Mohammed Roshan Khan

Day 9 – Assignment

SET

A **Set** is an unordered collection of unique items (no duplicates).

```
import 'dart:collection';

void main() {

  // Default Set

  Set<int> defaultSet = {10, 20, 30, 40};

  print('Default Set: $defaultSet');
```

HashSet

- The default Set type in Dart.
- Unordered, fast lookup, and no duplicate elements.

```
var set = {'apple', 'banana', 'apple'};
print(set); // {apple, banana}
```

LinkedHashSet

- Maintains insertion order.
- Also prevents duplicates.

```
import 'dart:collection';

var set = LinkedHashSet.from(['a', 'b', 'a']);
```

```
print(set); // {a, b}
```

SplayTreeSet

- Automatically sorts elements.
- Uses a self-balancing binary search tree.

```
import 'dart:collection';

var set = SplayTreeSet.from([5, 2, 9, 2]);
print(set); // {2, 5, 9}
```

MAP

A **Map** stores key-value pairs. Keys must be unique.

Map Literal

- Define maps directly using {} syntax.

```
var map = {'name': 'Alice', 'age': 25};
print(map['name']); // Alice
```

Map Constructor

- Create a map using the `Map()` constructor.

```
var map = Map();
map['city'] = 'Paris';
```

```
print(map); // {city: Paris}
```

Empty Map

- Creates an empty map with no entries.

```
var map = <String, int>{};
print(map); // {}
```

Map.from()

- Creates a new map from another map.

```
var original = {'a': 1};
var copy = Map.from(original);
print(copy); // {a: 1}
```

Map.of()

- Similar to `Map.from()`, creates a new map from another.

```
var base = {'x': 10};
var newMap = Map.of(base);
print(newMap); // {x: 10}
```

Map.fromEntries()

- Builds a map from a list of MapEntry objects.

```
var entries = [MapEntry('k1', 100), MapEntry('k2', 200)];
var map = Map.fromEntries(entries);
print(map); // {k1: 100, k2: 200}
```