

Dart Functions

Data Types

Expanded Widget



Syntax of Functions Dart

```
void printHello() {  
    print("Hello");  
}
```

```
String getText(String name) {  
    return 'Name: $name !';  
}
```

Syntax of Anonymous Functions

```
ElevatedButton(  
  onPressed: () {  
    print("السعودية ");  
  },  
  child: Text(  
    "السعودية",  
    style: TextStyle(fontSize: 30, color: Colors.green),  
  ),  
)
```

Passing Function as Parameter

```
ElevatedButton(  
  onPressed: printHello,  
  child: Text(  
    "Say Hi",  
    style: TextStyle(fontSize: 30, color: Colors.green),  
  ),  
)
```

Syntax of Arrow Functions

```
ElevatedButton(  
  onPressed: () => printHello("Hi"),  
  child: Text(  
    "Say Hi",  
    style: TextStyle(fontSize: 30, color: Colors.green),  
  ),  
)
```

Data Types in Dart

- Numbers (`int`, `double`)
- Strings (`String`)
- Booleans (`bool`)
- Lists (`List`, also known as *arrays*)
- Maps (`Map`)

Numbers

- Examples:

```
int x = 10;
```

```
double y = 1.5;
```

- Declaring a variable as a num which can have both integer and double values:

```
num n = 1;
```

```
n += 2.5;
```

Strings

- Examples:

```
String name = "Mohammed";
```

- Turning a string into a number:

```
int one = int.parse('1');
```

```
double oneHalf = double.parse('1.5');
```


boolean

- To represent boolean values, Dart has a type named `bool`.
- Only two objects have type `bool`: the boolean literals `true` and `false`
- Examples:

```
bool isValid = true;
```

Lists

- Ordered collection of items (like arrays).
- Examples:

```
List<int> list = [1, 2, 3];
```

```
List<String> names = ['Khaled', 'Fahad', 'Mohammed'];
```

- Access by index:

```
list[1]
```

Maps

- A collection of key-value pairs.
- Examples:

```
Map<String, int> scores = { 'Fahad': 80, 'Mohammed': 75 };
```

- Access by index:

```
scores[ 'Fahad' ];
```

dynamic

- The **dynamic** keyword can hold any data type and can be reassigned to a different type.
- Example:

```
dynamic data = 'Hello';
```

```
data = 100;
```

var

- The **var** keyword Automatically infers the data type based on the initial value.
- Example:

```
var name = 'Mohammed';
```

name has assigned String datatype.

final vs. const

- **final**: Immutable variable, can be set once.

```
final country = "Saudi Arabia";
```

- **const**: Compile-time constant, deeply immutable.

```
const pi = 3.14159;
```

Null Safety in Dart

- Dart introduced null safety to prevent null reference errors.
- Variables can't contain null values unless explicitly declared as nullable (?).
- Example: `int? age = null;`
- Helps in writing more robust and error-free code.

Preventing *out of screen* error in Flutter

- Use `SafeArea` to ensures that widgets are positioned within the safe boundaries of the screen, avoiding areas like notches or rounded screen edges.
- Example:

```
SafeArea(  
  child: Column(  
    children: <Widget>[  
      // Your content here  
    ],),)
```


Preventing *out of screen* error in Flutter

- Wrap Widgets `Expanded` Widget inside Row or Column widgets to prevent widgets from taking up more space than available.

- Example:

```
Row(  
  children: <Widget>[  
    Expanded(  
      child: Text("Expanding text that adjusts to screen  
size"),),],),
```

تذكير

الاختبار الفصلي الأول

الأسبوع القادم

ان شاء الله

Practice

- Write a function that takes two parameters (`student_name`, `score`), and return a sentence with grade: `Student got B+`.
- Create an `ElevatedButton` that call this function.
- Create a List of students.
- Write another function that do the same for all students.