

Dart – Day11

Emp-id : 4781

- **Map.fromIterable**

Map.fromIterable() creates a map from a single iterable by using functions to generate keys and values.

- You must provide a key function and a value function.
- Each element in the iterable is passed to both.

Example:

```
void main()
{
  var numbers = [1, 2, 3, 4];

  var squareMap = Map.fromIterable(
    numbers,
    key: (n) => n,    // keys are numbers
    value: (n) => n * n // values are squares
  );

  print(squareMap); // {1: 1, 2: 4, 3: 9, 4: 16}
}
```

- **Map.fromIterables**

Map.fromIterables() creates a map by combining two iterables:

- One for keys
- One for values

Example:

```
void main()
```

```
{  
  var names = ["Alice", "Bob", "Charlie"];  
  var scores = [90, 85, 88];  
  
  var studentScores = Map.fromIterables(names, scores);  
  print(studentScores); // { Alice: 90, Bob: 85, Charlie: 88}  
}
```

- **Iterating Over a Map**

There are multiple ways to iterate through a map in Dart.

1. Iterate Keys Only

```
void main()  
{  
  var map = {"a": 1, "b": 2, "c": 3};  
  for (var key in map.keys) {  
    print("Key: $key");  
  }  
}
```

2. Iterate Values Only

```
void main()  
{  
  var map = {"a": 1, "b": 2, "c": 3};  
  for (var value in map.values) {  
    print("Value: $value");  
  }  
}
```

3. Iterate Key-Value Pairs with forEach

```
void main()  
{  
  var map = {"a": 1, "b": 2, "c": 3};
```

```
map.forEach((key, value) {  
    print("$key → $value");  
});  
}
```

4. Iterate Entries (MapEntry)

```
void main()  
{  
    var map = {"a": 1, "b": 2, "c": 3};  
    for (var entry in map.entries) {  
        print("${entry.key}: ${entry.value}");  
    }  
}
```

5. Iterating Map with entries.toList()

```
void main()  
{  
    var studentScores = {  
        "Alice": 90,  
        "Bob": 85,  
        "Charlie": 88,  
    };  
  
    var entriesList = studentScores.entries.toList();  
  
    for (int i = 0; i < entriesList.length; i++) {  
        var entry = entriesList[i];  
        print("Student: ${entry.key}, Score: ${entry.value}");  
    }  
}
```

- **Ternary Operator**

The ternary operator is like a short form of if-else.

Syntax:

condition ? expressionIfTrue : expressionIfFalse;

Example:

```
void main()
{
    int age = 20;
    String result = (age >= 18) ? "Adult" : "Minor";
    print(result); // Adult
}
```

- **Chaining Ternary Operator**

You can chain multiple ternary operators, but be careful with readability.

Example:

```
void main()
{
    int marks = 75;

    String grade = marks >= 90 ? "A"
                        : marks >= 75 ? "B"
                        : marks >= 60 ? "C"
                        : "Fail";

    print(grade); // B
}
```

- **Null Coalescing Operator (??)**

The null coalescing operator is used to handle null values safely.

?? → Provides a fallback value if the left side is null.

Example:

```
void greet({String? name, int? age})
{
  print('Hello $name, your age is ${age ?? "Not disclosed"}');
}
void main()
{
  greet(name:"Chandini"); // Hello Chandini, your age is Not disclosed
}
```

- **Types of Parameters in Dart**

1. Positional Parameters

These are the default parameters that must be passed in order.

```
void greet(String name, int age)
{
  print("Hello $name, you are $age years old");
}

void main() {
  greet("Chandini", 21);
}
```

2. Optional Positional Parameters

Use [] to make parameters optional.

```
void greet(String name, [String? city])
{
  print("Hello $name from ${city ?? "Unknown City"}");
}
```

```
void main() {  
    greet("Chandini");    // Without city  
    greet("Chandini", "Salem"); // With city  
}
```

3. Named Parameters

Use {} to pass parameters by name (order doesn't matter).

```
void greet({required String name, int age = 18})  
{  
    print("Hello $name, age: $age");  
}
```

```
void main() {  
    greet(name: "Chandini", age: 21);  
    greet(name: "Sneha"); // age will take default 18  
}
```

4. Mix of Positional and Named

You can combine them.

```
void greet(String role, {required String name, int age = 18})  
{  
    print("$role: $name, age $age");  
}
```

```
void main() {  
    greet("Student", name: "Chandini", age: 21);  
}
```