1. Write a python program that accepts a list of country names as input and returns the longest country name as output.

Sample: Longest_Country_Name["Australia", "Germany", "United States of America"] Output: United States of America

```
countryList = [] # Initialize the list
k = int(input('How many country names do you want to enter? '))
length = 0

for i in range(k): # Loop to get k country names
    country = input("Enter country name[i+1]: ")
    countryList.append(country)  # Add the country entered to the list "country

    if length < len(country):   # Compare the length of the current country wit
       longest_country = country
       length = len(longest_country)

# Print the longest country name and the list of countries
print(f"The longest country name of {countryList} is '{longest_country}'.")
```

```
How many country names do you want to enter? 2
Enter country name: Australia
Enter country name: Germany
The longest country name of ['Australia ', 'Germany'] is 'Australia '.
```

2. Write a python program that iterates integers from 1 to 15 prints the number . But for multiples of three print "Fizz" instead of the number and for multiples of five print "Buzz". For number that is multiple of both three and five print "FizzBuzz".

```
for num in range(1, 16)
 if num % 3 == 0 and num % 5 == 0:
    print("FizzBuzz")
  elif num % 3 == 0:
    print("Fizz")
  elif num % 5 == 0:
    print("Buzz")
  else:
    print(num)
```

3.Write a program that takes a n number as input, stores the numbers in list and prints the numbers in list in reversed order with its index.

```
# Take the number of elements as input
n = int(input("Enter the number of elements: "))


numbers = []# Initialize an empty list to store the numbers

for i in range(n):# gets n numbers from the user and adds to list
    num = int(input(f"Enter number {i+1}: "))
    numbers.append(num)

# Print the numbers in reversed order with their index
for i,num in enumerate(reversed(numbers)):
    print(f"Index {i}, Number {num}")
```

```
Enter the number of elements: 2
Enter number 1: 4
Enter number 2: 8
Index 0, Number 8
Index 1, Number 4
```

4. Write a python program using a for loop to count the number of vowels in a given string.

```
# Take the input string from the user
input_string = input("Enter a string: ")

vowels = ['a','e','i','o','u'] # Initialize a list of vowels

vowel_count = 0 # Initialize a counter for vowels

for char in input_string.lower():
    if char in vowels:
        vowel_count += 1

print(f"Number of vowels in the string: {vowel_count}")
```

```
Enter a string: fron
Number of vowels in the string: 1
```

5. We want to create a program that will display every character of a string in new line with its position.

```
input_string = input("Enter a string: ")

for index, char in enumerate(input_string):
    print(f"Character '{char}' at position {index}")
```

```
Enter a string: rtg
Character 'r' at position 0
Character 't' at position 1
Character 'g' at position 2
```

6.Write a program in Python to display a pattern like a right-angle triangle using an asterisk. (Example Pattern)

```
*
**
***
****
```

```
n = int(input("Enter the number of rows: "))

for i in range(1, n+1):
    print('*' * i)
```

```
Enter the number of rows: 7
*
**
***
****
*****
******
*******
```

7.Write a program in Python to make a pyramid pattern with numbers increased by 1.

```
1
2 3
4 5 6
7 8 9 10
```

```
n = int(input("Enter the number of rows: "))

# Initialize the starting number
num = 1

# Loop to print each row
for i in range(1, n+1):
    for j in range(i):
        print(num, end=" ")
        num += 1
    print()
```

8.Write a program in Python to display the n terms of a harmonic series and their sum.

(1+ 1/2 + 1/3 + 1/4 + 1/5 ... 1/n)

```
n = int(input("Enter the number of terms: "))

# Initialize the sum of the harmonic series
harmonic_sum = 0.0

# Loop to display each term and calculate the sum
print("Harmonic series:")
for i in range(1, n+1):
    term = 1 / i
    harmonic_sum += term

# Print the sum of the harmonic series
print(f"\nSum of the harmonic series: {harmonic_sum}")
```

```
Enter the number of terms: 4
Harmonic series:

Sum of the harmonic series: 2.083333333333333
```

9.Write a python program to check whether a given number is an Armstrong number or not. For example, 153 is an Armstrong number because 153 = 13+53+33

.

```
# Take input from the user
num = input("Enter a number: ")

# Find the number of digits in the number
num_digits = len(str(num))

# Initialize the sum of powers
sum_of_powers = 0

# Make a copy of the original number
temp = num

# Calculate the sum of the digits raised to the power of num_digits
for digit in num:
    sum_of_powers += int(digit) ** num_digits
    print(sum_of_powers)

# Check if the sum equals the original number
if sum_of_powers == int(num):
    print(f"{num} is an Armstrong number.")
else:
    print(f"{num} is not an Armstrong number.")
```

```
Enter a number: 153
1
126
153
153 is an Armstrong number.
```

10. We are developing a DeLorean Test Drive game with our friend Emmett Brown.

The game will be a test drive of our time travel machine. The user will start with a vehicle that is stopped (0 mph). We will display on the screen that the user's current speed is 10mph. We will then ask them if they want to accelerate or brake. If they accelerate, then 10mph is added to their speed. If they brake, then 10mph is reduced to their speed. Minimum Speed that a person can reach is 0 and if speed goes to negative update the speed to minimum speed and warn the user that that they cannot brake anymore. We will then ask them if they want to accelerate/brake. We keep asking them and keep adding 10mph to their speed until they tell us to stop. Once we stop, we will print out the top speed that they achieved. There is one twist to our program – if they do not stop before going more than 88 mph, the program automatically hits the brakes and lets them know that they have travelled in time.

```
# Starting conditions
speed = 0  # Starting speed
top_speed = 0  # Track the top speed
time_travelled = False  # Track if time travel occurred

print("Welcome to the DeLorean Test Drive!")
print("Your current speed is 0 mph.")

# Main game loop
while True:
    action = input("Do you want to 'accelerate' or 'brake'? (type 'stop' to end): ")

    if action == 'accelerate':
        speed += 10  # Increase speed by 10 mph
        print(f"Your current speed is {speed} mph.")

        # Check if speed exceeds 88 mph
        if speed > 88:
            time_travelled = True
            print("You've exceeded 88 mph! Time travel initiated!")
            break  # End the game if they time travel

        # Update the top speed achieved
        if speed > top_speed:
            top_speed = speed

    elif action == 'brake':
        speed -= 10  # Decrease speed by 10 mph
        if speed < 0:
            speed = 0  # Ensure speed doesn't go below 0
            print("You've reached a negative speed! You can't brake anymore.")
        print(f"Your current speed is {speed} mph.")

    elif action == 'stop':
        print("You have chosen to stop the test drive.")
        break  # Exit the loop if they choose to stop

    else:
        print("Invalid input. Please type 'accelerate', 'brake', or 'stop'.")

# Game over message
if time_travelled:
    print("Congratulations! You've traveled through time!")
else:
    print(f"Your top speed was {top_speed} mph.")
    print("Thank you for playing the DeLorean Test Drive!")
```

11. Get user information from k user and create a user ID and random 4-digit pin number as a password for each new user account and display them as given below. User ID must be generated within a specific pattern.

Information to be received from user.

1. First Name
2. Last Name
3. Year of Birth(dd-mm-yyyy)
4. Email ID

Username Pattern: "**<First 3 characters of First Name> <First 3 characters of Last Name> <dd (Date**

```
import random # import library

first_name = input("Enter First Name: ")
last_name = input("Enter Last Name: ")
dob = input("Enter Year of Birth (dd-mm-yyyy): ")
```