

Strings in dart: strings are a fundamental data type used to represent a sequence of characters. They are commonly used to store and manipulate text.

Strings in Dart are enclosed in single quotes (') or double quotes ("):

Ex: void main()

```
{  
    String name = 'Alice';  
    String greeting = "Hello";  
    Print("$greeting $name "); //Hello Alice  
}
```

Multiline Strings:

Use triple quotes for multi-line strings:

Ex: String multiline1 = "This is

a multiline
string."";

String multiline2 = ""This is

another multiline
string.""";

String Interpolation:

You can embed variables or expressions inside strings using \$:

Ex: void main()

```
{  
    String name = 'Bob';
```

```
    print('Hello, $name'); // Hello, Bob

    int age = 30;

    print('Next year, you will be ${age + 1} years old.');//Next year, you will be 30 years old.

}
```

Methods of string:

1) trim(): Removes leading and trailing whitespace from a string.

Ex: void main()

```
{

    String message = "  Hello Dart!  ";

    String res = message.trim();

    print(res); // Hello Dart!

}
```

2) toUpperCase(): Converts all characters in the string to uppercase.

Ex: void main()

```
{

    String name = "dart programming";

    String up = name.toUpperCase();

    print(up); //DART PROGRAMMING

}
```

3) contain(): Checks if a string contains a given substring.

Ex: void main()

```
{

    String sentence = "Learning Dart is fun!";
```

```

print(sentence.contains("Dart")); // true

print(sentence.contains("Java")); // false

}

```

Raw String: A raw string treats all characters literally, meaning , No escape sequences like \n, \t, \\, etc., are processed

Ex: void main()

```

{

    print("hello\nhi"); // normal string , Output: hello
                                hi

    print(r"hello\nhi"); // raw string, Output: hello\nhi

}

```

Records in dart: Records are a new feature in Dart 3 that let you group multiple values together into a single compound value without defining a class.

Types of records:

- 1) Positional record: Values are stored by position. Accessed using \$1, \$2, etc.

```

Ex: void main()
{
    var person = ("Alice", 25);
    print(person.$1);    // Alice
    print(person.$2);    // 25
}

```

- 2) Named record: Values stored with names (keys). Accessed using the field names.

```

Ex: void main()
{
    var person = (name: "Alice", age: 25);
    print(person.name);  // Alice
    print(person.age);   // 25
}

```

```
}
```

Returning Multiple Values:

Ex1: (String, int) info()

```
{  
    return ("john", 15);  
}
```

void main()

```
{  
    var (name, age) = info();  
    print('$name is $age years old'); //john is 15 years old.  
}
```

Ex2: ({String name, int age, String email}) getUserProfile()

```
{  
    return (name: "Alice", age: 28, email: "alice@example.com");  
}
```

void main()

```
{  
    var (name: name, age: age, email: email) = getUserProfile();  
    print("Name: $name"); //Alice  
    print("Age: $age"); //28  
    print("Email: $email"); //alice@example.com
```

}