CS101. A3 - nth root

Write a function nth_root to accept 2 arguments: number and n. The function should return the nth root of the number up to two decimal places only.

Code Template

```
"""
nth root of a number
"""
def nth_root(number, n):
     pass
```

Test Cases

Function	Inputs	Output
`nth_root`	8, 3	2.0
`nth_root`	2, 2	1.41
`nth_root`	60, 10	1.51
`nth_root`	100, -1	`raise ValueError("Negative roots can't be calculated")`

***

CS101. A4 - Factor

Write a function check_factor to accept 2 integer input arguments: number and n. The function should return True if n is a factor of number. Otherwise the function returns False.

Code Template

```
"""
n is a factor of number?
"""
def check_factor(number, n):
     pass
```

Test Cases

Function	Inputs	Output
`check_factor`	8, 3	False
`check_factor`	2, 2	True
`check_factor`	60, 10	True
`check_factor`	100, 2.1	`raise ValueError("n must be an integer")`

****

CS101. A5 - Check Multiple

Write a function check_multiple to accept 1 integer argument: number. The function should return True if the input is a multiple of 7 but not a multiple of 3. Otherwise the function returns False.

Code Template

```
"""
Check multiple
"""
def check_multiple(number):
     pass
```

Test Cases

Function	Inputs	Output
`check_multiple`	77	True
`check_multiple`	42	False
`check_multiple`	0	False
`check_multiple`	2.1	`raise ValueError("input must be an integer")`

****

CS101. A6 - Triangle

Given lengths of 3 sides of a Triangle, identify its type amongst the following.

- ★ For a shape to be a triangle at all, all sides have to be of length > 0, and the sum of the lengths of any two sides must be greater than or equal to the length of the third side.
- ★ An equilateral triangle has all three sides the same length.
- ★ Isosceles triangle if two sides of the triangle are the same. An isosceles triangle has at least two sides the same length.
- ★ A scalene triangle has all sides of different lengths.
- ★ Degenerate triangle: sum of the lengths of two sides equals that of the third. It has zero area and looks like a single line

Write code inside 5 functions (each for a type of a triangle) to ascertain the type. Examples:

- is_equilateral(2, 2, 2) returns True, is_equilateral(2, 3, 4) returns False
- is_isosceles(3, 4, 4) will return True. is_isosceles(4, 4, 4) will return True. isosceles(2, 3, 4) will return False
- is_scalene(5, 4, 6) returns True, is_scalene(7, 3, 2) returns False
- is_degenerate(2,4,6) returns True, is_degenerate(2,4,3) returns False
- More in the Test Cases Table.

Code Template (grains.py)

```
"""
Numbers - Triangle example
```

```
"""
def is_triangle(side1, side2, side3):
      pass

def is_equilateral(side1, side2, side3):
      pass

def is_isosceles(side1, side2, side3):
      pass

def is_scalene(side1, side2, side3):
      pass

def is_degenerate(side1, side2, side3):
      pass
```

Test Cases

Function	Inputs	Output	Remarks
is_triangle	0, 0, 0	False	
is_triangle	10, 0, 0	False	
is_triangle	10, 20, 0	False	
is_triangle	10, 20, 30	True	
is_triangle	10, 1, 1	False	
is_triangle	10, 5, 5	True	
is_triangle	10, 5, 4	False	
is_equilateral	10, 10, 10	True	
is_equilateral	0, 0, 0	False	
is_equilateral	5, 5, 5	True	
is_equilateral	5, 5, 4	False	
is_equilateral	5, 3, 4	False	
is_equilateral	0.5, 0.5, 0.5	True	
is_equilateral	10, 1, 1	False	
is_isosceles	3, 4, 4	True	
is_isosceles	4, 3, 4	True	
is_isosceles	4, 4, 3	True	
is_isosceles	4, 4, 4	True	
is_isosceles	3, 4, 5	False	
is_isosceles	0, 0, 0	False	
is_isosceles	10, 1, 1	False	

is_isosceles	1, 10, 1	False	
is_isosceles	1, 1, 10	False	
is_isosceles	0.5, 0.5, 0.4	True	
is_scalene	5, 4, 6	True	
is_scalene	4, 4, 4	False	
is_scalene	4, 4, 3	False	
is_scalene	10, 1, 1	False	
is_scalene	0.5, 0.4, 0.6	True	
is_degenerate	0, 0, 0	False	
is_degenerate	10, 5, 5	True	
is_degenerate	10, 15, -5	`raise ValueError("Side cannot be negative.")`	
is_triangle	-10, 20, 30		
is_isosceles	-10, 20, 30		
is_scalene	-10, 20, 30		
is_equilateral	-10, 20, 30		

****