**Introduction to Python**

Python is a powerful, easy-to-learn programming language widely used in web development, automation, data science, and more. It has a simple syntax that makes it beginner-friendly.

**Data Types in Python**

Python has different types of data used to store values:

• Integer (int): Whole numbers

• Float: Decimal numbers

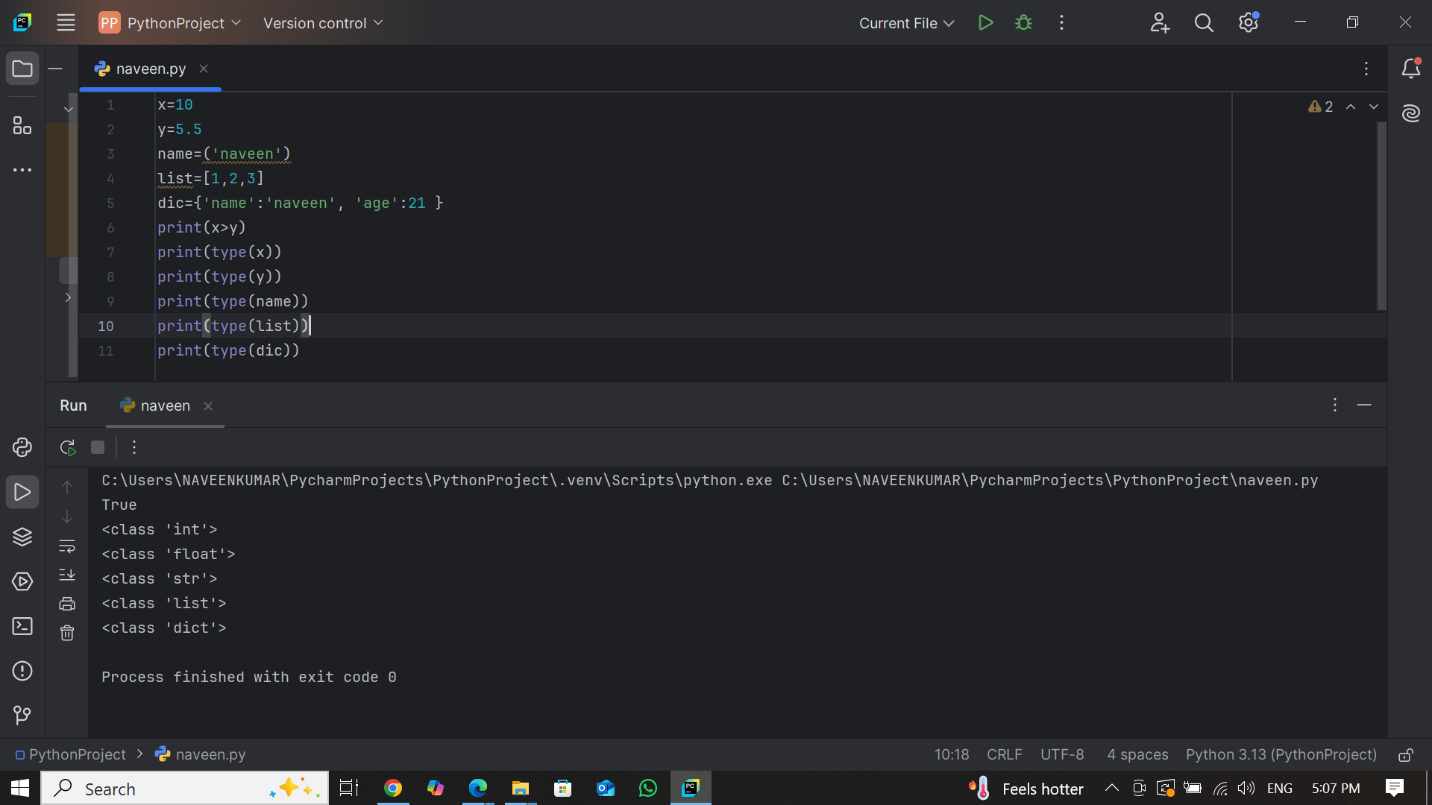
• String (str): Text enclosed in quotes

• Boolean (bool): True/False values

• List: Ordered, mutable, dynamic, indexable, iterable.

• Dictionary: Key-value, unordered, mutable, fast, flexible.

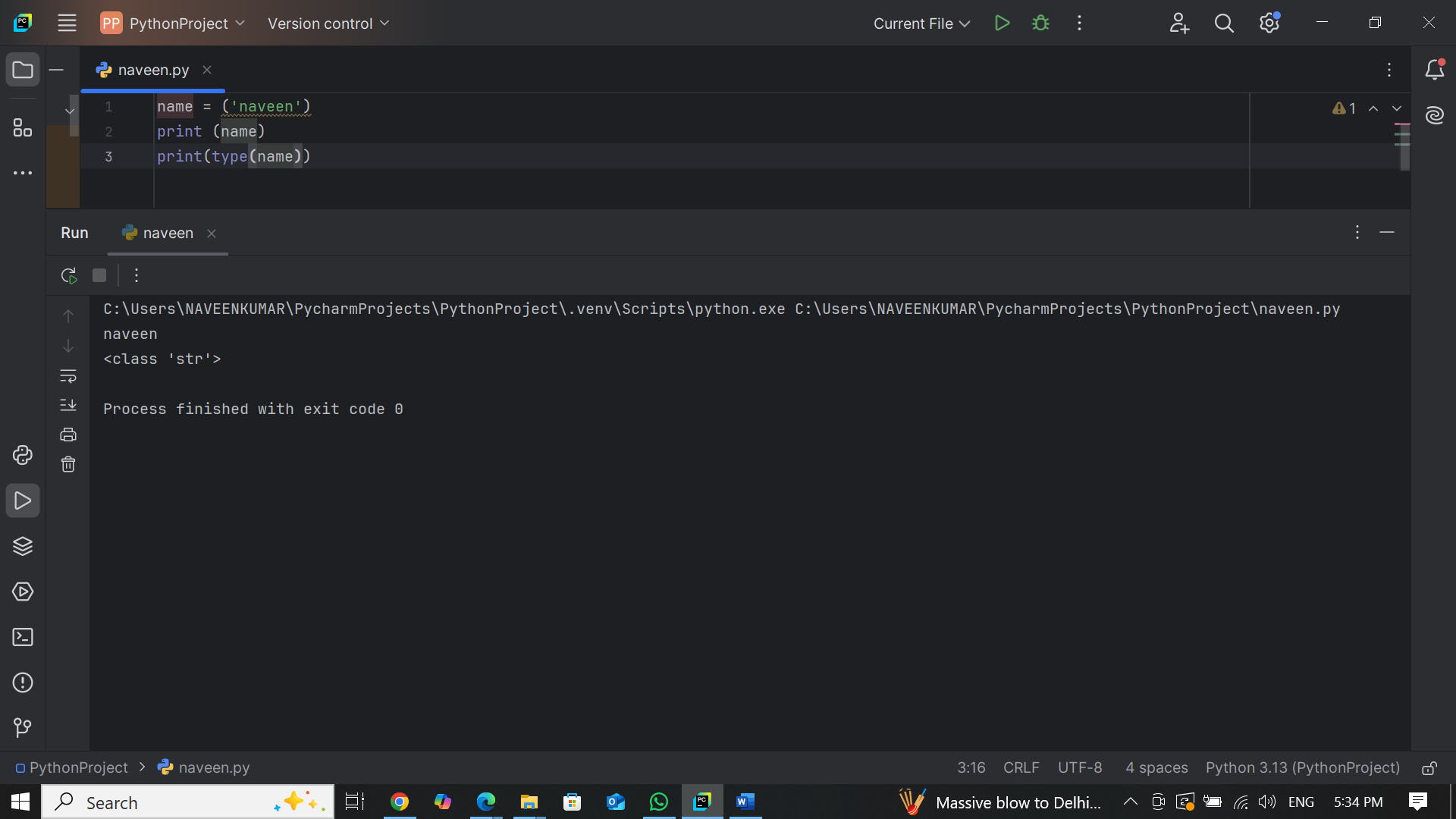
• Tuple : ordered, immutable, faster, hashable, fixed-size.



**Variables in Python**

Variables store values for later use. Python allows dynamic typing, meaning you don’t need to declare variable types explicitly.

**Example:**



**Loops in Python**

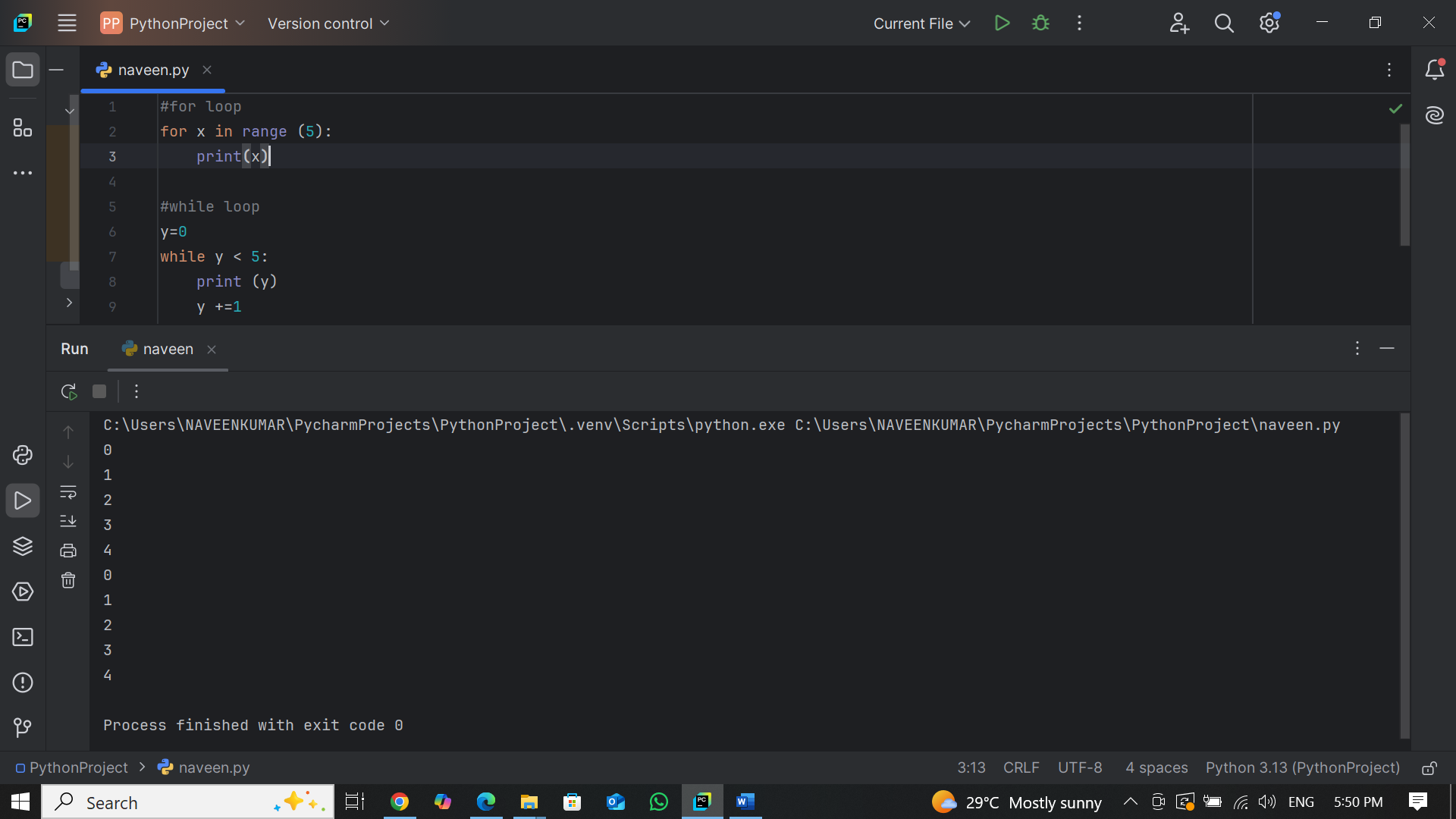
Loops allow repeated execution of code.

**4.1 For Loop**

Used to iterate over a sequence like a list or range.

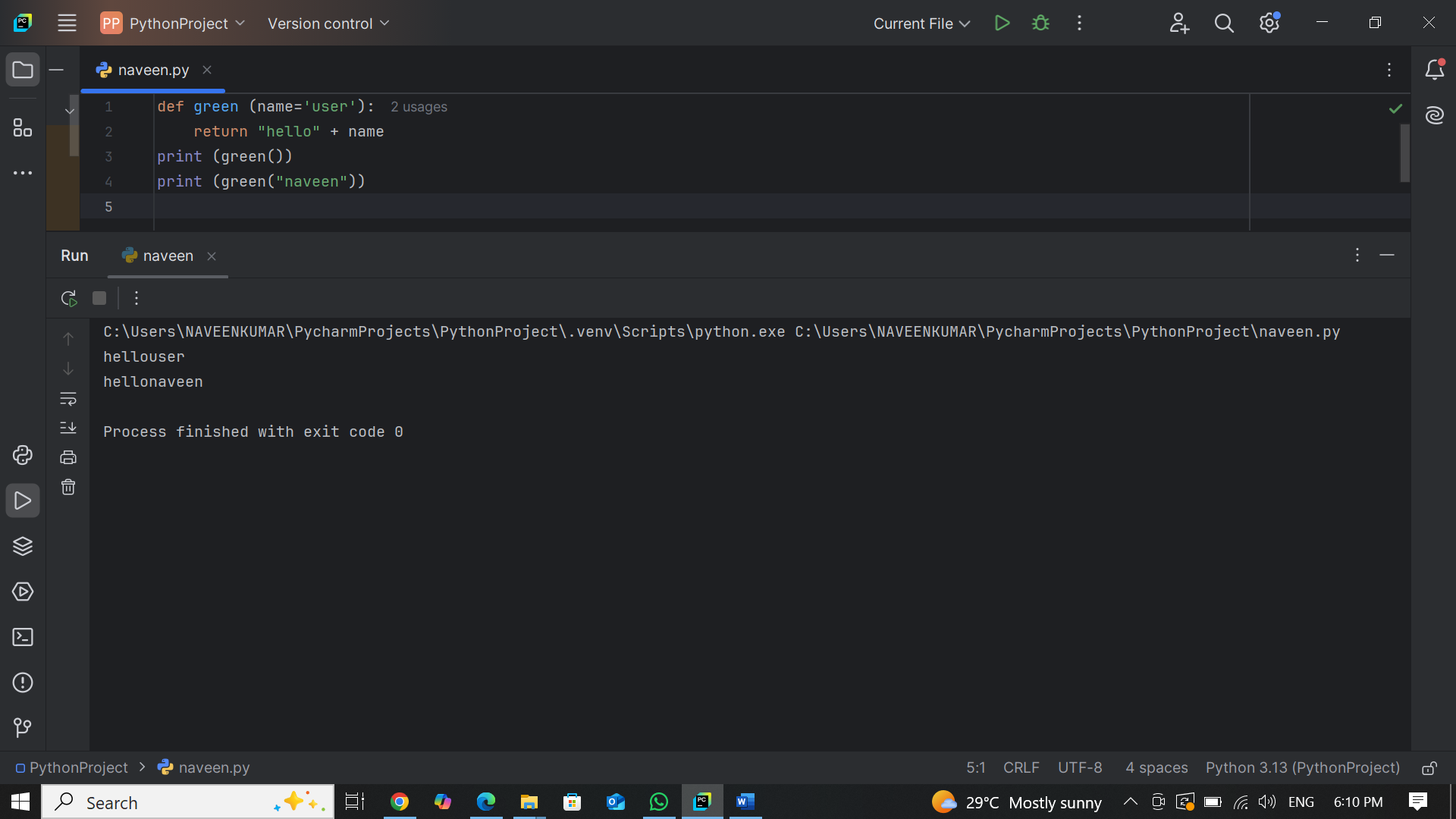
**4.2 While Loop**

Executes as long as a condition is true.'



**Functions in Python**

Functions help organize code into reusable blocks.



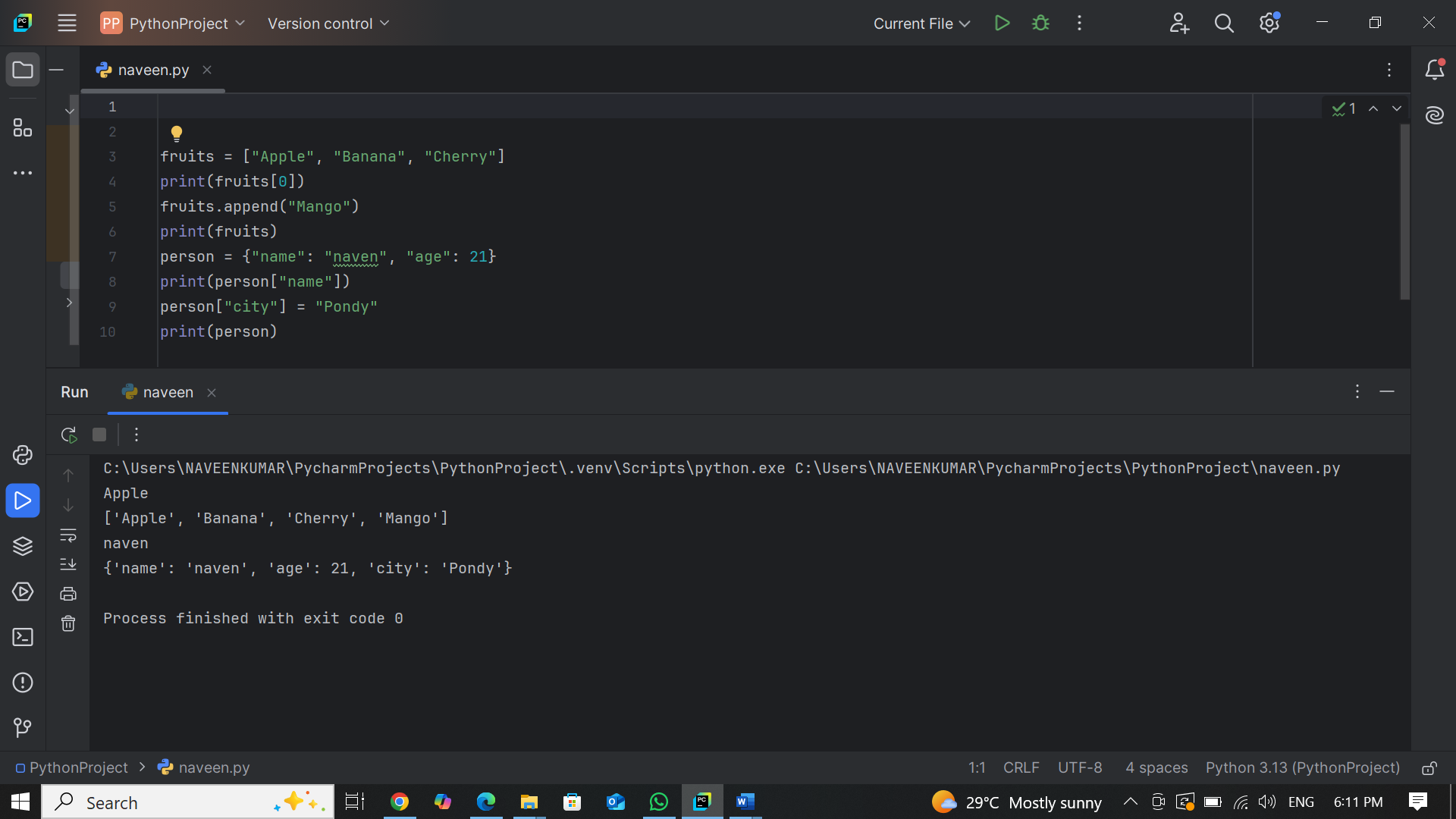
**Working with Lists & Dictionaries**

**Lists**

Lists store multiple values and support indexing.

**Dictionaries**

Dictionaries store data in key-value pairs.



**COMPARISON TABLE**

|  |  |  |  |
| --- | --- | --- | --- |
| Feature | For Loop | While Loop | For-Each Loop |
| Best for | Fixed iterations | Unknown iterations (condition-based) | Iterating over collections |
| Control | Uses range() (numeric control) | Uses a condition (while condition:) | Iterates over elements directly |
| Use Case | Counting, looping fixed times | Looping until a condition changes | Lists, dictionaries, sets |

* **Use for loop** → When you know **how many times** you need to loop.
* **Use while loop** → When you **don’t know** how many times it will loop.
* **Use for-each loop** → When looping through **lists, tuples, or dictionaries** without needing an index.