Python Exercise 3

Population of Organisms (20 points)

Write a program that predicts the approximate size of a population of organisms.

The program should prompt the user to enter:

- a starting number of organisms
- the average daily population increase (as a percentage)
- the number of days

Include input validation to ensure that the least value entered is at least 1.

The program should use a loop to display the data (neatly) in a table, showing the day and the population for that day. (Note that the population begins to increase after the first day.) Format the population to 6 decimal places.

Sample output

```
Starting number of organisms: -1
Starting number of organisms: 2
Average daily increase (as a %): 0
Average daily increase (as a %): 30
Number of days to multiply: -9
Number of days to multiply: 10
Day                     Population
-------------------------------------------
1                       2.000000
2                       2.600000
3                       3.380000
4                       4.394000
5                       5.712200
6                       7.425860
7                       9.653618
8                       12.549703
9                       16.314614
10                      21.208999
```

Good ol' Guessing Game (25 points)

Write a program that asks the user to guess a number between 1 and 100 (inclusive). The user will then have 10 tries to guess the number.

The program should randomly generate a number within the range and keep track of the number of times the user has guessed. The program should also ensure that every guess is within the correct range. Use a while loop to keep the user guessing if they haven't guessed the right number and they haven't used all their tries. If the user has correctly guessed the number (without using all their tries), terminate the loop.

If the user has correctly guessed the number, congratulate them and indicate the number of tries the user took to guess the number, otherwise, display the correct number.

Sample output 1

```
Enter a number between 1 and 100 (inclusive): 75
Too high. Enter another guess: 65
Too high. Enter another guess: -50
Really? Enter another guess between 1 to 100: 50
Too low. Enter another guess: 55
Too high. Enter another guess: 52
You guessed it right!! You got it in 5 guesses!
```

Sample output 2

```
Enter a number between 1 and 100 (inclusive): 200
Very funny. Enter a number between 1 and 100 (inclusive): 100
Too high. Enter another guess: 1
Too low. Enter another guess: 99
Too high. Enter another guess: -2
Really? Enter another guess between 1 to 100: 2
Too low. Enter another guess: 98
Too high. Enter another guess: 3
Too low. Enter another guess: 97
Too high. Enter another guess: 4
Too low. Enter another guess: 96
Too high. Enter another guess: 5
Sorry, the number was 53
```

Drawing Polygons with the turtle module (15 points)

Write a program that prompts the user for the number of sides of a polygon and the length of each side. The program should use this information to draw the polygon!

The program should use input validation to ensure the number of sides of the polygon are between 3 and 12 inclusive, and the length of each side is greater than 50. (The length will be used to determine the number of pixels of each side when the shape is drawn).

You will need to calculate the angle between each side of the polygon. This angle, in turn is used to determine the angle at which the turtle should turn to draw the polygon.

The formula used to calculate the angle between each side is

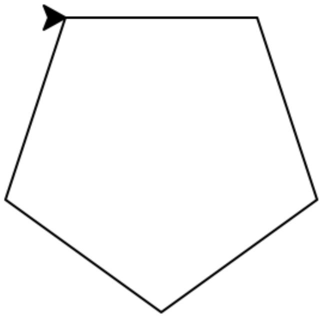
$$angle = \frac{180\ (n-2)}{n}$$

where n is the number of sides. The angle at which the turtle should turn will be calculated as 180 – angle.

e.g., for a triangle, the angle between each side is $\frac{180\ (3-2)}{3} = \frac{180}{3} = 60$. Therefore, the angle in which the turtle should turn to draw the triangle is 180 – 60 = 120.

Sample output is shown below:

```
Enter the number of sides between 3 and 12 (inclusive): -5
Invalid number of sides. Enter a number between 3 and 12 (inclusive): 5
Enter the length of each side (> 50): 10
Please enter a larger length of each side: 80
```



Extra credit (10 points):

Include text to label each polygon after your program draws them! You can find the names of the polygon here: https://en.wikipedia.org/wiki/Polygon

Include an if elif else statement that labels each polygon based on the number of sides. Use the turtle module to write text on the graphics window after the polygon has been drawn. You can learn how to do this by checking out the python documentation: https://docs.python.org/3.10/library/turtle.html

Sample output is shown below:

```
Enter the number of sides between 3 and 12 (inclusive): 6
Enter the length of each side (> 50): 100
```