

Jieun Lee

Prof. Esteban Morales

CS3A

11/12/19

Theater Ticket Booth

The program simulates selling movie tickets and earns money from customers.

Project Description

The project will use every concept I learned from this course. It includes loops, functions, lists, files, exception handling, dictionaries, and classes. The program has six parts main class with Tkinter, theater, movie, cart, counter. The class with Tkinter create and write component of GUI. It creates many different functions so that user can interface with the program properly. Theater class basically getting information about movie from Movie class. The movie class shows all the screening movies' titles, posters, available times, seats, and length. There is an existed local movie.txt file that has all the movie's details such as titles, available times. The program reads the text file and will show information about each movie to the window. For the posters, the program will get images local storage. By purchasing tickets from the user, the information such as remaining seats will be updated to the text files. In the cart section, the movie user selected from the movie section will appear in the cart. Two tasks are in cart sections which are a receipt, no receipt. In the counter section, the money earned from selling tickets will goes into the counter. When the counter is closed, the total money earned and movie sold will be on the counter.txt file. Users will interact with the program by clicking the buttons. The purpose of this project is simulating ticket booth using python.

Project Use Case

- Tkinter class

This section, I implemented several functions from Tkinter. Making buttons, labels, text, etc. made from this step. It will show main window at the first, and user will pick movie by clicking buttons with specific time and seat. There are two frames movie frame and receipt, counter frame.

- Theater class

Getting title, show times, number of seats from reading text file. And save it into Movie object and do the basic task such as get title, get row, update seat calling functions from Movie class.

- Movie class

The user chooses the movie by clicking a specific time and movie. There are buttons with available times with a number of remaining seats. The user picks a time, and the program shows available seats. It will change the color of the seats clicked. The selected seats will be colored when the next user sees left seats and they are unavailable to click.

- Cart class

There is another part of the cart in the window. After the user picked a movie with selected seats, it will show the title, time, quantity, and price in the cart. If the user clicked the receipt button, it saves the information into receipt.txt file and adds subtotal to the counter money. If the user clicked no receipt button, it just adds subtotal to the counter money.

- Counter class

The other part is a counter. Counter collects all the money and saves what movies and how many tickets sold today. Then it will save the invoice to the counter.txt. Also, it will store movies title with the number of tickets sold.

Concepts Used

It has five classes Tkinter, Theater, movie, cart, and counter. The program uses text files for each part to read or write information about the movie. Title – string, times – list, length – int, seats – 2d list. I will use 2d lists to manage the seats. Instead of store times with seats into the dictionary, I decided to make a Movie class manage all together. In the Movie class, it manages seats with each screen time using a dictionary. The key is list of times and the value is 2d list seats. There are few constant global variables such as filename, number of seats. The customer class is the parent class of Cart Class. Cart class inherits attributes in the Customer class. In the Counter class, I will use a dictionary to store key – movies, and value – the number of tickets sold. every class has getters methods and some of the classes have setters. In Tkinter class, I used buttons, labels, text, frame etc. I have learned in the class. For the seat buttons, I managed them by creating list of buttons. I made them change color when the seat is clicked. In Theater class, I created list of objects(Movies) and and store each movie's data into each Movie class and mange them using list.

App
<div><div>_theater = object</div><div>_counter = object</div><div>_cart = object</div><div>extra components (buttons ,labels...)</div></div>
<div><div>create_movie_component()</div><div>update()</div><div>pass_to_counter()</div><div>clear_cart_with_total()</div><div>make_receipt()</div><div>pass_to_cart()</div><div>new_window()</div></div>

Theater
<div><div>file_name = string()</div><div>movie = list of objects</div></div>
<div><div>movies_from_file()</div><div>get_num_movies()</div><div>get_titles()</div><div>get_movie_num_w_title()</div><div>more getters()</div><div>check_seat()</div><div>update_seat()</div></div>

Movie
<div><div>_title = string</div><div>_length = int</div><div>_time_seats = dict()</div><div>_seats = list()</div></div>
<div><div>_show_times()</div><div>_show_seats(time)</div><div>_update_seats(time,r,w)</div><div>_get_title()</div><div>_get_num_of_left_seats(tim</div></div>

Cart(Customer)
<div><div>_title = string</div><div>_time = int</div><div>_seats = list()</div></div>
<div><div>receipt()</div><div>clear()</div><div>getters</div><div>setters</div></div>

Counter
<div><div>_file_name = file_name</div><div>_total = int</div><div>_tickets_sold = tickets_sold</div><div>_movies = dict()</div></div>
<div><div>add(title,tickets_sold)</div><div>update()</div><div>get_total()</div><div>get_movies()</div></div>

Timeline

- Week 12 - I just finished writing pseudocode in this week. I will finish writing code which can do all the basic tasks.
- Week 13 - I will add more stuff that I learned from week 13 to the program. If I can use storing and searching method into my program I will use those methods. From this point, I will work on the GUI and will finish the project by week 14 during the thanksgiving.
- After week13 – I spent time to figure out how to utilize Tkinter and implement it into my program.