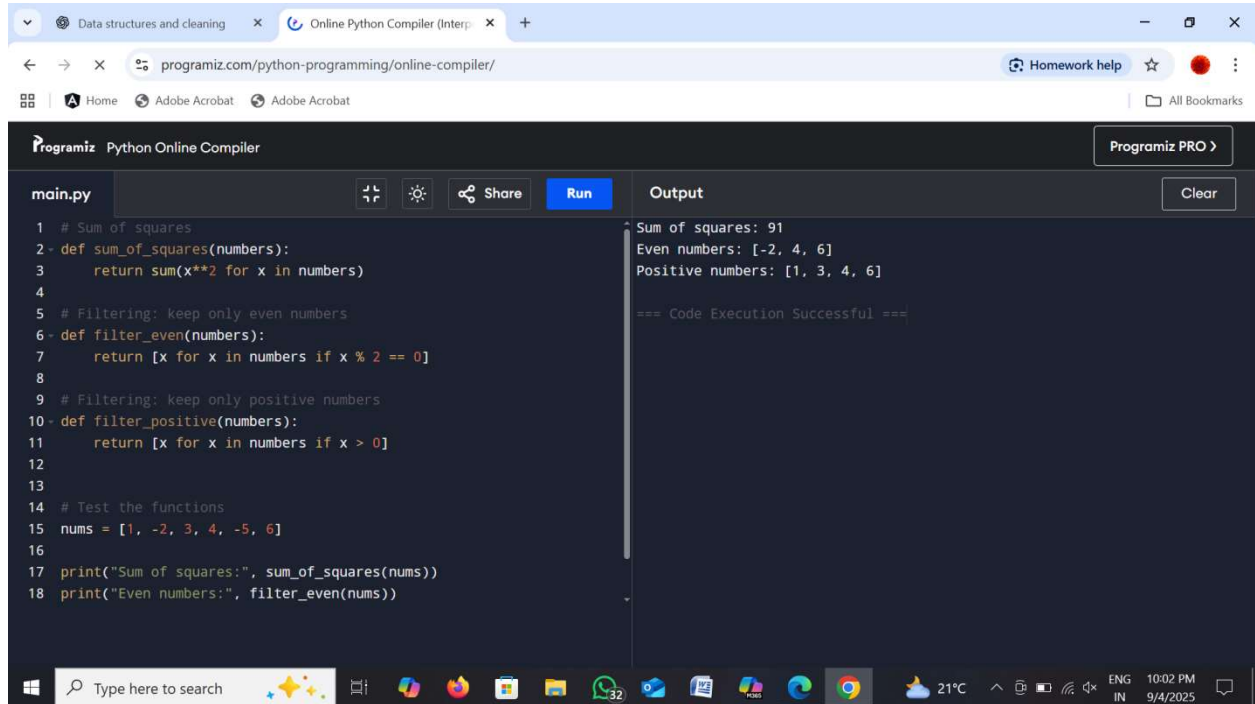


## Week 2: Data Structures and Functions

Hands-On: Work with data structures and write functions for data transformations (e.g., sum of squares, filtering).



The screenshot shows a web browser window with the URL `programiz.com/python-programming/online-compiler/`. The browser tabs include "Data structures and cleaning" and "Online Python Compiler (Interp...". The browser's address bar shows the URL and a "Homework help" bookmark. The browser's toolbar includes "Home", "Adobe Acrobat", and "All Bookmarks".

The main content area is the "Programiz Python Online Compiler" interface. It features a "main.py" tab, a "Run" button, and a "Share" button. The code editor displays the following Python code:

```
1 # Sum of squares
2 def sum_of_squares(numbers):
3     return sum(x**2 for x in numbers)
4
5 # Filtering: keep only even numbers
6 def filter_even(numbers):
7     return [x for x in numbers if x % 2 == 0]
8
9 # Filtering: keep only positive numbers
10 def filter_positive(numbers):
11     return [x for x in numbers if x > 0]
12
13
14 # Test the functions
15 nums = [1, -2, 3, 4, -5, 6]
16
17 print("Sum of squares:", sum_of_squares(nums))
18 print("Even numbers:", filter_even(nums))
```

The "Output" panel on the right shows the results of the code execution:

```
Sum of squares: 91
Even numbers: [-2, 4, 6]
Positive numbers: [1, 3, 4, 6]

=== Code Execution Successful ===
```

The bottom of the screenshot shows a Windows taskbar with a search bar, several application icons, and system tray information including the temperature (21°C), time (10:02 PM), and date (9/4/2025).

Client Project: Write a script for data cleaning (e.g., remove duplicates, filter data).

The screenshot shows the Programiz Python Online Compiler interface. The left pane contains a Python script named `main.py` that uses pandas to clean a dataset. The right pane shows the output of the script, which includes the original data, the cleaned data (with duplicates removed and rows filtered by age), and a success message.

**main.py**

```
1 import pandas as pd
2 # Sample dataset (you can replace with CSV or Excel file)
3 data = {
4     "Name": ["Alice", "Bob", "Charlie", "Alice", "David", "Bob"],
5     "Age": [25, 30, 35, 25, 40, 30],
6     "City": ["NY", "LA", "SF", "NY", "LA", "LA"]
7 }
8 df = pd.DataFrame(data)
9 print("Original Data:")
10 print(df)
11 # Step 1: Remove duplicates
12 df_clean = df.drop_duplicates()
13 # Step 2: Filter (e.g., only Age > 30)
14 df_clean = df_clean[df_clean["Age"] > 30]
15 print("\nCleaned Data:")
16 print(df_clean)
```

**Output**

Original Data:

	Name	Age	City
0	Alice	25	NY
1	Bob	30	LA
2	Charlie	35	SF
3	Alice	25	NY
4	David	40	LA
5	Bob	30	LA

Cleaned Data:

	Name	Age	City
2	Charlie	35	SF
4	David	40	LA

=== Code Execution Successful ===