**MACHINE PROBLEM**

**PROGRAM DESCRIPTION**

A new video store in your neighborhood is about to open. However, it does not have a program to keep track of its videos and customers. The store managers want someone to write a program for their system so that the video store can operate.

The program will require you to design **2 ADTs** as described below:

|  |  |
| --- | --- |
| **[1] VIDEO ADT** | |
| **Data** | **Operations** |
| Video\_ID (preferably int, auto-generated)  Movie Title  Genre  Production  Number of Copies  Movie Image Filename | [1] Insert a new video  [2] Rent a video; that is, check out a video  [3] Return a video, or check in, a video  [4] Show the details of a particular video  [5] Display all videos in the store  [6] Check whether a particular video is in the store |

|  |  |
| --- | --- |
| **[2] CUSTOMER PARENT ADT** | |
| **Data** | **Operations** |
| Customer\_ID (preferably int, auto-generated)  Name  Address | [1] Add Customer  [2] Show the customer details  [3] Print list of all customers |
| **[3] CUSTOMER-RENT CHILD ADT** | |
| Customer\_ID (  Video\_ID (of all rented videos of a customer) | [1] Rent a Video (Add to the video ids rented)  [2] Return a Video (Remove the video id from the list)  [3] Print list of all videos rented by each customer |

The program will require you to maintain **3 text files** with specifications below:

|  |  |
| --- | --- |
| **Text File** | **Requirements** |
| **[1] VIDEO Text File** | [1] Will store the information about the Videos  [2] Should contain the following by default:  5 Horror Movies  5 Romance Movies  5 Sci-Fi Movies  5 Action Movies  5 Comedy Movies  ***\*\*Must be authentic and true.\*\**** |

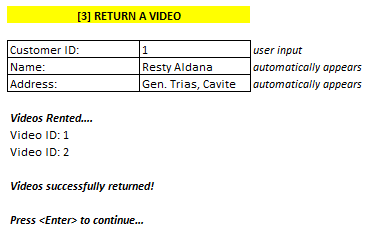
|  |  |
| --- | --- |
| **[2] CUSTOMER Text File** | [1] Will store the basic information about the Customers  [2] Should contain at least 10 customers by default  ***\*\*Must be authentic and true.\*\**** |
| **[3] CUSTOMER-RENT** | [1] Will store all customers that RENT a VIDEO  [2] Will store all the Video\_IDs of all rented videos  [3] Will delete Video\_IDs and Customer\_ID when videos are returned |

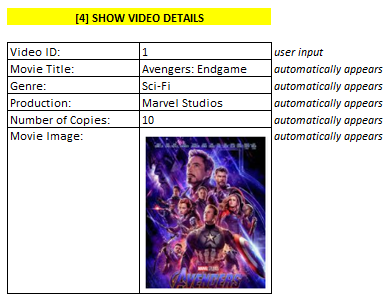
**NOTE: IF YOU FIND THE NEED TO ADD OR MAKE ALTERATIONS TO THE SPECIFICATIONS OF THE ADT AND TEXT FILES ABOVE, PLEASE DO SO. JUST MAKE SURE THAT YOU HAVE AN ACCEPTABLE LOGICAL REASON.**

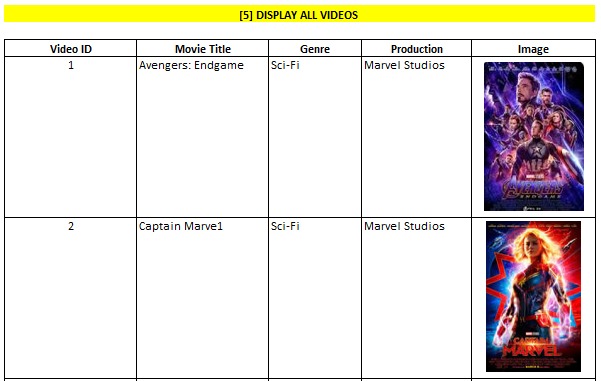
**INSTRUCTIONS:**

* The program will have the following options/menus:
  + **[1] New Video**
  + **[2] Rent a Video //**
  + **[3] Return a Video //**
  + **[4] Show Video Details**
  + **[5] Display all Videos**
  + **[6] Check Video Availability**
  + **[7] Customer Maintenance**
    - **[1] Add New Customer**
    - **[2] Show Customer Details**
    - **[3] List of Videos Rented by a Customer //**
  + **[8] Exit Program**
* Follow the suggested screen dialogs below for each of the option above:

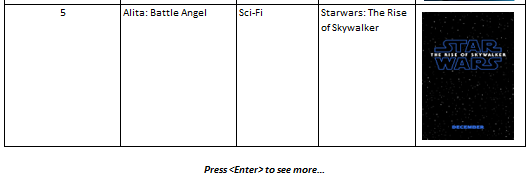


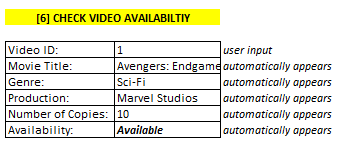


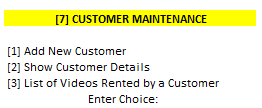


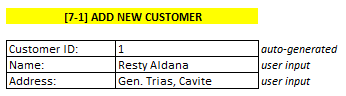


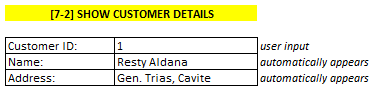


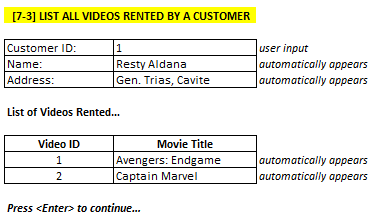












**OTHER REQUIREMENTS:**

* You must store your videos in a **linked list** when you retrieved them from the text file. They must also be stored in a linked list data structure during processing. Saving back to the text file will be done when the user chooses [8] Exit Program.
* You must store you customers in a **queue 🡪 linked list** when you retrieved them from the text file. They must also be stored in a queue during processing. Saving back to the text file will be done when the user chooses [8] Exit Program.
* Rented videos will be stored in a **stack** and will be saved in the CUSTOMER-RENT text file when the user chooses [8] Exit Program.
* Use **data structures, files, functions, ADTs, STLs and algorithms** in your program**.** Use **STLs** inside your **ADTs**.
* Use existing and appropriate **sorting** and **searching** algorithms.
* Put **necessary comments** to your program.
* **No global declarations**.
* Provide error messages whenever necessary.