DATA SCIENCE COURSE TUTORIAL # 24

3.16 Range and Enumerate Functions

In Python, `range()` and `enumerate()` are very useful functions when working with loops.

Range Function

The `range()` function generates a sequence of numbers. It is commonly used in loops.

Syntax:

```
range(start, stop, step)
```

- **start** → Starting number (default 0).
- **stop** → End number (not included).
- **step** → Difference between numbers (default 1).

Example:

```
for i in range(5):
    print(i)
```

Output:

```
0
1
2
3
4
```

Example with start and step:

```
for i in range(2, 10, 2):
    print(i)
```

Output:

```
2
4
```

```
6
8
```

Enumerate Function

The `enumerate()` function adds an index to each element in an iterable (like a list or string). Useful when you need both index and value.

Example:

```
fruits = ["apple", "banana", "cherry"]
for index, fruit in enumerate(fruits):
    print(index, fruit)
```

Output:

```
0 apple
1 banana
2 cherry
```

Example with custom start index:

```
for index, fruit in enumerate(fruits, start=1):
    print(index, fruit)
```

Output:

```
1 apple
2 banana
3 cherry
```

Summary

- **range()** → Creates a sequence of numbers for loops.
- **enumerate()** → Provides index along with elements in an iterable.