|  |
| --- |
|  |

|  |
| --- |
| **QA Consulting** |
| Java Logic |
| Exercise Guide |

# Exercise 1 – HelloWorld

Create:

* Create new a new Maven Project: **HelloWorld**

Code a project that:

* Outputs “HelloWorld” to the console.
* Return “HelloWorld” as a String.
* Has a method that takes a String parameter and outputs this to the console.

# Exercise 2 – Calculator

Create:

* Create new a new Maven Project: **Calculator**

Code a project that:

* Has a method that takes 2 Integer Inputs, adds them together, and returns the result.
* Create 3 more methods that do the same but with multiplication, division and subtraction.

# Exercise 3 – CheckString

Create:

* Create new a new Maven Project: **CheckString**

Code a project that:

* Has a method that when given a String, checks that it matches a hard coded String.
* Has a method that when given 2 Strings, compares the Strings and checks if they are equal.

# Exercise 4 – Calculator Continued.

Return to your Calculator project you created earlier.

Code a project that:

* Combines the 4 methods from earlier into one method that takes 3 parameters, the third parameter represents which arithmetic operation you need to perform.

# Exercise 5 – Print Numbers

Create:

* Create new a new Maven Project: **Iteration**

Code a project that:

* Has a method that prints out the numbers between 1 and 10
* Has a method that prints out the numbers between 100 and 999
* Has a method that prints out the numbers between 1 and 10 in words.
* As above but the numbers 1 and 100.
* As above but the numbers 1 and 1000.

# Exercise 6 – Collections

Create:

* Create new a new Maven Project: **Collections**

Code a project that:

* Has a method that iterates through the numbers 1 to 100 and adds them into an Array.
* Has a method that iterates through the number 100 to 999 and adds them into a List.
* Has a method that iterates through your List from above and multiply each number by 10.

# Exercise 7 – Streams

Create:

* Create new a new Maven Project: **Streams**

Code a project that:

Has a method that completes Exercise 6, but using Streams.

# Exercise 8 – HelloWorld with Lambdas

Return to your HelloWorld Project you created earlier.

Code a project that:

* Achieved the final functionality of Exercise 1, but using the Consumer Interface and Lambdas.

# Exercise 9 – CheckString with Lambdas

Return to your CheckString Project you created earlier.

Code a project that:

* Achieved the final functionality of Exercise 3, but using the BiPredicate Interface and Lambdas.

# Exercise 10 – Calculator with Lambdas

Return to your Calculator Project you created earlier.

Code a project that:

* Achieved the final functionality of Exercise 4, but using the Bi Function Interface and Lambdas.