

Function Exercises

Exercise 6 – Declaring and using functions

Create a welcome message function called `WelcomeMessage()`. When called it will display the program's welcome message. This will call several `Console.WriteLine()` functions to print an opening message. E.g.:

```
*****
```

```
Welcome to My Application!
```

```
*****
```

Exercise 7 – Parameters

Create a mathematical function that takes two parameters that are numeric types and performs a basic arithmetic function on them and prints the result to the console. Try addition and multiplication.

Exercise 8a – Simple Name

Update the welcome message function from exercise 6 to include a parameter that allows you to pass the name of the user into it and print that along with the function.

```
*****
```

```
Welcome <Your Name>, to My Application!
```

```
*****
```

Exercise 8b – Complex Info

Recreate the user information exercise (1 and 2) but this time create a function that takes those inputs as parameters. Call the function and pass different values.

Example output:

I am older than them.

The age difference is 5 Years.

They have a longer name than I do.

Exercise 9 – String check for int

Create a function that takes two parameters, one that is a string, and the other is an int. Check if the string contains the value of the int anywhere in it. *Note: There are a few ways to achieve this exercise. One uses string contains, the other can be to loop and check each character. What are the drawbacks of each?*

Exercise 10 – Return

Update your exercise 7 functions to return values and remove the print statement from the function itself. Instead, the function will calculate and return the value only. Use the function inside a `Console.WriteLine()` to then print the result like this:

```
Console.WriteLine(Add(4,7));
```

The output should look like this:

`"4 x 10 = 40" or "5 + 6 = 11"`