Question 1: Code Along
Problem: Write a Python program that takes a string as input and prints out the following:

1. The string in reverse order.
2. The number of vowels in the string.

Instructions:

- Use basic string manipulation techniques such as loops, conditionals, and string methods.
- Ensure that the program counts both uppercase and lowercase vowels (a, e, i, o, u).

Example Input:

Enter a string: Hello World

Example Output:

Reversed string: dlroW olleH

Number of vowels: 3

```
# Ask to user enter the input
name = input("Enter your Name: ")

# 1.  The string in reverse order.
reverse_input = name[::-1]
print("Reversed string: ",reverse_input)     # Display the reverse
input

Reversed string:  naqruF

# 2.  The number of vowels in the string.
vowels = {'a', 'e', 'i', 'o', 'u', 'A', 'E', 'I', 'O', 'U'}     # Set
of vowels
count = 0

for char in name:
    if char in vowels:
        count += 1     # Increment

print("Number of vowels: ", count)

Number of vowels:  2
```

Question 2: Hands-on Coding Project

Problem: Create a Python program that:

- Takes an input number from the user.
- Checks whether the number is even or odd.

● Prints the result.

Instructions:

● Write a Python script that accepts a number from the user, determines

whether it is even or odd, and prints the result.

● Use conditionals to check if the number is even or odd.

Example Input:

Enter a number: 5

Example Output:

The number 5 is Odd.

```
# Ask to user enter input
num = int(input("Enter a number: "))

if num%2==0:
    print(f"The number {num} is Even.")
else:
    print(f"The number {num} is odd.")
```

```
The number 45 is odd.
```

Question 3: Virtual Environment Application

Problem: Create a Python program that:

1. Takes a list of integers as input.
2. Creates a new virtual environment called sortenv.
3. Installs a package (such as numpy) in the virtual environment.
4. Sorts the list using a numpy method (numpy.sort()).
5. Prints the sorted list.

Instructions:

● You will not be expected to actually run the environment creation commands but rather simulate what would happen by describing the steps in your code and then focusing on the sorting operation.

● Make sure that your program explains each step as comments and performs sorting using numpy.

Example Input:

Enter a list of numbers: [4, 2, 7, 1, 3]

Example Output:

Sorted list: [1, 2, 3, 4, 7]

```
import numpy as np

user_input = input("Enter a list of numbers separated by space: ")
input_list = list(map(int, user_input.split()))

sorted_list = np.sort(input_list)
print("Sorted list: ", sorted_list)
```

```
Sorted list:  [0 1 3 4 6 7 9]
```