

Steffano Martinez

GitHub Username: SteffanoMartinez

COP4808 Full Stack Web Development

January 31st, 2023

GitHub Repo: [Repo](#)

HTML Quiz :

HTML Quiz

Result:

38 of 40

95%


You can be proud of yourself!

Time Spent
14:24

[Check your answers](#) [Try Again](#) [Back to Quizzes](#)

Share your score:

[f](#) [t](#) [in](#)



CSS Quiz:

CSS Quiz

Result:

23 of 25

92%


You can be proud of yourself!

Time Spent
22:47

[Check your answers](#) [Try Again](#) [Back to Quizzes](#)

Share your score:

[f](#) [t](#) [in](#)



JavaScript Quiz :

JavaScript Quiz

Result:

24 of 25

96%


You can be proud of yourself!

Time Spent
18:21

[Check your answers](#) [Try Again](#) [Back to Quizzes](#)

Share your score:

[f](#) [t](#) [in](#)



jQuery Quiz:

jQuery Quiz

Result:

23 of 25

92%


You can be proud of yourself!

Time Spent
17:33

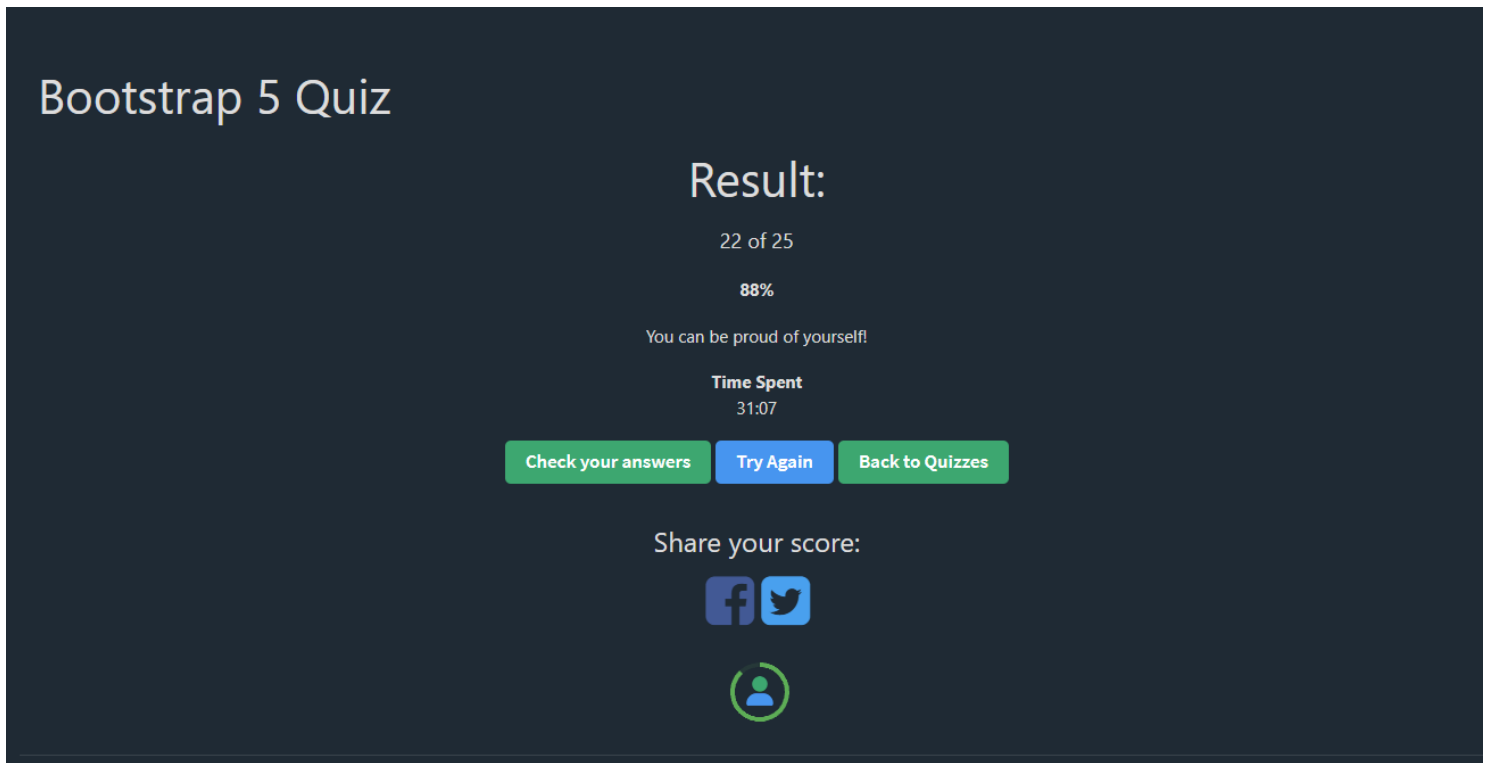
[Check your answers](#) [Try Again](#) [Back to Quizzes](#)

Share your score:

[f](#) [t](#) [in](#)



Bootstrap Quiz:



Objective 2:

As per the assignments, we will modify the provided calculator HTML/CSS/JS via the presented requirements, to be able to achieve multiple commits we will commit each time we complete one of the 4 requirements, however extra commits might exist for formatting or last minute debugging/added requirements.

Let's begin with the following requirement:

Add a new row of buttons at the top of the calculator - making it the first row of buttons. It must be below the number output and above the "AC, +/-, %, /" buttons.

Include screenshot showing you adding the row and the empty buttons in the writeup document (10 points)

First within version control I make sure that I do not have any pending changes or commits that need to be pushed, to do so I go into my repo directory locally through Git and type the command `git status`, when I get the message nothing to commit, working tree clean I am ready to begin working within the master branch as no external branch is necessary for this assignment. This is shown in Figure 1

```
MINGW64:/c/Users/steff/Documents/Spring 2023/Full Stack Web Developm...
$ cd Homework\ 1/

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack W
eb Development/Repos/Homework 1
$ ls
cop4808-git-and-github-fundamentals-SteffanoMartinez/

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack W
eb Development/Repos/Homework 1
$ cd cop4808-git-and-github-fundamentals-SteffanoMartinez/

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack W
eb Development/Repos/Homework 1/cop4808-git-and-github-fundamentals-SteffanoMart
inez (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack W
eb Development/Repos/Homework 1/cop4808-git-and-github-fundamentals-SteffanoMart
inez (main)
$ .....
```

Figure 1: Clean main branch, unwanted commits

Since we are adding new buttons, let's look at how the structure of the buttons look with currently existing ones. In the **calculator.html** which is the index.html file in this case we can see that within the <body> tags of the HTML we have included within a <div> with attribute container (as per BS5), and within that container we have all the content of the calculator, as the body. We can see that there about 18 <button> tags, they all contain the same attributes with different values, the values differ to their intended use, we also see the use of a data-key attribute, this is linked to an event listener to enable the input from your keyboard as keystrokes have specified values (ASCII related) and trigger an input into the calculator, since we are not adding any input but instead adding operations we will ignore this specific attribute due to the 95th of keyboards not having a designated button for the 4 new functions. The new buttons must be added as the first 4 lines of code within the div "Display" to maintain the top row. I've labeled them as "NEW" to note this within the HTML as well as comments.

```
<!-- This are my new added buttons-->
<button type="button" class="equals" value="=" data-key="69">NEW</button>
<button type="button" class="equals" value="=" data-key="69">NEW</button>
<button type="button" class="equals" value="=" data-key="69">NEW</button>
<button type="button" class="equals" value="=" data-key="69">NEW</button>
<!-- ##### -->
```

Let's look at the change live in our page:



We see that there are 4 new buttons added on the top row as per the requirement, however, we now introduced a new bug, the bottom three buttons are squashed! As per the standard among many developers we will put this bug in a ticket, will be dealt after the new functionality is introduced and working, this needs to be noted in the first commit after finishing the requirement to alert other developers in the case of a company. We will now commit this as 1/4th of our job done.

Below is attached the previous commit process, shown in Figure 2:

```

eb Development/Repos/Homework 1/cop4808-git-and-github-fundamentals-SteffanoMart
inez (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   calculator.html
        deleted:    scripts/script.js.download

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        scripts/script.js

no changes added to commit (use "git add" and/or "git commit -a")

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack W
eb Development/Repos/Homework 1/cop4808-git-and-github-fundamentals-SteffanoMart
inez (main)
$ git add --all
warning: in the working copy of 'calculator.html', LF will be replaced by CRLF t
he next time Git touches it
warning: in the working copy of 'scripts/script.js', LF will be replaced by CRLF
the next time Git touches it

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack W
eb Development/Repos/Homework 1/cop4808-git-and-github-fundamentals-SteffanoMart
inez (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   calculator.html
        renamed:    scripts/script.js.download -> scripts/script.js

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack W
eb Development/Repos/Homework 1/cop4808-git-and-github-fundamentals-SteffanoMartinez (main)
$ git commit -m "Added 4 new buttons to the calculator | WARNING: BUG introduced, the last row of buttons
are squashed, this must be revisited before final revision"
[main fac3b0f] Added 4 new buttons to the calculator | WARNING: BUG introduced, the last row of buttons ar
e squashed, this must be revisited before final revision
 2 files changed, 7 insertions(+), 1 deletion(-)
 rename scripts/{script.js.download => script.js} (100%)

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack Web Development/Repos/Homew
ork 1/cop4808-git-and-github-fundamentals-SteffanoMartinez (main)
$ git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 623 bytes | 623.00 KiB/s, done.
Total 4 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/cop4808-spring-2023-fullstack-web/cop4808-git-and-github-fundamentals-SteffanoMartin
ez.git
   00e0caf..fac3b0f  main -> main

```

Figure 2: First commit to project related actions with bug notice.

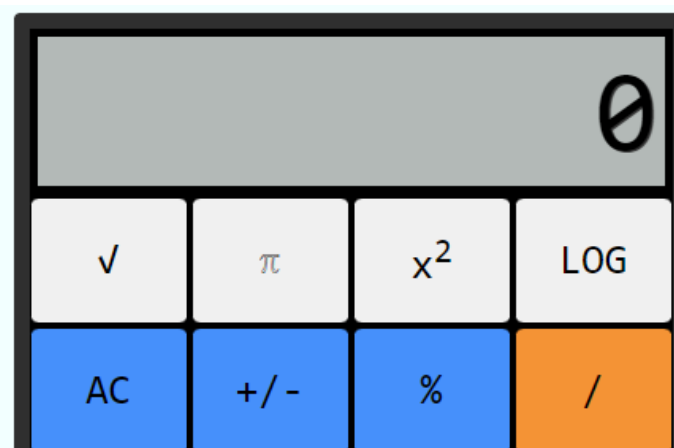
Lets move into our second requirement as per the assignment which is:

Create a new CSS class with a different background color for these new buttons and buttons should be the same size as the others and the text should be bold on the buttons (15)

For this requirement it is asking us to change something visible but not component wise, meaning that we will not investigate the HTML for very long, it also does not mention new functionality, so we will skip the JS file, meaning that we need to add/modify the CSS that commands the style of the added buttons. For this we need to understand how are the previous buttons styled with different colors depending on their input type or operator, operand, type of buttons. This is done via the class attribute, which is defined to have specific parameters within the CSS, meaning that all we need to do, since our functionalities are going to be new, is to add a new class attribute to be able to control our new button colors, as well as the bolded text. The new class for our buttons will be named “**extended-operator**”, adding functionality to existing code must have harmony within the intended added functionality in order to allow future developers or your future self to quickly understand what the context is, even if the comments are clearly stating something, is always good to keep a clear naming convention. Since we are talking about the inner descriptions of the new buttons we will also change the value attribute to the new operators, the data-key can be exchanged by id as a simpler attribute, its important to note that I’m not linking any keys in your keyboard to these functionalities since it’s not common, examples such as DESMOS and multiple online calculators can read input from keyboard but you cannot press “k” and obtain the SIN() function. I added the button content according to my four functions, this was found in the web as I did not know then, for the square button I had to be creative, and used the <sup> tag that is a superscript in order to get the two to show, writing exp is too simple, maybe a lot more useful than x^2, but it ties with my sqrt function. This is what I’ve changed so far in the HTML.

```
<!-- This are my new added buttons-->
<button type="button" class="extended-operator" id="sqrt">&radic;</button>
<button type="button" class="extended-operator" value="3.14" id="pi">&#8508;</button>
<button type="button" class="extended-operator" id="square">x<sup>2</sup></button>
<button type="button" class="extended-operator" id="log">LOG</button>
<!-- ##### -->
```

As you can see all of my functions, there is only one that remains with the value attribute, and that is PI, since I will intend to use that whenever its called in the logic part, later asked. For now this is how they are displayed in the website:



Now let's move into the requirement of making them their own background color and having them in bold. For this we move into CSS stylesheets file and look at how the previous operand buttons were styled. They are just called by their class and under the declaration block are attributed the CSS declaration background-color. Lets do the same for our new added buttons, I want to make them green but like a nice green, this is how I decided to make it:

```
.extended-operator {  
  background-color: ■ rgb(46, 160, 118);  
  font-weight: bold;  
}
```

This added the boldness and the color, use the rgb function for me to be able to choose from a UI graph. Here is how my current calculator looks like:



I believe this meets the requirements mentioned in this second part, now its time to commit and to push, always remembering that the bug still exists, this process is shown in Figure 3:


```

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack Web Development/Repos/Homew
ork 1/cop4808-git-and-github-fundamentals-SteffanoMartinez (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack Web Development/Repos/Homew
ork 1/cop4808-git-and-github-fundamentals-SteffanoMartinez (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   calculator.html
        modified:   styles/style.css

no changes added to commit (use "git add" and/or "git commit -a")

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack Web Development/Repos/Homework 1/cop4808-git-and-github-fundamentals-SteffanoMartinez (main)
$ git add --all
warning: in the working copy of 'calculator.html', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'styles/style.css', LF will be replaced by CRLF the next time Git touches it

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack Web Development/Repos/Homework 1/cop4808-git-and-github-fundamentals-SteffanoMartinez (main)
$ git commit -m "Changed the buttons background and added the defined functionality with the bold | WARNING: The bottom buttons are still squashed, this bug needs to be resolved before final release"
[main 8c026e6] Changed the buttons background and added the defined functionality with the bold | WARNING: The bottom buttons are still squashed, this bug needs to be resolved before final release
 2 files changed, 9 insertions(+), 4 deletions(-)

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack Web Development/Repos/Homework 1/cop4808-git-and-github-fundamentals-SteffanoMartinez (main)
$ git push
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 12 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 682 bytes | 682.00 KiB/s, done.
Total 5 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/cop4808-spring-2023-Fullstack-web/cop4808-git-and-github-fundamentals-SteffanoMartinez.git
 fac3b0f..8c026e6  main -> main

```

Figure 3: Second requirement commit, and push made.

Let's now begin the third requirement:

Implement 4 new functions - one for each button and implement them (25)

For this one, we now must look into the JS file, we can begin looking at how the other calculator operators work to understand the flow, as well as how the screen gets updated, and how it can refresh. We can see that the buttons are all selected from the HTML File, and they are parsed through a for loop only cycling through the class value, in this case it will hit the first IF statement where it looks if the button contains the **'operator'** value, here we need to add our new class of buttons to be added into the list to be able to be listened and its input taken care of. IF it does then it will send the value of the operator through the function called **inputOperator** which takes in the current button iteration within the "i" index. Our addition is shown in the following image:

```

        inputOperator(buttons[i].value);
        // Additional query added to button array //
    } else if(buttons[i].classList.contains('extended-operator')){
        if(buttons[i].classList.contains('pi')){
            inputOperator(buttons[i].value);
        }
        inputOperator(buttons[i].id);
        // ##### //
    } else if(buttons[i].classList.contains('equals')) {
        inputEquals();
    }

```

Its important to understand that since in the HTML the pi button is really a value pair compared to the rest of my buttons with the id pair. I could have also changed them all to value pairs but it doesn't break functionality, keep in mind that the π function works as a constant as it not really an operator.

Upon reaching **inputOperator** it now looks for what position the operand and operator are at, this is still not too important as it doesn't seem to introduce the type of operator, however we can see that the result in any of the arguments for operand and operator positions stores the value of the operation in a global variable called **result**. The result variables get value from a new function called **operate** which takes in the firstOperand and typecasts it as a number, the second operand as well typecast and also the first operator, or second operator, depending on the if statement executed. Within the operate function we can see that the actual logic is implemented for the pre-existing operators, in this case, we will introduce our new four operators, within the function we have x representing the first operand, y representing the second operand, and op representing the actual operator, these are all local variables and will be only local to the function except returning the operation within the return statement.

```
// Added functionality //
} else if (op == 'sqrt') {
    return Math.sqrt(x);
} else if (op == 'pi'){
    return 3.14;
} else if (op == 'square') {
    return x * x;
} else if (op == 'log'){
    return Math.log10(x);
}
}
```

Steffano Martinez - Z#23516915



From here all the functions work, since they are single operand they work with input of the operand, operator, and then equals will display the value. This would conclude the third requirement with the new functionality added. Now our next step is to commit this and move into the next requirement, always pointing out that the bug still exists. (As shown in Figure 4)

```
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   scripts/script.js

no changes added to commit (use "git add" and/or "git commit -a")

stefff@Steffano-Desktop MINGW64 /c/Users/stefff/Documents/Spring 2023/Full Stack W
eb Development/Repos/Homework 1/cop4808-git-and-github-fundamentals-SteffanoMart
inez (main)
$ git add --all
warning: in the working copy of 'scripts/script.js', LF will be replaced by CRLF
the next time Git touches it

stefff@Steffano-Desktop MINGW64 /c/Users/stefff/Documents/Spring 2023/Full Stack W
eb Development/Repos/Homework 1/cop4808-git-and-github-fundamentals-SteffanoMart
inez (main)
$ git commit -m "Added functionality to the buttons | WARNING: BUG is still pres
ent on the bottom buttons, they need to be fixed before final release"
[main 8da71cc] Added functionality to the buttons | WARNING: BUG is still presen
t on the bottom buttons, they need to be fixed before final release
1 file changed, 16 insertions(+)

stefff@Steffano-Desktop MINGW64 /c/Users/stefff/Documents/Spring 2023/Full Stack W
eb Development/Repos/Homework 1/cop4808-git-and-github-fundamentals-SteffanoMart
inez (main)
$ git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 670 bytes | 670.00 KiB/s, done.
Total 4 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/cop4808-spring-2023-fullstack-web/cop4808-git-and-github-f
undamentals-SteffanoMartinez.git
8c026e6..8da71cc main -> main
```

Figure 4: Third requirement met, commit done with bug still included.

Now we investigate the fourth requirement:

Output result must stay within the digital readout. (10 points)

The issue with this requirement that I've found is that I'm unable to get the output numbers to extend past the display, this is due to implementation of the functions within pre-existing formatted functions that take care of the values to not extend over the display. Since I'm unable to replicate this issue I will now take into account the second visual requirement and its that bug, lets visualize again the HTML file that house most of our calculator requirements. We can see that everything is house in the <div> with container id, we must look into this id within the CSS to find out the dimensions for the calculator as new buttons have been added.

After some revision it seems that the attribute grid-template-rows needed to be expanded in the repetition aspect to add one more row, so the change was made, and the bottom buttons now fit appropriately while maintaining functionality. Overall all four requirements have been met and the calculator is functioning properly. Figure 5 shows the last commit:

```
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   styles/style.css

no changes added to commit (use "git add" and/or "git commit -a")

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack Web Development/Repos/Homework 1/cop4808-git-and-github-fundamentals-SteffanoMartinez (main)
$ git add --all
warning: in the working copy of 'styles/style.css', LF will be replaced by CRLF the next time Git touches it

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack Web Development/Repos/Homework 1/cop4808-git-and-github-fundamentals-SteffanoMartinez (main)
$ git commit -m "Fixed visual bugs as well as tested for the fourth requirement, did not founda any digits going out of the display"
[main 6350d09] Fixed visual bugs as well as tested for the fourth requirement, did not founda any digits going out of the display
1 file changed, 1 insertion(+), 1 deletion(-)

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack Web Development/Repos/Homework 1/cop4808-git-and-github-fundamentals-SteffanoMartinez (main)
$ git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 410 bytes | 410.00 KiB/s, done.
Total 4 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/cop4808-spring-2023-fullstack-web/cop4808-git-and-github-fundamentals-SteffanoMartinez.git
8da71cc..6350d09 main -> main
```

Figure 5: Final commit in regards to the changes proposed as well as bug fixes.

Now as part of the assignment it is also necessary to write and document what it was done in the readme-file, so that shall be done now with an added gif or image of the gif showing the new functions properly working. Here is the commit for that regard:

```
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   GIFS/LogarithmFunction.gif
        new file:   GIFS/PIFunction.gif
        new file:   GIFS/SquareFunction.gif
        new file:   GIFS/SquareRootFunction.gif
        modified:   README.md

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack Web Development/Repos/Homework 1/cop4808
$ git commit -m "Added GIFS, working on README File"
[main beca660] Added GIFS, working on README File
5 files changed, 2 insertions(+), 106 deletions(-)
create mode 100644 GIFS/LogarithmFunction.gif
create mode 100644 GIFS/PIFunction.gif
create mode 100644 GIFS/SquareFunction.gif
create mode 100644 GIFS/SquareRootFunction.gif

steff@Steffano-Desktop MINGW64 /c/Users/steff/Documents/Spring 2023/Full Stack Web Development/Repos/Homework 1/cop4808
$ git push
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 12 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (8/8), 154.07 KiB | 19.26 MiB/s, done.
Total 8 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/cop4808-spring-2023-fullstack-web/cop4808-git-and-github-fundamentals-SteffanoMartinez.git
6350d09..beca660 main -> main
```