

Übung 1: Grundlegende Series-Erstellung und Zugriff auf Elemente



Aufgabe 1:

- Erstelle eine `Series` mit den folgenden Daten: ['Apfel', 'Banane', 'Kirsche', 'Datteln', 'Erdbeere'] und nutze als Index die Buchstaben ['A', 'B', 'C', 'D', 'E'].
- Greife auf das zweite und das letzte Element der Serie zu.
- Ändere das erste Element der `Series` auf 'Ananas' und gib die Serie erneut aus.

Aufgabe 2:

- Erstelle eine `Series` mit Temperaturen der Woche: `Temperaturen = [20, 21, 19, 22, 23, 24, 18]` und als Index die Tage der Woche ['Mo', 'Di', 'Mi', 'Do', 'Fr', 'Sa', 'So'].
- Gib die Temperatur für Mittwoch aus.
- Ändere die Temperatur für Sonntag auf 19 und gib die Serie erneut aus.

Aufgabe 3:

- Erstelle eine `Series` mit den Namen ['Alice', 'Bob', 'Charlie', 'David', 'Eve'] und ordne den Namen jeweils eine zufällige Punktzahl zu, z.B. [85, 90, 78, 88, 92].
- Greife auf die Punktzahl von Bob zu und aktualisiere seine Punktzahl auf 95.
- Gib die `Series` mit den aktualisierten Werten aus.

Übung 2: Mathematische Operationen auf Series

Aufgabe 1:

- Erstelle eine `Series` namens `preise` mit den Werten [2.5, 3.0, 4.5, 3.75, 2.25], die die Preise von Artikeln darstellt.
- Multipliziere alle Werte in der `Series` mit 1.19, um die Preise inklusive Mehrwertsteuer zu berechnen.
- Gib die `Series` nach der Berechnung aus und berechne anschließend den Gesamtpreis.

Aufgabe 2:

- Erstelle eine Series namens `vorrat` mit den Werten `[5, 3, 6, 8, 2]`, die die Anzahl der vorrätigen Artikel in einem Lager darstellt.
- Erhöhe die Werte um 10 %, um eine Lagerauffüllung zu simulieren.
- Gib die durchschnittliche Anzahl der Artikel im Lager vor und nach der Erhöhung an.

Aufgabe 3:

- Erstelle eine Series namens `umsatz` mit monatlichen Umsätzen in Tausend `[50, 45, 60, 55, 70, 65]`.
 - Berechne für jeden Monat einen Bonus von 10 % des Umsatzes und speichere diesen als neue Series `bonus`.
 - Gib den Gesamtwert von `bonus` sowie die Summe von `umsatz + bonus` aus.
-

Übung 3: Logische Operationen und Filter

Aufgabe 1:

- Erstelle eine Series namens `alter` mit Alterswerten `[18, 21, 30, 16, 22, 25, 17, 19]`.
- Filtere alle Einträge, bei denen das Alter über 18 Jahre liegt, und speichere diese in einer neuen Series.
- Gib die Anzahl der Alterswerte an, die genau 18 Jahre betragen.

Aufgabe 2:

- Erstelle eine Series mit dem Namen `punkte` und den Werten `[60, 45, 88, 72, 90, 55, 40]`, die die Punktzahl verschiedener Teilnehmer darstellt.
- Filtere alle Teilnehmer heraus, die mindestens 70 Punkte erreicht haben, und gib diese in einer neuen Series aus.
- Gib außerdem die Anzahl der Teilnehmer mit weniger als 50 Punkten an.

Aufgabe 3:

- Erstelle eine `Series` namens `note` mit Noten `[2.5, 3.0, 1.7, 4.0, 2.0, 3.3, 1.0, 2.8]`.
 - Filtere alle Einträge heraus, bei denen die Note besser als 2.0 ist, und speichere diese in einer neuen `Series` `beste_noten`.
 - Gib die Anzahl und den Durchschnittswert der Noten in `beste_noten` aus.
- 