

## 14 Exercises

### 1. Creating a Series for Product Prices

- (a) A company has a list of prices for five products: [250, 500, 750, 1000, 1250]. Create a pandas Series for these prices.
- (b) Assign product names as the index for the Series: ['Product A', 'Product B', 'Product C', 'Product D', 'Product E'].
- (c) Access the price of Product C.

### 2. Series Operations on Product Data

- (a) Sort the Series by product names (index) in alphabetical order.
- (b) Sort the Series by product prices in descending order to find the most expensive product.
- (c) Use describe() to get a quick statistical summary of the product prices.

### 3. Series Creation

- (a) Create a Series for a list of items in stock: ['Laptop', 1200, 'Tablet', 800, 'Monitor', 300].

### 4. Creating a DataFrame for Customer Orders

- (a) Given the dictionary { 'Customer': ['Alice', 'Bob', 'Charlie'], 'Order Amount': [200, 450, 300], 'Location': ['NY', 'LA', 'Chicago'] }, create a pandas DataFrame.
- (b) Display the DataFrame to review the customer order information.

### 5. DataFrame Index Operations for Customer Orders

- (a) Reset the index of the DataFrame to the default numerical index, and confirm the change by displaying the DataFrame.
- (b) Set the Customer column as the index of the DataFrame.
- (c) Reset the index back to the default numerical index again.

### 6. Sorting and Describing Customer Orders

- (a) Sort the DataFrame by the Order Amount in ascending order to see the smallest orders first.
- (b) Sort the DataFrame by the Location column in alphabetical order to group orders by city.
- (c) Use describe() to obtain a summary of the Order Amounts.

### 7. Filtering Data Using Boolean Indexing

- (a) Using the Order Amount column, filter the DataFrame to show only orders where the amount is greater than 300.
- (b) Filter the DataFrame to display only the orders placed by customers located in 'NY'.