Experiment11

April 8, 2025

Name - Shravani Sandeep Raut

SE - 48

[1]:
```
import pandas as pd
```

[2]:
```
# Step 1: Creating a Data Series using a list
data_list = [10, 20, 30, 40, 50]
series_from_list = pd.Series(data_list)
```

[3]:
```
# Step 2: Creating a Data Series using a dictionary
data_dict = {'A': 10, 'B': 20, 'C': 30, 'D': 40, 'E': 50}
series_from_dict = pd.Series(data_dict)
```

[4]:
```
# Step 3: Display the Series
print("Series from List:")
print(series_from_list)

print("\nSeries from Dictionary:")
print(series_from_dict)
```

```
Series from List:
0    10
1    20
2    30
3    40
4    50
dtype: int64

Series from Dictionary:
A    10
B    20
C    30
D    40
E    50
dtype: int64
```

[5]:
```
# Step 4: Accessing Elements (by index)
```

```
print("\nAccessing the element at index 2 in series_from_list:",␣
 ↪series_from_list[2])
```

```
Accessing the element at index 2 in series_from_list: 30
```

[6]:
```
# Step 5: Operations on the Series (e.g., adding a constant)
series_added = series_from_list + 10
print("\nAdding 10 to each element in series_from_list:")
print(series_added)
```

```
Adding 10 to each element in series_from_list:
0    20
1    30
2    40
3    50
4    60
dtype: int64
```

[7]:
```
# Step 6: Applying a function (e.g., doubling the values)
doubled_series = series_from_list.apply(lambda x: x * 2)
print("\nDoubling the values of series_from_list:")
print(doubled_series)
```

```
Doubling the values of series_from_list:
0     20
1     40
2     60
3     80
4    100
dtype: int64
```

[8]:
```
# Step 7: Checking for NaN values (example with missing data)
series_with_nan = pd.Series([1, 2, None, 4, 5])
print("\nSeries with NaN value:")
print(series_with_nan)

print("\nChecking for NaN values:")
print(series_with_nan.isna())
```

```
Series with NaN value:
0    1.0
1    2.0
2    NaN
3    4.0
4    5.0
```

```
dtype: float64

Checking for NaN values:
0    False
1    False
2     True
3    False
4    False
dtype: bool
```