CMS Office Hours
Project 1: Create an Integration Code **2020-Aug-10**

Use Python to develop a script which integrates a User-defined 1-dimensional function ($f(x)$) over User-specified integration bounds.

Example:

```
user_defined_function = 2 * sin(x)
your_integrator(user_defined_function, 0, pi)
# returns: 4.0
```

Here is generic way to structure your code:

```
"""
This code returns the value of integration when the User
specifies a function and the integration bounds.

Example:
    user_defined_function = 2 * sin(x)
    your_integrator(user_defined_function, 0, np.pi)  # (fn, x_min, x_max)
    # returns: 4.0
"""
import numpy as np

#--- User-defined/Global variables.
x_min = 0.0
x_max = 2.0
fn = 2 * np.sin(x)

#--- Script functions.
def integrate(fn, x_min, x_max):
    """Take in a function (fn), x_min, x_max and return the value of integration."""
    # Your code goes here.

def calc_area_of_rect(width, height):
    """Calculate the area of a rectangle."""
    return width * height

# You will definitely need more functions than the two above.

#--- Instead, you can take an object-oriented approach (using classes).
# class Integration():
#     def __init__():
#         # Lots of juicy code goes below.

#--- When you do: python this_script.py, then this code will be run over:
if __name__ == "__main__":
    # This is the logical, sequential call to the main functions.
    # Just a few simple function calls.
    integrate(fn, x_min, x_max)
```