String Length

A string in Java is actually an object, which contain properties and methods that can perform certain operations on strings. For example, the length of a string can be found with the length() method(在 Java 中，字串實際上是一個物件，包含了可以對字串執行特定操作的屬性和方法。例如，可以使用 length() 方法來取得字串的長度):

```
public class HelloWorld
{
        public static void main(String[] args)
        {
                String txt = "ABCD1234";
                System.out.println("The length of the txt string is: " + txt.length());
        }
}
```

ToUpperCase() and ToLowerCase()

There are many string methods available, for example ToUpperCase() and ToLowerCase(), which returns a copy of the string converted to uppercase or lowercase(有許多字串方法可供使用，例如 ToUpperCase() 和 ToLowerCase()，這些方法會返回一個將字串轉換為大寫或小寫的副本):

```
public class HelloWorld
{
      public static void main(String[] args)
       {
             String txt = "Hello World";
             System.out.println(txt.toUpperCase());     // Outputs "HELLO WORLD"
             System.out.println(txt.toLowerCase());     // Outputs "hello world"
       }
}
```

String Concatenation

The + operator can be used between strings to combine them. This is called **concatenation**:

```
public class HelloWorld
{
	public static void main(String[] args)
	{
		String firstName = "John ";
		String lastName = "Doe";
		String name = firstName + lastName;
		System.out.println(name);
	}
}
```

```
public class HelloWorld
{
	public static void main(String[] args)
	{
		String x = "10";
		String y = "20";
		String z = x + y;
		System.out.println(z);
	}
}
```

Access Strings

You can also find the index position of a specific character in a string, by using the indexOf() method(你也可以使用 indexOf() 方法來查找字串中特定字符的索引位置):

```
public class HelloWorld
{
    public static void main(String[] args)
    {
        String myString = "Hello";
        System.out.println(myString.indexOf("e"));
    }
}
```

Another useful method is substring(), which extracts the characters from a string, starting from the specified character position/index, and returns a new string(另一個有用的方法是 substring()，它會從字串中擷取字符，從指定的字符位置/索引開始，並返回一個新的字串):

```
public class HelloWorld
{
    public static void main(String[] args)
    {
        // Full name
        String name = "John Doe";

        // Get last name
        String lastName = name.substring(5);

        // Print the result
        System.out.println(lastName);
    }
}
```

substring(m, n) which extracts the characters from a string and returns a new string, the new string is starting from starting from the m th ordinal position/index, ending with the n th ordinal position/index in the original string(substring(m, n) 方法會從字串中擷取字符並回傳一個新的字串，該新字串從原字串第 m 個序數位置/索引開始，在原字串中第 n 個序數位置/索引結束):

```
public class HelloWorld
{
	public static void main(String[] args)
	{
		// Full name
		String name = "Stephen M.S. Lee";

		// Get last name
		String middleName = name.substring(8, 12);

		// Print the result
		System.out.println(middleName);
	}
}
```

Strings - Special Characters

The backslash (\) escape character turns special characters into string characters:

Escape character	Result	Description
\'	'	Single quote
\"	"	Double quote
\\	\	Backslash

```
public class HelloWorld
{
    public static void main(String[] args)
    {
        String txt = "We are the so-called \"Asian American\" from Asia.";
        System.out.println(txt);
    }
}
```

```
public class HelloWorld
{
    public static void main(String[] args)
    {
        String txt = "It\'s alright.";
        System.out.println(txt);
    }
}
```

```
public class HelloWorld
{
    public static void main(String[] args)
    {
        String txt = "The character \\ is called backslash.";
        System.out.println(txt);
    }
}
```

String Interpolation

Another option of string concatenation, is **string interpolation**, which substitutes values of variables into placeholders in a string. Note that you do not have to worry about spaces, like with concatenation(另一種字串串接的選擇是**字串插值**，它會將變數的值替換到字串中的佔位符。請注意，與串接不同，你不需要擔心空格的問題):

```
public class HelloWorld
{
    public static void main(String[] args)
    {
        String a = "W3School";
        String b = "coding";

        // Print and display the string Interpolated
        System.out.println(a + " is the best platform to learn " + b);
    }
}
```

The String format() method one of other ways to perform string interpolation in Java which uses placeholders, for example, “%s" to insert variable values into a string, with the variables provided as additional arguments after the format string(String 的 format() 方法是 Java 中執行字串插值的其中一種方式，它使用佔位符，例如 "%s"，將變數的值插入字串中，這些變數會作為格式字串後的附加引數提供):

```
public class HelloWorld
{
    public static void main(String[] args)
    {
        String a = "W3School";
        String b = "coding";

        // Print and display the string Interpolated
        System.out.println(String.format("%s is the best platform to learn %s", a, b));
    }

}
```