

Strings

Strings and their methods
– REMAINING SLIDES IN BONUS VIDEO

Produced Dr. Siobhán Drohan
by: Mr. Colm Dunphy
 Mr. Diarmuid O'Connor



Waterford Institute of Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

Department of Computing and Mathematics
<http://www.wit.ie/>

Topics list - **Strings**

1. Primitive Types: **char**
2. Object Types: **String**
3. **Primitive Types versus Object Types**
4. Strings and **Java API**
5. Strings - **methods**
6. **Method calls**
 - **Internal**
 - **External**
 - **Dot notation**
7. Using String methods: some **examples**

Strings are objects

- Variables created with the **String** data type are called **objects**.
- Objects are **software structures** that combine
 - **variables**
 - with **methods** that operate on those variables
 - e.g.
 - every String object has a built-in method that can capitalise its letters.

Strings and Java's API

- This link is to Java's **Application Programming Interface (API)**, version 8.

<https://docs.oracle.com/javase/8/docs/api/index.html?overview-summary.html>

- More information on the **String's methods**:

<https://docs.oracle.com/javase/8/docs/api/java/lang/String.html>

- Explore newer versions

Topics list - **Strings**

1. Primitive Types: **char**
2. Object Types: **String**
3. **Primitive** Types **versus** **Object** Types
4. Strings and **Java API**
5. Strings - **methods**
6. **Method Calls**
 - **Internal**
 - **External**
 - **Dot notation**
7. Using String methods: some **examples**

Strings - some API methods

Return Type	Method Name	Description
int	length()	Returns the length of this string.
String	toLowerCase()	Converts all of the characters in this String to lower case.
String	toUpperCase()	Converts all of the characters in this String to upper case.
String	trim()	Returns a string whose value is this string, with any <i>leading and trailing</i> whitespace removed.
String	substring(int beginIndex, int endIndex)	Returns a string that is a substring of this string.
char	charAt(int index)	Returns the char value at the specified index.

Topics list - **Strings**

1. Primitive Types: **char**
2. Object Types: **String**
3. **Primitive** Types **versus** **Object** Types
4. Strings and **Java API**
5. Strings and **methods**
6. **Method calls**
 - ➔ – **Internal**
 - **External**
 - **Dot notation**
7. Using String methods: some **examples**

Strings and methods

- To use these built-in methods, we must first understand the difference between:
 - **Internal** method calls
 - and
 - **External** method calls

Internal method calls

```
void draw()
{
  background(204);
  drawX(0);
}
```

This is an *internal method call*...

*Calls
Invokes*

```
void drawX(int gray)
{
  stroke(gray);
  strokeWeight(20);
  line(0,5,60,65);
  line(60,5,0,65);
}
```

...to this method that
exists in the same sketch.


Internal method calls

- **drawX(0)** is a method call.
- The sketch has a method with the following *signature/header*:

void drawX(int gray)

- The *method call* *invokes* this method.
- As the method is **in the same sketch** as the call of the method, we call it an *internal method call*.
- Internal method calls have the **syntax**:
methodname (parameter-list)

Topics list - **Strings**

1. Primitive Types: **char**
2. Object Types: **String**
3. **Primitive** Types **versus** **Object** Types
4. Strings and **Java API**
5. Strings - **methods**
6. **Method calls**
 - **Internal**
 -  – **External**
 - **Dot notation**
7. Using String methods: some **examples**

External method calls

- We want to check the length of this String:

`String name = "Joe Soap";`

- Looking at the **String API**, we can see this method:

Return Type	Method	Description
int	length()	Returns the length of this string.

- A call to a method of another object is called an **external method call**.

(objects {e.g.String} are usually defined in their own separate files)

External method calls

- External method calls have the syntax:

object.methodname (parameter-list)


- To find out the length of this String:

```
String name = "Joe Soap";
```

- We make the following external method call:

```
name.length();
```

Topics list - **Strings**

1. Primitive Types: **char**
2. Object Types: **String**
3. **Primitive** Types **versus** **Object** Types
4. Strings and **Java API**
5. Strings - **methods**
6. **Method calls**
 - **Internal**
 - **External**
 -  – **Dot notation**
7. Using String methods: some **examples**

Dot Notation

- Java code can call methods of other objects using dot notation.

- The syntax is:

object.methodname (parameter-list)

- It consists of:

- An **object**
- A dot
- A method name
- The parameter(s) for the method



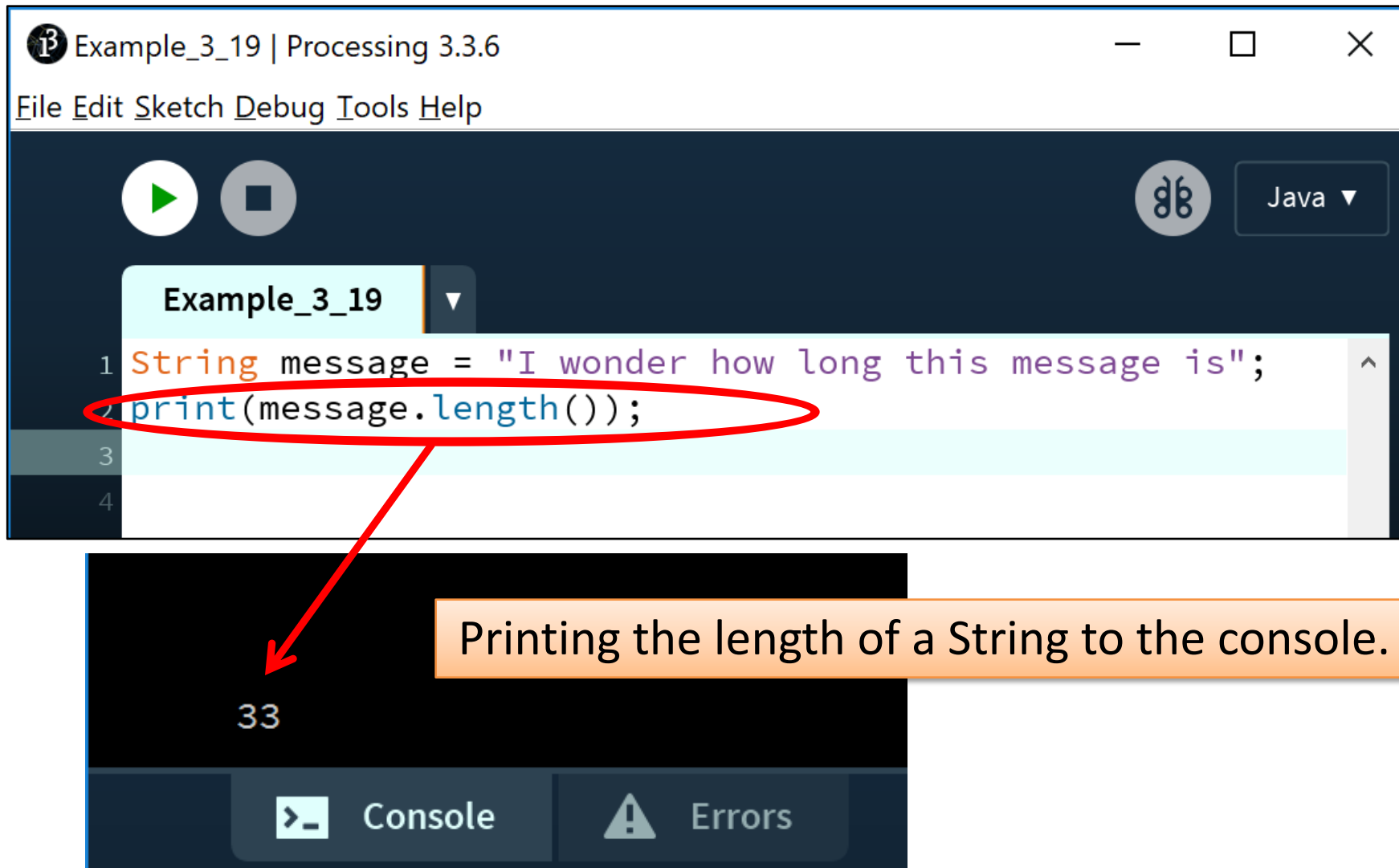
name.length();

The diagram shows the code snippet 'name.length();' enclosed in a blue rectangular box. Three red arrows originate from the list of components on the left and point to the corresponding parts of the code: one points to 'name', one to the dot '.', and one to 'length()'. The opening parenthesis of the parameter list is also enclosed in the box.

Topics list - **Strings**

1. Primitive Types: **char**
2. Object Types: **String**
3. **Primitive** Types **versus** **Object** Types
4. Strings and **Java API**
5. Strings - **methods**
6. **Method calls**
 - **Internal**
 - **External**
 - **Dot notation**
7. Using String methods: some **examples**

Example 3.19, Version 1



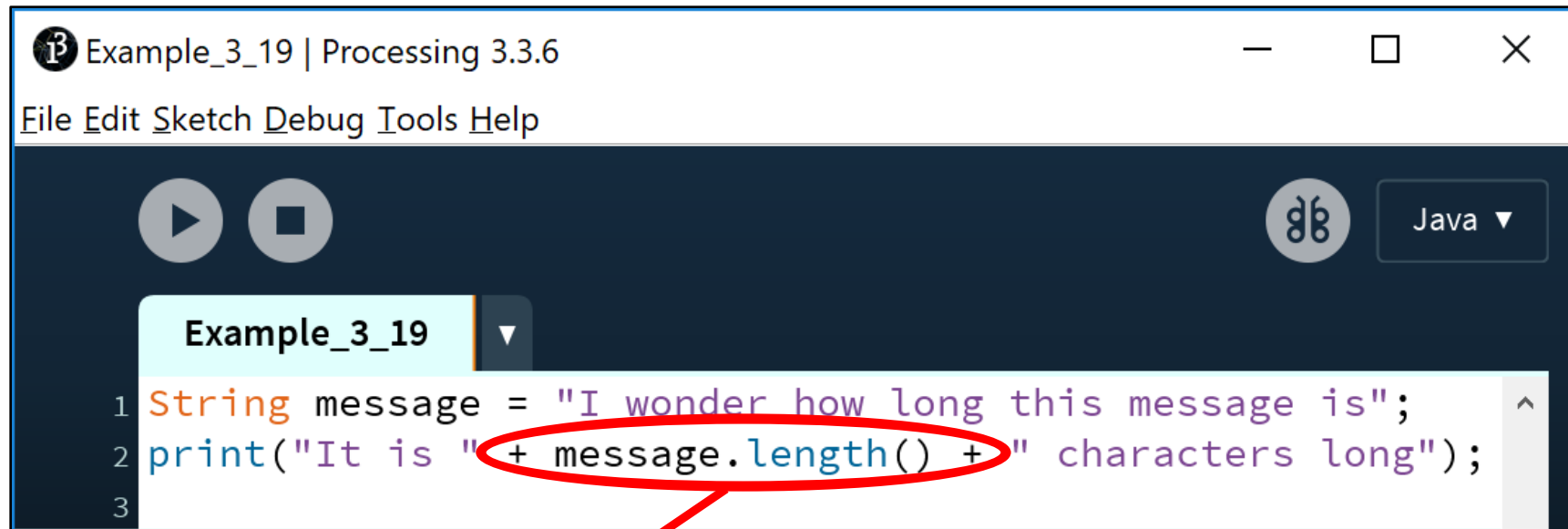
The screenshot displays the Processing IDE interface. The window title is "Example_3_19 | Processing 3.3.6". The menu bar includes "File", "Edit", "Sketch", "Debug", "Tools", and "Help". The toolbar shows a play button, a stop button, a gear icon, and a language dropdown set to "Java". The code editor shows the following code:

```
1 String message = "I wonder how long this message is";  
2 print(message.length());  
3  
4
```

The second line of code, `print(message.length());`, is circled in red. A red arrow points from this line to the console output, which shows the number "33". Below the console output, there are buttons for "Console" and "Errors".

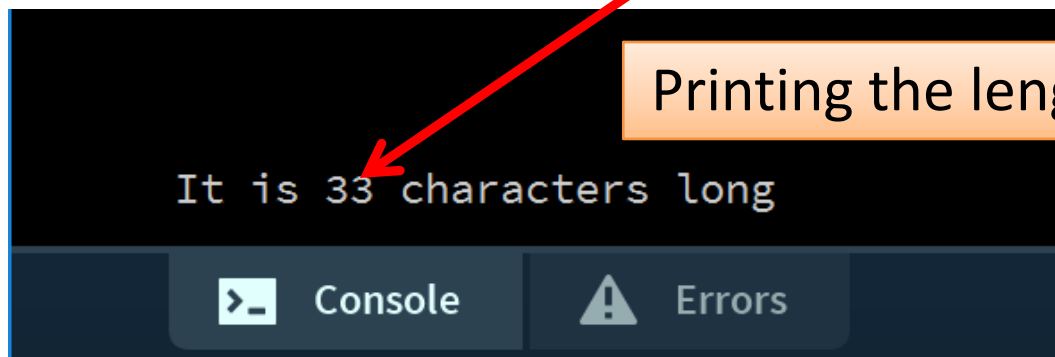
Printing the length of a String to the console.

Example 3.19, Version 2



```
Example_3_19 | Processing 3.3.6
File Edit Sketch Debug Tools Help

Example_3_19
1 String message = "I wonder how long this message is";
2 print("It is " + message.length() + " characters long");
3
```



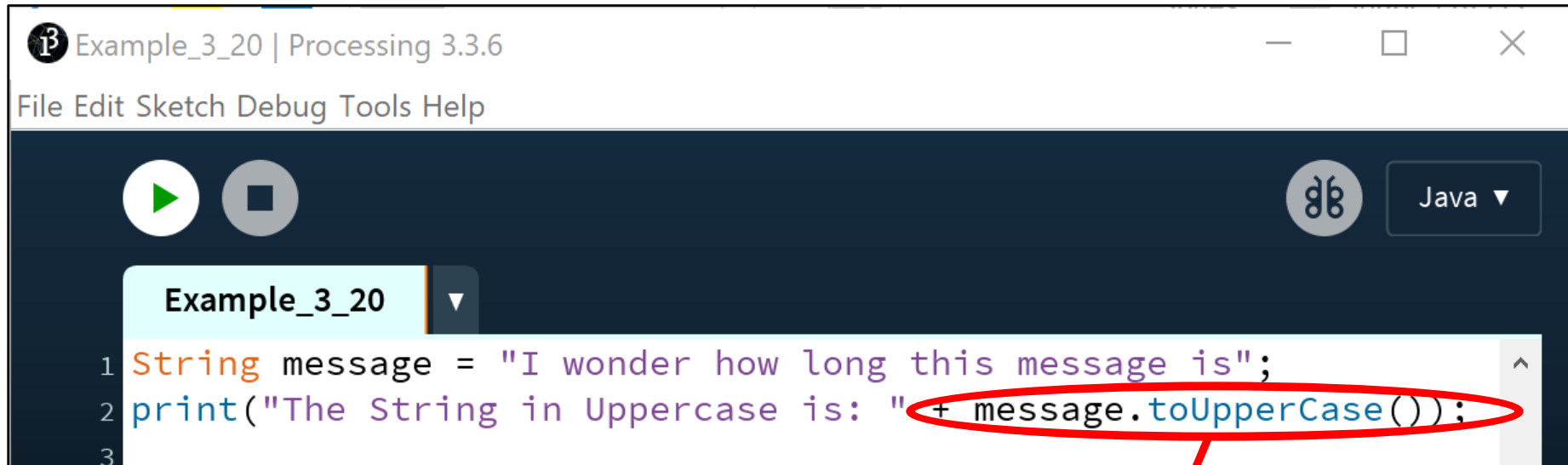
```
It is 33 characters long

Console Errors
```

Printing the length of a String to the console.

Example 3.20

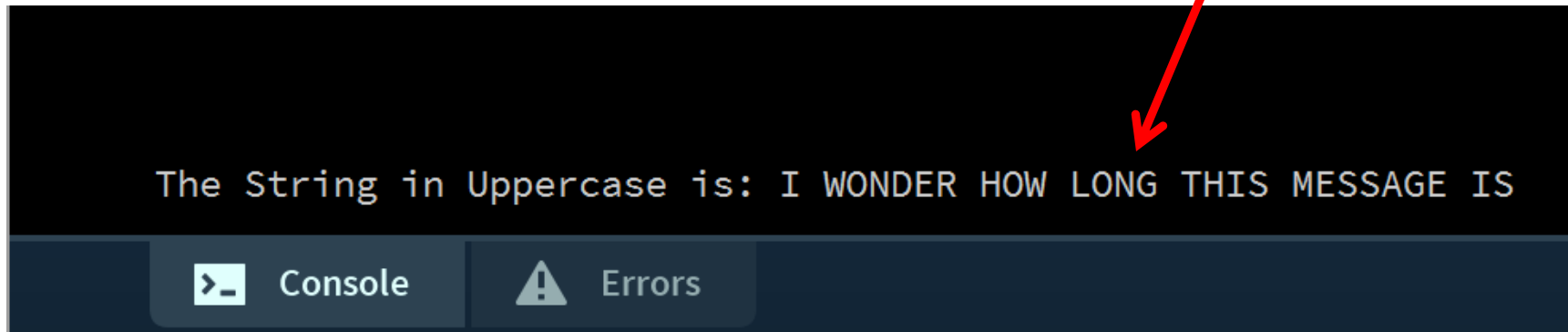
Converting a String to UPPERCASE and printing it to the console.



The screenshot shows the Processing IDE interface. The title bar reads "Example_3_20 | Processing 3.3.6". The menu bar includes "File Edit Sketch Debug Tools Help". The toolbar contains a play button, a stop button, a gear icon, and a language dropdown set to "Java". The code editor shows the following code:

```
1 String message = "I wonder how long this message is";  
2 print("The String in Uppercase is: " + message.toUpperCase());  
3
```

The `message.toUpperCase()` method call in line 2 is circled in red. A red arrow points from this circle to the console output below.



The console output displays the result of the code execution:

```
The String in Uppercase is: I WONDER HOW LONG THIS MESSAGE IS
```

The console interface includes a "Console" tab with a terminal icon and an "Errors" tab with a warning icon.

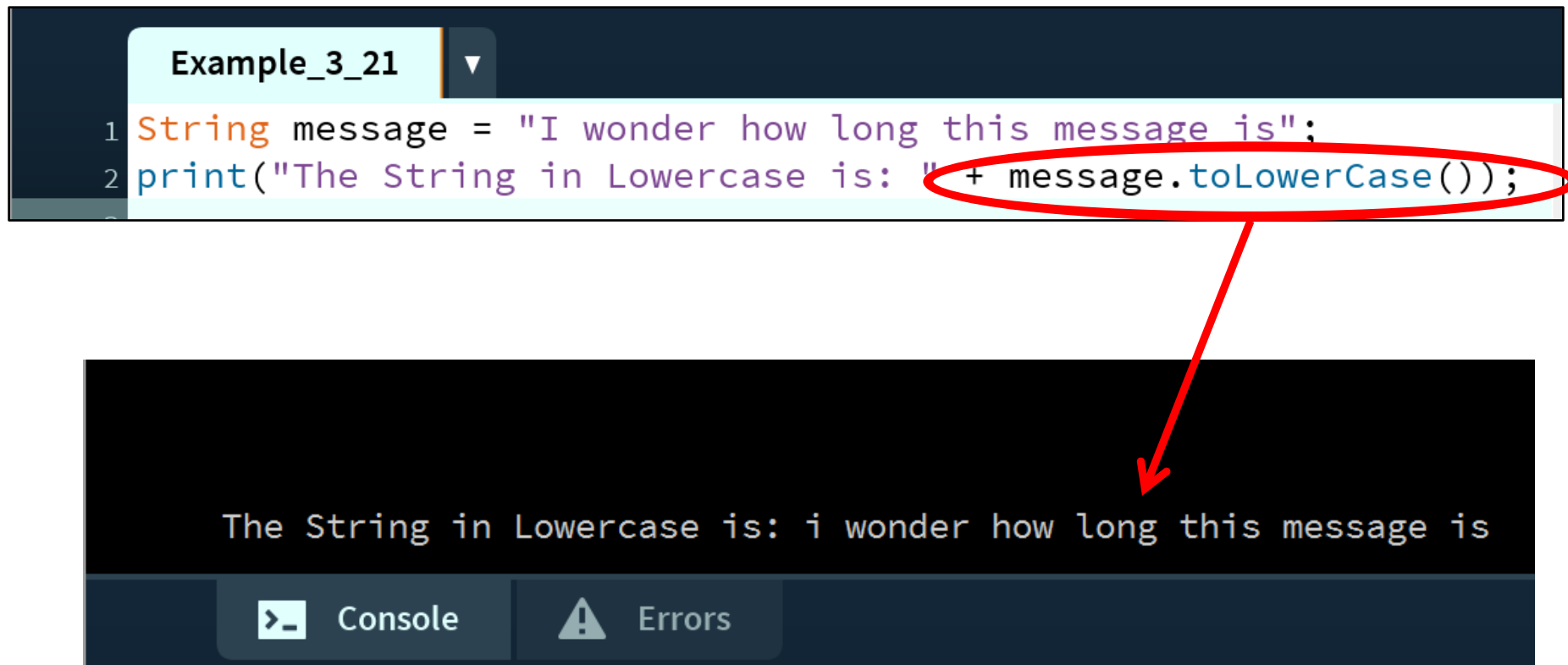
Example 3.21

Converting a String to lowercase and printing it to the console.

```
Example_3_21
1 String message = "I wonder how long this message is";
2 print("The String in Lowercase is: " + message.toLowerCase());
```

The String in Lowercase is: i wonder how long this message is

Console Errors



Example 3.22

Removing all the leading and trailing spaces in a String and printing it to the console.

```
Example_3_22
3
4 String trimmedMessage = message.trim();
5 int trimmedLengthOfMsg = trimmedMessage.length();
6
7 println("The original message " + message
8     + " is " + originalLengthOfMsg + " characters long");
9
10 println("The trimmed message " + trimmedMessage
11     + " is " + trimmedLengthOfMsg + " characters long");
12
```

```
The original message HTTP 404 Not Found Error is 33 characters long
The trimmed message HTTP 404 Not Found Error is 24 characters long
```

Console Errors

SUMMARY - **Strings**

1. Primitive Types: **char**
2. Object Types: **String**
3. **Primitive** Types **versus Object** Types
4. Strings and **Java API**
5. Strings - **methods**
6. **Method calls**
 - **Internal**
 - **External**
 - **Dot notation**
7. Using String methods: some **examples**

Questions?



References

- Reas, C. & Fry, B. (2014) Processing – A Programming Handbook for Visual Designers and Artists, 2nd Edition, MIT Press, London.