What's New in Dart 3?

Shrihriday Bhagwat

@shreebhagwt94

in/shrihriday

Dart 3 is the default version for Flutter 3.10

droidcon academy

Section Overview

- Upgrading to the latest Flutter and Dart Version
- Records
  - Return Multiple Values
    - Positional Values
    - Named Values
- Pattern Matching
  - Syntax and Structure Matching
  - JSON Pattern Matching
- Switch Statements
- Class Modifiers
  - Introduction to class modifiers.

Upgrading to Dart 3

Check Flutter and Dart Version

```
flutter doctor
```

If dart version < 3

```
flutter upgrade
```

Records

Return Value From Function

```
String myPetName(){
     String petName = 'Brownie’;
     return petName;
}
```

Single Value

What about multiple Values?

Map<String, dynamic>

- Instead of returning a String or Integer we return a Collection like Map so that we get multiple values out of the Function

Flaws or Drawbacks

- Flaws like type safe, data binding.

```
Map<String, dynamic> myPet(){
    Map<String, dynamic> petDetails = {
        'petName': 'Brownie’,
        'petType': 'Doggo’
        };
    return petDetails
}

final petType = myPet()['petType'];
final petName = petName()['petType'];
```

Create Class Object

- A robust and verbose method is to create a Model Class
- Use class object to return specified values

```
class Pet {
String petName;
String petType;
//...

Pet({required this.petName, required this.petType});

}

Pet myPet(){
final pet = Pet(petName: 'Brownie', petType: 'Doggo');
return pet;
}

final pet = mypet();
final petName = pet.petName;
final petType = pet.petType;
```

droidcon academy

How Dart 3 Tackles this?

```
(String, String, int) getMyPet(){
      String petName = 'Brownie';
      String petType = 'Dog';
      int petAge = 3;
      return (petName, petType, petAge);
}


final mypet = getMypet();
print(myPet);
```

Dart 3 Records Preview

- Instead of Returning single value, **dart 3** creates a record and returns the record of data.

Lets check this out

Patterns

List

- To get object out of the list using the index value

Map

- To get object out of the map using
- Key of the data

```
final list = ['Animals', 'Birds', 'Fish'];
print(list[0])



final mapData = {
     'name' : 'Brownie’
     'type': 'Dog’
     'breed': 'labrador'
}

print(mapData['name']);
```

droidcon academy

With Pattern Matching

- The syntax and the structure of the Pattern that extracts the data matches with the syntax and structure if the data it self

```
final list = ["Animals","Birds","Fish"];
final [a, b, c] = list;
print(a)
print(b)
print(c)
```

Lets check pattern matching out

droidcon academy

Switch Statement

- Using switch statement for decision making before dart 3
- We have to mention the case always before we can return any value.

```
petType getPetType() {
    String animalName = "Brownie";
    switch (animalName) {
        case "Brownie":
            return petType.Dog;
        case "Mittens":
            return petType.Cat;
        case "Bugs":
            return petType.Rabbit;
            default:
            return petType.Dog;
    }
}
```

Lets check switch statement out

Class Modifiers

- Class modifiers are used to add modification to class which affects its runtime behavior.

- eg: abstract

```
abstract class Animal {}
```

New Class Modifiers

- sealed :- Manage all the subclasses when used in a switch statement
- final :- Like sealed class, but you can create instance of the class
- base :- Base classes cannot be implemented, but can be extended, class extending a base class must be a base, final or sealed class.
- mixin :- Cannot be instantiated, but can be mixed in. Now only works be used with with keyword
- abstract :- Cannot be instantiated, but can be extended.

Lets check Modifiers out

Thank You