Walkthrough: Working with a Fruit List

The Main Function

```
def main():
```

This defines the `main` function — the main block of code that will run when we start the program.

---

1. Create a List of Fruits

```
fruit_list = ['apple', 'banana', 'orange', 'grape', 'pineapple']
```

- This creates a list named `fruit_list` that holds **five fruits**.
- Lists are ordered collections, so the items stay in the order you added them.
- The list now looks like:
  `['apple', 'banana', 'orange', 'grape', 'pineapple']`

---

2. Find and Print the Length of the List

```
fruit_length = len(fruit_list)
print(fruit_length)
```

- `len(fruit_list)` returns how many items are in the list (which is 5 at this point).
- The number `5` will be printed.

---

3. Add a New Fruit to the End

```
fruit_list.append('mango')
```

- `.append()` adds a new item to the **end** of the list.
- Now the list becomes:
  `['apple', 'banana', 'orange', 'grape', 'pineapple', 'mango']`

---

4. Print a Slice of the List

```
for i in range(2,5):
    print(fruit_list[i])
```

- `range(2, 5)` creates the numbers 2, 3, and 4.
- The loop prints the items at **index 2, 3, and 4** from the list.

So this will print:

- `'orange'` (index 2)
- `'grape'` (index 3)
- `'pineapple'` (index 4)

Note: It does **not** print `'mango'`, because that's at index 5 and is not included in the loop range.

Entry Point

```
if __name__ == '__main__':
    main()
```

This checks if the file is being run directly, and if so, it calls the `main()` function to start the program.

Summary

What this program does:

Creates a list of five fruits
Prints how many items are in the list
Adds `'mango'` to the end
Prints only the fruits from index 2 to 4

This is a great example of working with lists:
You learn how to **create**, **modify**, and **access** parts of a list using both `append()` and loops with `range()`.