`while` Loops

For each of these problems, think about what variable controls your loop, what your initial conditions should be, how the loop control variable should be updated, and the conditions that cause your loop to terminate.

1. **Bug Collector**
   A bug collector collects bugs every day for fve days. Write a program that keeps a running total of the number of bugs collected during the fve days. The loop should ask for the number of bugs collected for each day, and when the loop is finished, the program should display the total number of bugs collected.

2. **Budget Analysis**
   Write a program that asks the user to enter the amount that he or she has budgeted for a month. A loop should then prompt the user to enter each of his or her expenses for the month and keep a running total. When the loop fnishes, the program should display the amount that the user is over or under budget.

3. **Distance Traveled**
   The distance a vehicle travels can be calculated as:

   $$distance = speed \times time$$

   For example, if a train travels 40 miles per hour for three hours, the distance traveled is 120 miles.
   Write a program that asks the user for the speed of a vehicle (in miles per hour) and the number of hours it has traveled. It should then use a loop to display the distance the vehicle has traveled for **each hour** of that time period. Here is an example of the desired output, where **bold underline** represents user input:

   ```
   What is the speed of the vehicle in mph? 40
   How many hours has it traveled? 3
   Hour         Distance Traveled
   1                    40
   2                    80
   3                   120
   ```

4. **Pennies for Pay**
   Write a program that calculates the amount of money a person would earn over a period of time if his or her salary is one penny the frst day, two pennies the second day, and continues to double each day. The program should ask the user for the number of days.
   Display a table showing what the salary was for each day, then show the total pay at the end of the period. The output should be displayed in a **dollar amount**, not the number of pennies

5. **Sum of Numbers**
   Write a program with a loop that asks the user to enter a series of positive numbers. The user should enter a negative number to signal the end of the series. After all the positive numbers have been entered, the program should display their sum,

6. **Ocean Levels**
   Assuming the ocean's level is currently rising at about 1.6 millimeters per year, create an application that

displays the number of millimeters that the ocean will have risen each year for the next 25 years.

7. **Tuition Increase**
   At one college, the tuition for a full-time student is $8,000 per semester. It has been announced that the tuition will increase by 3 percent each year for the next 5 years. Write a program with a loop that displays the projected semester tuition amount for the next 5 years.

8. **Calculating the Factorial of a Number**
   In mathematics, the notation n! represents the factorial of the nonnegative integer n. The factorial of n is the product of all the nonnegative integers from 1 to n. For example:

$$7! = 7 \times 6 \times 5 \times 4 \times 3 \times 2 \times 1 = 5040$$

   and

$$4! = 4 \times 3 \times 2 \times 1 = 24$$

   Write a program that lets the user enter a **nonnegative integer** then uses a loop to calculate the factorial of that number. Display the factorial.