

14 Exercises

- What is the output of running this code?

What is the output?

```
list_1 = []
for item in range(3):
    list_1[item].append(item)
print(item)
print(list_1)
```

- Create a one-dimensional array (a `list` or a `tuple`) with 10 integer numbers and try to access the item at index 10. What happens when you run your code?
- Create a `tuple` with 5 items. Convert the `tuple` into a `list`, add two more items, and convert it back to a `tuple`. What are the final contents of the `tuple`?
- Create a `tuple` with 5 numbers, then try to change the third number in the `tuple` to another value. What happens? How can you "modify" the `tuple` while keeping its immutability intact?
- Consider the following snippet

working with str

```
str_a = "Python is fun!"
```

Use slicing to reverse the `str`, then extract only the characters from the 4th to the 8th position (inclusive) in the reversed `str`. What is the result?

- Create a `range()` that starts from 5 and ends at 25, incrementing by 3. Convert this range into a `list`. Can you change the 5th item in the `list` to 100? What happens if you try to change the item directly in the `range`?
- Use the `range()` function to generate a sequence of numbers from 0 to 100 (inclusive) with a step of 10. Convert this sequence into a `list` and calculate the sum of the elements. What is the result?
- You are given a list of daily sales figures for a week:

daily sales data

```
sales = [100, 150, 200, 250, 300, 350, 400]
```

Write a snippet that calculates the cumulative sum (`cumsum`) of the sales list using basic `list` operations (i.e., without using any external libraries like `numpy` or Python's built-in `itertools` functions). **Hint:** rely on `append()` method inside a `for` loop. The output `list` should be:

ideal output for cumsum

```
cumsum_sales = [100, 250, 450, 700, 1000, 1350, 1750]
```

As an additional challenge: write a snippet that calculates the running average of the sales figures as well. The running average up to each day is the cumulative sum divided by the number of days up to that

point. The ideal output would be:

ideal output for running average

```
running_avg_sales = [100, 125, 150, 175, 200, 225, 250]
```

9. Create a dictionary where the keys are the first 5 letters of the alphabet and the values are their corresponding positions (e.g., {'a': 1, 'b': 2, ...}). Swap the keys and values so that the letters become values and the numbers become keys. Write a snippet to perform the swap automatically. **Hint:** You may want to consider dictionary comprehension.
10. The following dataset represents weekly sales figures for a company. Calculate the average sales.

Weekly sales

```
weekly_sales = [200, 300, 250, 400, 350, 380, 420]
```

Reminder:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

11. The dataset below represents stock prices over 10 days. Calculate the (sample) standard deviation of the stock prices. You must use list comprehension. Feel free to use functions from the `math` module.

stock prices

```
stock_prices = [100, 102, 98, 105, 107, 106, 103, 110, 108, 107]
```

Reminder:

$$s = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2}$$

12. Consider the following two sets:

set operations

```
set_a = {1, 2, 3, 4, 5}
set_b = {4, 5, 6, 7, 8}
```

Perform and print the following operations: union, intersection, and difference between the two sets. What is the result when you check whether `set_a` is a subset of `set_b`?

13. Create a `frozenset` from a list of numbers `[10, 20, 30, 40, 50]`. Try adding a new number to this `frozenset`. What happens? Why can't you modify it?
14. Consider the following snippet containing two small datasets:

price datasets

```
prices_A = [100, 102, 98, 105, 107, 106, 103, 110, 108, 107]
prices_B = [101, 103, 99, 106, 108, 105, 102, 111, 109, 108]
```

These two lists represent the stock closing prices of two companies over a period of 10 days. Using list comprehension, determine on which days company B had a higher closing price than company A.

15. Create a `list` of numbers from 1 to 50. Using list comprehension, create a new list that contains the squares of all odd numbers.
16. It looks like both of the following snippets delete `list` elements. What is the difference between these two snippets, if any?

Snippet A

```
sample_list = [10, 20, 30]
sample_list[:] = []
print(sample_list)
```

Snippet B

```
sample_list = [10, 20, 30]
sample_list = []
print(sample_list)
```

Hint: Use the `id` function before and after modifying the `list` in each snippet to see the difference.

17. Create a dictionary where the keys are names of 3 students and the values are another dictionary containing their scores in three subjects: Data Analytics, Business Analytics, and Python for Data Analysis. Access and print the Business Analytics score of the second student. Then, update the Python for Data Analysis score of the first student.
18. A warehouse manager at a factory monitors the daily demand for a certain tool. Based on historical data, the daily demand, X , can take any value between 0 and 9 (inclusive) with the following probabilities:

probability mass function (pmf)

```
demand = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
probabilities = [0.1, 0.05, 0.07, 0.01, 0.08, 0.12, 0.15, 0.09, 0.14, 0.09]
```

- Use `zip()` to combine the `demand` and `probabilities` lists into a dictionary.
- Using dictionary comprehension, calculate the expected daily demand and variance for the tool.
- The factory incurs a cost of \$100 each time the tool is used. Define $Y = 100X$, as the daily cost. Find the mean and variance of the daily cost Y .

Remember that

$$E(X) = \sum_i x_i \cdot P(X = x_i) \quad \text{and} \quad \text{Var}(X) = E(X^2) - (E(X))^2$$

19. Create a dictionary with some names (as strings) as keys and ages as values. Try adding a new key that is the same name but with a different case (e.g., "Anna" and "anna"). What happens? Does the dictionary treat them as different keys?
20. Consider the following snippet

Count chars

```
str_1 = "Success is the result of consistent, persistent effort."
```

Use a dictionary to count how many times each character appears in this `str`. Ignore spaces and make sure the count is case-insensitive.

21. Consider the following list:

list with repetition

```
list_nums = [1, 2, 1, 3, 5, 7, 7, 6, 3, 10, 7, 2, 1]
```

Write a snippet that returns a new `list` with only the unique elements from the above `list`. Do this in two different ways: with and without `set()` function.

22. Consider the following data

Employee salaries

```
employees = {'Emily': 48_000, 'Andrew': 52_000, 'Mike': 47_000, 'Anna': 60_000}
```

Write a snippet that increases the salary of every employee by 10%, but only if their salary is below \$50_000. Print the updated dictionary of employees and their salaries.

23. An external audit of a company has revealed the following annual salaries (in thousands of dollars) for employees of an organization.

salaries

```
salaries = [50, 40, 500, 50, 40, 50, 40, 40, 80, 40]
```

Do you notice a significant difference between the mean and median salary values? Why do you think this difference exists?

Hint: A common way to find a median is to sort the data first and then use the following formula in the sorted data:

$$\text{Median} = \begin{cases} \text{Middle value,} & \text{if } n \text{ is odd} \\ \frac{1}{2} \times (\text{Sum of two middle values}), & \text{if } n \text{ is even} \end{cases}$$

24. A shoe store wants to store the most common shoe sizes to meet customer demand. The following `list` contains the shoe sizes sold in a retail store over the past month.

sold shoe sizes

```
shoe_sizes = [
    10, 9, 7, 6, 11, 8, 7, 9, 10, 8, 7, 12, 6, 9, 10, 9, 10, 10, 8, 7,
    7, 6, 8, 9, 7, 11, 6, 6, 9, 8, 7, 9, 8, 10, 11, 7, 7, 6, 12, 10,
    8, 7, 10, 11, 9, 7, 8, 8, 7, 6, 12, 8, 10, 6, 10, 11, 9, 7, 6, 7,
    8, 10, 9, 12, 11, 6, 10, 6, 9, 7, 9, 7, 8, 10, 7, 9, 7, 10, 8, 8,
    7, 8, 9, 11, 12, 9, 10, 8, 11, 9, 6, 10, 7, 6, 7, 6, 10, 8, 9, 12]
```

]

- (a) Find mean, median, and mode of the sold shoe sizes.
 (b) Determine which metric (mean, median, or mode) is the most appropriate for answering the store manager's question.

25. Consider the following snippet showing the closing stock prices of three companies over several days:

stock price datasets

```
# nvidia prices
nvid_p = [113.37, 117.87, 116.0, 116.26, 120.87, 123.51, 124.04,
          121.4, 121.44, 117.0, 118.85, 122.85, 124.92, 127.72,
          132.89, 132.65, 134.81, 134.8, 138.07, 131.6, 135.72, 136.93]

# apple prices
appl_p = [220.69, 228.87, 228.2, 226.47, 227.37, 226.37, 227.52, 227.79,
           233.0, 226.21, 226.78, 225.67, 226.8, 221.69, 225.77, 229.54,
           229.04, 227.55, 231.3, 233.85, 231.78, 232.15]

# microsoft prices
msft_p = [430.81, 438.69, 435.27, 433.51, 429.17, 432.11, 431.31,
           428.02, 430.3, 420.69, 417.13, 416.54, 416.06, 409.54,
           414.71, 417.46, 415.84, 416.32, 419.14, 418.74, 416.12, 416.72]
```

- (a) Using list comprehension, calculate the sample covariance between nvidia and microsoft stock prices.
 (b) Using list comprehension, calculate the sample variance of apple stock prices.
 (c) Using list comprehension, calculate the sample covariance of apple stock prices with itself.
 (d) What is the relationship between the results from parts b and c? Does the outcome surprise you?
 Reminder: s_{xy} is the sample covariance and s^2 is the sample variance, calculated as:

$$\text{Cov}(x, y) = s_{xy} = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) \quad \text{and} \quad s_x^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$$

26. In the previous problem, using list comprehension:

- (a) Calculate the Pearson correlation coefficient between nvidia and microsoft stock prices.
 (b) Calculate the Pearson correlation coefficient of apple stock prices with itself.
 Reminder: r_{xy} is the sample correlation coefficient, calculated as:

$$\text{Corr}(x, y) = r_{xy} = \frac{s_{xy}}{s_x s_y}$$

27. Consider the following dictionaries

kids favorite color

```
kid_color_1 = {'Locas': 'red', 'Ava': 'green', 'Mia': 'orange'}
kid_color_2 = {'Evelyn': 'black', 'Mia': 'white', }
```

Write a program that merges these two dictionaries. How many different ways do you know to merge these dictionaries? Print the resulting dictionary.

28. Consider the following dictionaries

sales data

```
sales_q_1 = {'tv': 100, 'xbox': 200, 'macbook': 30}
sales_q_2 = {'xbox': 150, 'speaker': 5 }
```

Write a snippet that merges these two dictionaries. If a key exists in both dictionaries, the value should be the sum of the two values.

Hint: this is a challenging question; you may want to consider `if.. else` block inside a `for` loop.

29. Consider the following dictionary with students names and grades:

grades

```
students = {'Noah': 90, 'Jack': 82, 'Sophia': 97, 'Lily': 92, 'Chloe': 77}
```

Write a snippet that returns a `list` of students who have grades above 85.

30. **A Mini Case Study** You are a data analyst working at a financial services company. Your task is to handle a small dataset that represents quarterly financial data for three companies. Below is a table representing the quarterly revenue and expenses for these companies:

Company	Q1 Revenue	Q1 Expenses	Q2 Revenue	Q2 Expenses
Alpha	120,000	80,000	140,000	90,000
Beta	200,000	150,000	220,000	160,000
Gamma	300,000	250,000	320,000	240,000

- (a) Represent this data as a `list` of dictionaries. Each dictionary should have the company name as a key and a sub-dictionary as its value, where the sub-dictionary stores the quarterly revenue and expenses data. The structure should look like this:

sample output

```
[
{
  'Company': 'Alpha',
  'Financials': {
    'Q1 Revenue': 120000,
    'Q1 Expenses': 80000,
    'Q2 Revenue': 140000,
    'Q2 Expenses': 90000
  }
}
```

```
    }  
}, ...  
]
```

- (b) Access and print the Q2 revenue of the company **Beta**.
(c) Access and print the total expenses for **Gamma** across both quarters (sum of Q1 and Q2 expenses).
(d) Calculate and print the net profit (Revenue - Expenses) for each company in **Q1**.
(e) Update the **Q2 expenses** of **Alpha** to 95,000.
(f) Add a new field called **Q1 Profit Margin** for each company in the dictionary. This should be calculated as:

$$\text{Profit Margin} = \left(\frac{\text{Q1 Revenue} - \text{Q1 Expenses}}{\text{Q1 Revenue}} \right) \times 100$$

Update the `list` of dictionaries to include this value for each company.

- (g) A new company, **Delta**, has just been added to your dataset. The company has the following financials:
- Q1 Revenue: 180,000
 - Q1 Expenses: 130,000
 - Q2 Revenue: 190,000
 - Q2 Expenses: 140,000

Add **Delta** to your `list` of dictionaries and print the updated `list`.

- (h) Calculate the total revenue and total expenses across both quarters for all companies.

sample output

```
{'Total Revenue': total_revenue_value,  
'Total Expenses': total_expenses_value  
}
```

Print the summary report.

15 Exercise Solutions

Solutions to these problems can be found on the following GitHub page:

https://github.com/NaserNikandish/Python_For_Data_Analysis

You can also access the same link using the QR code below:

