

Programming Assignment #4

[Submit Assignment](#)

Due Oct 21 by 11:59pm **Points** 10 **Submitting** a file upload

Available Oct 9 at 12am - Oct 23 at 11:59pm 15 days

Your task is to build a WebAPI calculator. The requirements are as follows:

1. The API should have two different controller classes: Basic and Advanced.
2. The Basic controller should provide the following functionality:
 1. Addition(x, y)
 2. Subtraction(x, y)
 3. Multiplication(x, y)
 4. Division(x, y)
 5. Sqrt(x)
 6. Power(Base, Exponent)
 7. Exponential(Power): e^{Power}
 8. Log(base, parameter) (Hint: remember change of base formula, or find a library)
3. The Advanced controller should provide the following functionality:
 1. QuadraticEquation(a,b,c): find the roots of $f(x) = ax^2 + bx + c$ using quadratic equation
 2. Vertex(a,b): return the vertex of the parabola $f(x) = ax^2 + bx + c$
 3. Slope(x1,y1,x2,y2): return the slope given two points (x1,y1) and (x2,y2)
 4. Random(min, max): return a random integer between min and max
 5. AreaTriangle(base, height): return the area of a triangle with the provided base and height
 6. AreaRectangle(length, width): return the area of a rectangle with the provided length and width
 7. AreaCircle(radius): return the area of a circle with the provided radius
 8. Fibonacci(n): return the nth fibonacci number
4. Each method should have a POST method and a GET method
 1. The GET method should pass all parameters in the route of the API call
 2. The POST method should pass all parameters as a request class in the body of the API call
5. There are no requirements about the shape of your request class
6. You should have a separate client project that is just a console application that provides the user access to each of those methods in any way you see fit with the caveat that the user must have access to all 32 methods through the client application.
 1. One way is to simply provide a number-based menu like we did for programming assignment 2, but split the functionality first into Advanced/Basic, then into GET/POST, and finally into the options a-h for each controller.