
  print("LinkedHashSet: $linkedHashSet"); // Maintains insertion order
}
```

3. SplayTreeSet

Sorted set using a self-balancing binary search tree.

```
import 'dart:collection';

void main() {
  SplayTreeSet<String> splayTreeSet = SplayTreeSet();

  splayTreeSet.add("banana");
  splayTreeSet.add("cherry");
  splayTreeSet.add("apple");

  print("SplayTreeSet: $splayTreeSet"); // Automatically sorted
}
```

MAP:

A Map is a collection of key-value pairs. Each key is unique. It's like a dictionary in Python.

1. Using Map literals

```
void main() {
  Map<String, int> ages = {
    'Alice': 25,
    'Bob': 30,
    'Charlie': 28,
  };

  print(ages);
}
```

2. Using the Map constructor

```
void main() {
  Map<String, String> capitals = Map();
```

```
  capitals['India'] = 'New Delhi';
  capitals['USA'] = 'Washington D.C.';
  capitals['France'] = 'Paris';

  print(capitals);
}
```

3. Using map.of ():

```
void main () {
  Map<String, int> original = {'a': 1, 'b': 2};

  Map<String, int> copy = Map.of (original);

  print(copy); // {a: 1, b: 2}
}
```

4. Using map.from ():

```
void main() {
  var planets = <num, String>{1: 'Mercury', 2: 'Venus'};
  var mapFrom = Map<int, String>.from(planets);
  print(mapFrom); // Output: {1: Mercury, 2: Venus}
}
```

5. MapEntry ():

```
void main() {
  var entry = MapEntry('name', 'Aarav');
  print('${entry.key}: ${entry.value}'); // Output:- name: Aarav
}
```