**Technical Design Document Template**

**Name:** Chris McCathern

**Date Created:** 08/20/2025

This program manages the pre-sale of cinema tickets. A total of 20 tickets are available, and each buyer can purchase up to 4 tickets. The program prompts each buyer for how many tickets they want, ensures that the request follows the rules, updates the remaining ticket count, and keeps track of how many buyers there were. Once all 20 tickets are sold, the program displays the total number of buyers.

The purpose of this program is to automate and control ticket sales so that:

No buyer purchases more than 4 tickets.

No more than 20 tickets are sold in total.

The seller knows how many buyers there were.

**Functions used in the Program (list in order as they are called):**

1. **Function Name:** get\_ticket\_request

**Description**: Asks the buyer how many tickets they want to purchase and ensures the request is valid.

**Parameters: remaining tickets (int)** the number of tickets left before this purchase.

**Variables: requested (int)** stores the number of tickets the buyer wants.

**Logical Steps:**

1. Prompt the user to show how many tickets they want.

2. Use an if statement to check that the requested number is between 1 and 4.

3. If the request is greater than the remaining tickets, adjust the request down to remaining tickets.

4. Return the number of tickets that the buyer will purchase**.**

**Returns:**

(int) Number of tickets the buyer will buy.

2. **Function Name:** process\_sale

**Description: Updates** the number of remaining tickets and counts how many buyers there are.

**Parameters: tickets\_to\_sell (int) the** number of tickets purchased by the current buyer. **Remaining\_tickets (int) the** number of tickets available before the sale**.**

**buyers (int)** current count of buyers**.**

**Variables:**

**1.new\_remaining (int)** tickets left after this sale.

**2.new\_buyers (int)** updated buyer count.

**Logical Steps:**

1. Subtract tickets\_to\_sell from remaining\_tickets.

2. Increase buyers by 1.

3. Return updated values.

**Returns:**

**(tuple: int, int) updated** remaining\_tickets and buyers.

**Logical Steps:**

**1.Set remaining\_tickets** = 20 and buyers = 0.

**2. Use a loop (while tickets remain):**

Call get\_ticket\_request(remaining\_tickets) to get how many tickets the buyer wants.

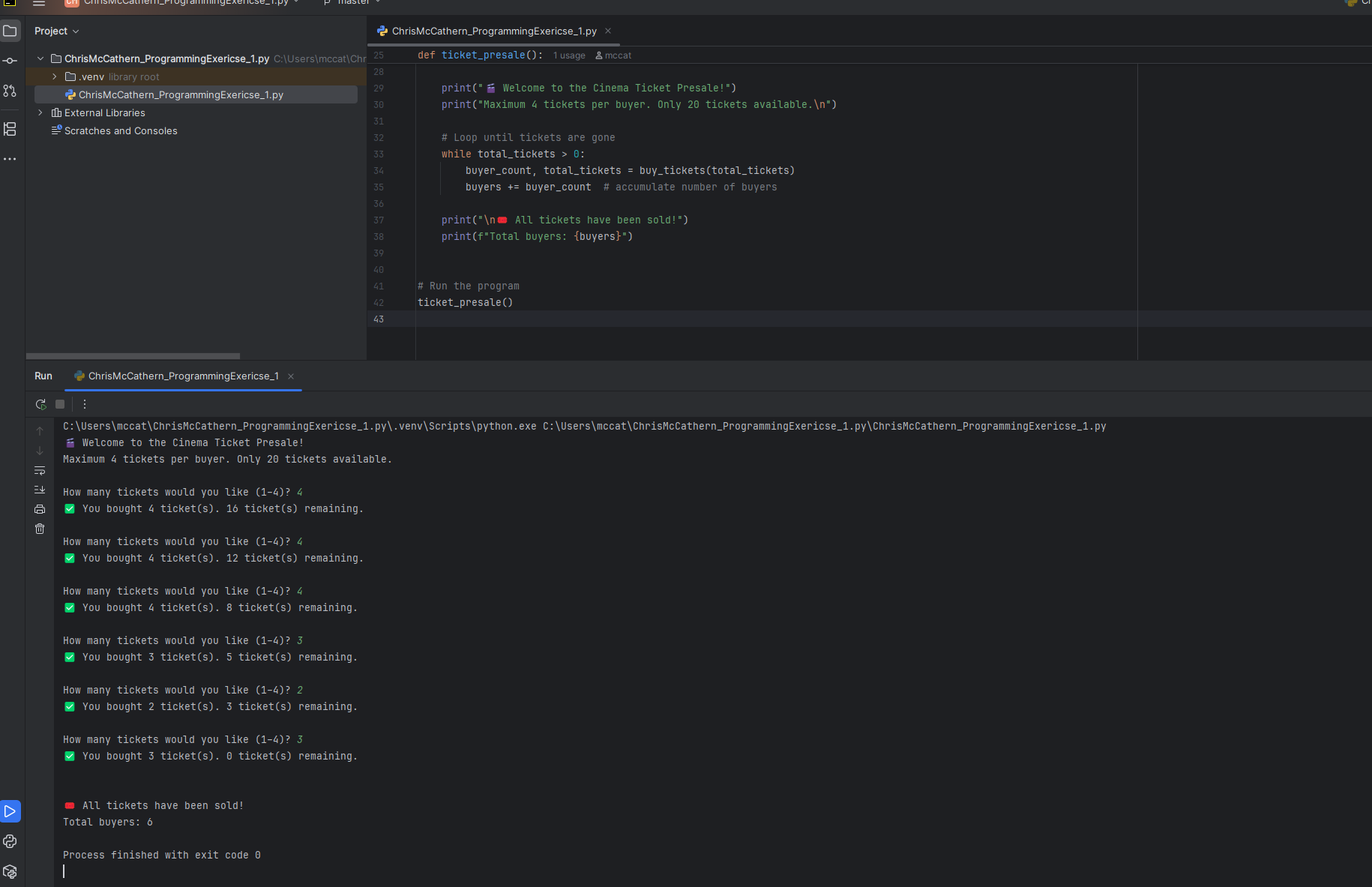
Call process\_sale(...) to update ticket count and buyer count.

Display how many tickets remain.

When remaining\_tickets reaches 0, exit the loop.

Output the total number of buyers.

**Link to your repository:** [cmac2371/ChrisMcCathern\_ProgrammingExericse\_1.py](https://github.com/cmac2371/ChrisMcCathern_ProgrammingExericse_1.py)



A black screen with white dots

AI-generated content may be incorrect.