Strings

In this lesson, you will learn about strings and string literals in Dart.

We'll cover the following

What Are Strings?

A Dart string is a sequence of **UTF-16** code units. **UTF** stands for *Unicode Transformation Format*. Unicode is a set of characters in which each character is a unique code unit.

String Literals

At face-value, string literals are simply text encapsulated in quotation marks.

```
"This is a string"
```

```
'This is also a string'
```

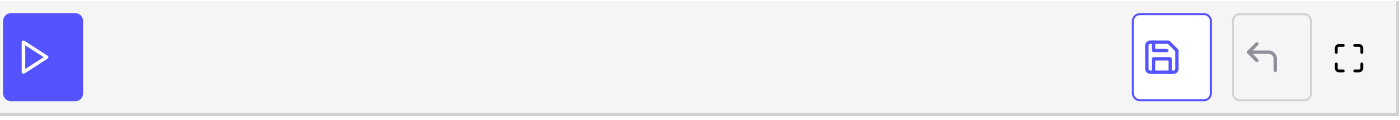
Let's look at the various ways we can create strings in Dart.

```
main() {
  // Single Quotes
  print('Using single quotes');

  // Double Quotes
  print("Using double quotes");

  // Single quotes with escape character \
  print('It\'s possible with an escape character');
```

```
  // Double quotes
  print("It's better without an escape character");
}
```



All we are doing in the code snippet above is printing strings using multiple techniques. They all are pretty straight forward except for maybe **line 9**. On **line 9** we are using single quotes to create a string, however, the string itself has a single quote in the word `It's`. If we print the string as-is, `'It's possible with an escape character'`, we will get an error because the compiler will see the single quotation in the word `It's` as the closing quotation of the string. To solve this problem, we can use an escape character (`\`) which tells the compiler to *escape* the built-in functionality of the single quotation and evaluate it as a string.

Defining a String Variable

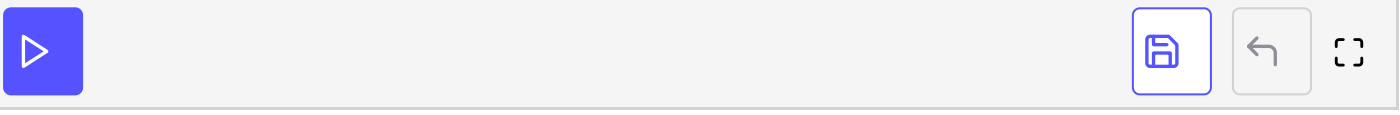
If we want to define a variable that stores values of type `String`, we would use the following syntax:

```
String variableName = String Literal
```

Let's map the syntax onto some actual Dart code.

```
main() {
    String s1 = "A String";

    print(s1);
}
```



Let's continue our discussion on strings in the next lesson.