# Week 06: Coursework

## Spring Framework

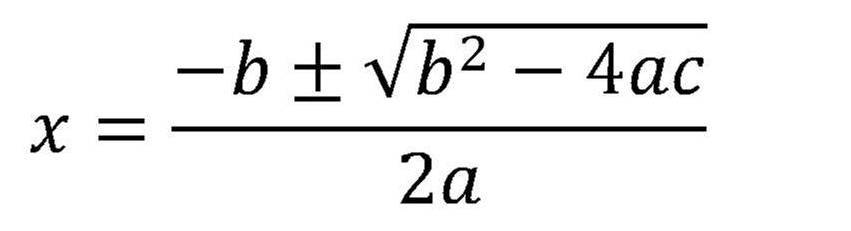
1. Fork and clone the following Github repository.

<https://github.com/busyQA-java-developer/busyqa-06-spring-2-coursework-initial.git>

1. Import the Maven project into Eclipse.
2. This project is a small web application that implements a Quadratic Equation Calculator.

**This Calculator finds the solution of equations in the form:**

**ax² + bx + c = 0**

**where x :**

We need to implement the logic for the formula stated above.

This project uses a Thymeleaf template named *calculator.html* that requires NO modifications unless you want to customize it. The element you need to modify is the Controller only. The five steps below describe the steps to follow.

* 1. Annotate the class *CalculatorController* with:

@Controller

* 1. Annotate the method *show()* with:

@GetMapping("/")

The string "/" passed as a parameter to the @GetMapping annotation tells Spring MVC that this method takes no parameters and returns this website's main page.

* 1. Annotate the method "solve()" with:

@PostMapping("/{parameter\_name}")

The string *"/{parameter\_name}"* passed as a parameter to the @PostMapping annotation has to be modified. The portion with curly braces *"/{parameter\_name}"* must be replaced with the action value used in the *calculator.html* file.

Hint: Open the *calculator.html* file in the /*resources/templates* folder and look for the “*th:action”* attribute. Copy the value associated with this attribute and replace the "*{parameter\_name}*" indicated above with it (including the curly braces).

* 1. The *solve() method* has three parameters annotated with @RequestParam. Replace the "{tag\_?}" passed to the annotations with the proper value.

The parameters are as follows:

@RequestParam("{tag\_a}") double a,

@RequestParam("{tag\_b}") double b,

@RequestParam("{tag\_c}") double c

Hint: Go to the *calculator.html* file inside the */resources/templates* folder and look for the *"input type=number"* elements. Each element has a *name* attribute. Copy the value associated with the *name* attribute and replace it in the "solve()() method. Then, do the same with the other two elements.

For instance:

You find the following *input type=number* element in the *calculator.html* template.

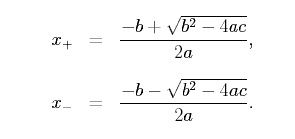
<input type="number" name="param\_a">

Here you see that the attribute’s *name* value equals *param\_a*. So what you need to do is to replace in the @RequestParam annotated parameter the "{tag\_a}" with "*param\_a*" (Note that we replace the curly braces are as well)

The *solve()* method can read the value from the input type=number element of the *calculator.html* template. Repeat this for the other two elements.

* 1. The solve() method implements the business logic to solve the quadratic equation using the three parameters explained in the step above.

A quadratic equation has two solutions or roots. We can find each root with the following formulas:



Before performing any calculation, ensure that the expression inside the square root is greater or equal to zero; otherwise, you would get an Exception.

So make sure that:

(b)² - 4(a)(c) >= 0

If the expression above is less than zero, assign 0 to both root variables and add a variable called “*mssge*” to the model with the string “Equation has no real roots!” otherwise, calculate the two roots (root1 and root2) and add the variable “mssge” with the string "Calculation completed successfully!".

The Java Math library contains practical methods that will help you with the calculations.

1. After completing the project, test it, then commit and push your changes to your GitHub repository.
2. You can find the solution for the project in the Github repo:

<https://github.com/busyQA-java-developer/busyqa-06-spring-3-coursework-complete.git>

// End