

# "Custom Functions" Exercise

---

Prerequisites: [Functions](#)

```
-----  
CUSTOM FUNCTIONS EXERCISE...  
-----  
THE CELSIUS TEMP IS: 0 DEGREES  
THE FAHRENHEIT EQUIVALENT IS: 32.0 DEGREES  
-----  
THE NUMERIC SCORE IS: 87.5  
THE LETTER-GRADE EQUIVALENT IS: B+
```

## Learning Objectives

1. Find practical applications for learning new programming concepts like functions and parameters.

## Setup

To setup these exercises, create a new directory on your Desktop called "custom-functions", and navigate there from the command line:

```
cd ~/Desktop/custom-functions
```

Open the directory with your text editor, and use your text editor or the command-line to create a new file in it called "my\_functions.py", then place the following contents inside:

```
# custom-functions/my_functions.py  
  
# TODO: define temperature conversion function here  
  
# TODO: define gradebook function here  
  
if __name__ == "__main__":  
    print("-----")  
    print("CUSTOM FUNCTIONS EXERCISE...")  
  
    #print("-----")  
    #c = 0  
    #print("THE CELSIUS TEMP IS:", c, "DEGREES")
```

```
#f = celsius_to_fahrenheit(c)
#print("THE FAHRENHEIT EQUIVALENT IS:", f, "DEGREES")

#print("-----")
#score = 87.5
#print("THE NUMERIC SCORE IS:", score)
#grade = numeric_to_letter_grade(score)
#print("THE LETTER-GRADE EQUIVALENT IS:", grade)
```

Activate any virtual environment where you'll have access to the `python` command-line utility (the "base" Anaconda environment should be fine for this exercise).

From within the virtual environment, run the exercise script:

```
python my_functions.py
```

If you see your printed message, you're ready to tackle the basic challenges below.

## Basic Challenges

### Temperature Converter

Near the top of the script, define a custom function called `celsius_to_fahrenheit()`, the responsibility of which is to convert a Celsius temperature (e.g. `0`) to a Fahrenheit temperature (e.g. `32`). Inside the function's definition, use a mathematical formula to convert the temperature and return the resulting value.

Near the bottom of the script, uncomment the lines of code relating to this function, and run the program. The program should invoke your custom function to perform the example conversion and print the corresponding Fahrenheit temperature.

### Gradebook

Near the top of the script, define a custom function called `number_to_letter_grade()`, the responsibility of which is to convert a numeric grade (e.g. `87.5`) to a corresponding letter grade (e.g. `B+`). Inside the function's definition, use your own custom algorithm to convert the grade and return the resulting value.

Near the bottom of the script, uncomment the lines of code relating to this function, and run the program. The program should invoke your custom function to perform the example conversion and print the corresponding letter grade.

## Further Exploration Challenges

HINT: use the `input()` function to capture user inputs

## User Inputs Temperature

The program should ask the user to input a Celsius temperature via the command-line, and pass that value into the custom function instead of the hard-coded `0`.

## User Inputs Grade

The program should ask the user to input a numeric grade via the command-line, and pass that value into the custom function instead of the hard-coded `87.5`

## Importing Functions from Other Files

Create another file in this exercise directory called "my\_script.py" and place inside the following contents:

```
# custom-functions/my_script.py

from my_functions import celsius_to_fahrenheit

if __name__ == "__main__":

    f = celsius_to_fahrenheit(0)
    print("F TEMP IS:", f)
```

Run this script:

```
python my_functions.py
```

It should work, as long as you kept the `if __name__ == "__main__"` conditional in the "my\_functions.py" file.

Nice job! You've demonstrated how to import a function from one file to be used in another. This is basically what happens when we import functions from built-in modules and third-party packages. It's just that the code we import from those modules and packages has been defined for us, whereas in this exercise we have defined our own custom functions.