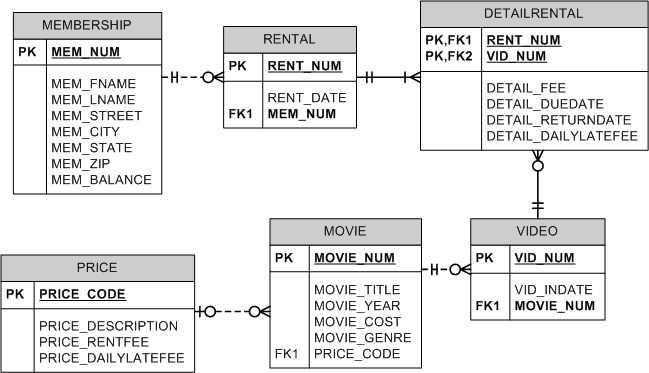
**HW 3**

CASES

**EliteVideo is a startup company providing concierge DVD kiosk service in upscale neighborhoods. EliteVideo can own several copies (VIDEO) of each movie (MOVIE). For example, the store may have 10 copies of the movie “Twist in the Wind”. “Twist in the Wind” would be one MOVIE and each copy would be a VIDEO. A rental transaction (RENTAL) involves one or more videos being rented to a member (MEMBERSHIP). A video can be rented many times over its lifetime, therefore, there is a M:N relationship between RENTAL and VIDEO. DETAILRENTAL is the bridge table to resolve this relationship. The complete ERD is provided in Figure P7.65.**

Figure P7.65 EliteVideo ERD



1. **Write the SQL code to create the table structures for the entities shown in Figure P7.65. The structures should contain the attributes specified in the ERD. Use data types that would be appropriate for the data that will need to be stored in each attribute. Enforce primary key and foreign key constraints as indicated by the ERD.**
2. **The following tables provide a very small portion of the data that will be kept in the database. This data needs to be inserted into the database for testing purposes. Write the INSERT commands necessary to place the following data in the tables that were created in problem 65.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **MEMBERSHIP** | | | | | | | |
| **Mem\_**  **Num** | **Mem\_**  **Fname** | **Mem\_**  **Lname** | **Mem\_Street** | **Mem\_City** | **Mem\_**  **State** | **Mem\_Zip** | **Mem\_**  **Balance** |
| 102 | Tami | Dawson | 2632 Takli Circle | Norene | TN | 37136 | 11 |
| 103 | Curt | Knight | 4025 Cornell Court | Flatgap | KY | 41219 | 6 |
| 104 | Jamal | Melendez | 788 East 145th Avenue | Quebeck | TN | 38579 | 0 |
| 105 | Iva | Mcclain | 6045 Musket Ball Circle | Summit | KY | 42783 | 15 |
| 106 | Miranda | Parks | 4469 Maxwell Place | Germantown | TN | 38183 | 0 |
| 107 | Rosario | Elliott | 7578 Danner Avenue | Columbia | TN | 38402 | 5 |
| 108 | Mattie | Guy | 4390 Evergreen Street | Lily | KY | 40740 | 0 |
| 109 | Clint | Ochoa | 1711 Elm Street | Greeneville | TN | 37745 | 10 |
| 110 | Lewis | Rosales | 4524 Southwind Circle | Counce | TN | 38326 | 0 |
| 111 | Stacy | Mann | 2789 East Cook Avenue | Murfreesboro | TN | 37132 | 8 |
| 112 | Luis | Trujillo | 7267 Melvin Avenue | Heiskell | TN | 37754 | 3 |
| 113 | Minnie | Gonzales | 6430 Vasili Drive | Williston | TN | 38076 | 0 |

|  |  |  |
| --- | --- | --- |
| **RENTAL** | | |
| **Rent\_Num** | **Rent\_Date** | **Mem\_Num** |
| 1001 | 01-MAR-13 | 103 |
| 1002 | 01-MAR-13 | 105 |
| 1003 | 02-MAR-13 | 102 |
| 1004 | 02-MAR-13 | 110 |
| 1005 | 02-MAR-13 | 111 |
| 1006 | 02-MAR-13 | 107 |
| 1007 | 02-MAR-13 | 104 |
| 1008 | 03-MAR-13 | 105 |
| 1009 | 03-MAR-13 | 111 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **DETAILRENTAL** | | | | | |
| **Rent\_Num** | **Vid\_Num** | **Detail\_Fee** | **Detail\_Duedate** | **Detail\_Returndate** | **Detail\_Dailylatefee** |
| 1001 | 34342 | 2 | 04-MAR-13 | 02-MAR-13 |  |
| 1001 | 61353 | 2 | 04-MAR-13 | 03-MAR-13 | 1 |
| 1002 | 59237 | 3.5 | 04-MAR-13 | 04-MAR-13 | 3 |
| 1003 | 54325 | 3.5 | 04-MAR-13 | 09-MAR-13 | 3 |
| 1003 | 61369 | 2 | 06-MAR-13 | 09-MAR-13 | 1 |
| 1003 | 61388 | 0 | 06-MAR-13 | 09-MAR-13 | 1 |
| 1004 | 44392 | 3.5 | 05-MAR-13 | 07-MAR-13 | 3 |
| 1004 | 34367 | 3.5 | 05-MAR-13 | 07-MAR-13 | 3 |
| 1004 | 34341 | 2 | 07-MAR-13 | 07-MAR-13 | 1 |
| 1005 | 34342 | 2 | 07-MAR-13 | 05-MAR-13 | 1 |
| 1005 | 44397 | 3.5 | 05-MAR-13 | 05-MAR-13 | 3 |
| 1006 | 34366 | 3.5 | 05-MAR-13 | 04-MAR-13 | 3 |
| 1006 | 61367 | 2 | 07-MAR-13 |  | 1 |
| 1007 | 34368 | 3.5 | 05-MAR-13 |  | 3 |
| 1008 | 34369 | 3.5 | 05-MAR-13 | 05-MAR-13 | 3 |
| 1009 | 54324 | 3.5 | 05-MAR-13 |  | 3 |
| 1001 | 34366 | 3.5 | 04-MAR-13 | 02-MAR-13 | 3 |

|  |  |  |
| --- | --- | --- |
| **VIDEO** | | |
| **Vid\_Num** | **Vid\_Indate** | **Movie\_Num** |
| 54321 | 18-JUN-12 | 1234 |
| 54324 | 18-JUN-12 | 1234 |
| 54325 | 18-JUN-11 | 1234 |
| 34341 | 22-JAN-11 | 1235 |
| 34342 | 22-JAN-11 | 1235 |
| 34366 | 02-MAR-13 | 1236 |
| 34367 | 02-MAR-13 | 1236 |
| 34368 | 02-MAR-13 | 1236 |
| 34369 | 02-MAR-13 | 1236 |
| 44392 | 21-OCT-12 | 1237 |
| 44397 | 21-OCT-12 | 1237 |
| 59237 | 14-FEB-13 | 1237 |
| 61388 | 25-JAN-11 | 1239 |
| 61353 | 28-JAN-10 | 1245 |
| 61354 | 28-JAN-10 | 1245 |
| 61367 | 30-JUL-12 | 1246 |
| 61369 | 30-JUL-12 | 1246 |

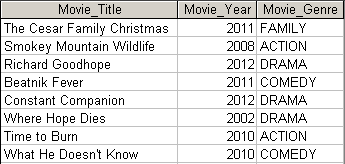
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **MOVIE** | | | | | |
| **Movie\_Num** | **Movie\_Name** | **Movie\_Year** | **Movie\_Cost** | **Movie\_Genre** | **Price\_Code** |
| 1234 | The Cesar Family Christmas | 2011 | 39.95 | FAMILY | 2 |
| 1235 | Smokey Mountain Wildlife | 2008 | 59.95 | ACTION | 1 |
| 1236 | Richard Goodhope | 2012 | 59.95 | DRAMA | 2 |
| 1237 | Beatnik Fever | 2011 | 29.95 | COMEDY | 2 |
| 1238 | Constant Companion | 2012 | 89.95 | DRAMA | 2 |
| 1239 | Where Hope Dies | 2002 | 25.49 | DRAMA | 3 |
| 1245 | Time to Burn | 2009 | 45.49 | ACTION | 1 |
| 1246 | What He Doesn't Know | 2010 | 58.29 | COMEDY | 1 |

|  |  |  |  |
| --- | --- | --- | --- |
| **PRICE** | | | |
| **Price\_Code** | **Price\_Description** | **Price\_Rentfee** | **Price\_Dailylatefee** |
| 1 | Standard | 2 | 1 |
| 2 | New Release | 3.5 | 3 |
| 3 | Discount | 1.5 | 1 |
| 4 | Weekly Special | 1 | .5 |

**For questions 66 – 97, use the tables that were created in Problem 64 and the data that was loaded into those tables in Problem 65.**

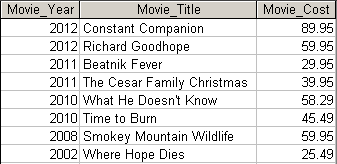
1. **Write the SQL command to save the rows inserted in Problem 66. (skip this question)**
2. **Write the SQL command to change the movie year for movie number 1245 to 2010.**
3. **Write the SQL command to change the price code for all Action movies to price code 3.**
4. **Write a single SQL command to increase all price rental fee values by $0.50.**
5. **Write the SQL command to save the changes made to the PRICE and MOVIE tables in Problems 67 – 70. (skip this question)**
6. **Write a query to display the movie title, movie year, and movie genre for all movies (result shown in Figure P7.72).**

Figure P7.72 All Movies



1. **Write a query to display the movie year, movie title, and movie cost sorted by movie year in descending order (result shown in Figure P7.73).**

Figure P7.73 Movies by year



1. **Write a query to display the movie title, movie year, and movie genre for all movies sorted by movie genre in ascending order, then sorted by movie year in descending order within genre (result shown in Figure P7.74).**

Figure P7.74 Movies with multicolumn sort



1. **Write a query to display the movie number, movie title, and price code for all movies with a title that starts with the letter “R” (result shown in Figure P7.75).**

Figure P7.75 Movies starting with R



1. **Write a query to display the movie title, movie year, and movie cost for all movies that contain the word “hope” anywhere in the title. Sort the results in ascending order by title (result shown in figure P7.76).**

Figure P7.76 Movies with “Hope” in the title



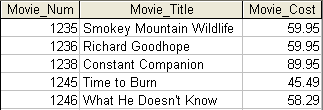
1. **Write a query to display the movie title, movie year, and movie genre for all action movies (result shown in Figure P7.77).**

Figure P7.77 Action movies



1. **Write a query to display the movie number, movie title, and movie cost for all movies with a cost greater than $40 (result shown in Figure P7.78).**

P7.78 Movies costing more than $40



1. **Write a query to display the movie number, movie title, movie cost, and movie genre for all movies that are either action or comedy movies and that have a cost that is less than $50. Sort the results in ascending order by genre. (Result shown in Figure P7.79.)**

Figure P7.79 Action or comedy movies costing less than $50



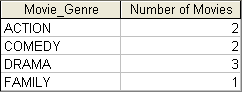
1. **Write a query to display the movie number, and movie description for all movies where the movie description is a combination of the movie title, movie year and movie genre with the movie year enclosed in parentheses (result shown in Figure P7.80).**

Figure P7.80 Movies with concatenated descriptions



1. **Write a query to display the movie genre and the number of movies in each genre (result shown in Figure P7.81).**

Figure P7.81 Number of movies in genre

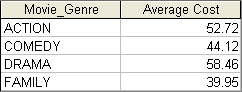


1. **Write a query to display the average cost of all of the movies (result shown in Figure P7.82).**

Figure P7.82 Average movie cost 

1. **Write a query to display the movie genre and average cost of movies in each genre (result shown in Figure P7.83).**

Figure P7.83 Average movie cost by genre



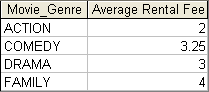
1. **Write a query to display the movie title, movie genre, price description, and price rental fee for all movies with a price code (result shown in Figure P7.84).**

Figure P7.84 Rental fees for movies



1. **Write a query to display the movie genre and average price rental fee for movies in each genre that have a price (result shown in Figure P7.85).**

Figure P7.85 Average rental fee by genre



1. **Write a query to display the movie title, movie year, and the movie cost divided by the price rental fee for each movie that has a price to determine the number of rentals it will take to break even on the purchase of the movie (result shown in Figure P7.86).**

Figure P7.86 Breakeven rentals



1. **Write a query to display the movie title and movie year for all movies that have a price code (result shown in Figure P7.87).**

P7.87 Movies with a price



1. **Write a query to display the movie title, movie year, and movie cost for all movies that have a cost between $44.99 and $49.99 (result shown in Figure P7.88).**

Figure P7.88 Movies costs within a range



1. **Write a query to display the movie title, movie year, price description, and price rental fee for all movies that are in the genres Family, Comedy, or Drama (result shown in Figure P7.89).**

Figure P7.89 Movies with specific genres



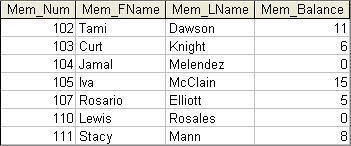
1. **Write a query to display the movie number, movie title, and movie year for all movies that do not have a video (result shown in Figure P7.90).**

Figure P7.90 Movies without videos



1. **Write a query to display the membership number, first name, last name, and balance of the memberships that have a rental (result shown in Figure P7.91).**

Figure P7.91 Balances of memberships with rentals



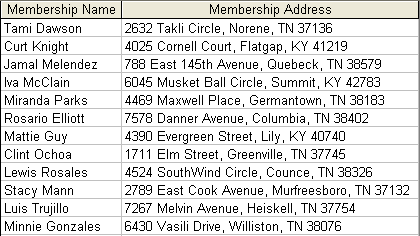
1. **Write a query to display the minimum balance, maximum balance, and average balance for memberships that have a rental (result shown in Figure P7.92).**

Figure P7.92 Minimum, maximum, and average balances



1. **Write a query to display the membership name (concatenate the first name and last name with a space between them into a single column), membership address (concatenate the street, city, state, and zip codes into a single column with spaces (result shown in Figure P7.93).**

Figure P7.93 Concatenated membership data



1. **Write a query to display the rental number, rental date, video number, movie title, due date, and return date for all videos that were returned after the due date. Sort the results by rental number and movie title (result shown in Figure P7.94).**

Figure P7.94 Late video returns



1. **Write a query to display the rental number, rental date, video number, movie title, due date, return date, detail fee, and number of days past the due date that the video was returned for each video that was returned after the due date. Sort the results by rental number and movie title. (Result shown in Figure P7.95.)**

Figure P7.95 Number of days late



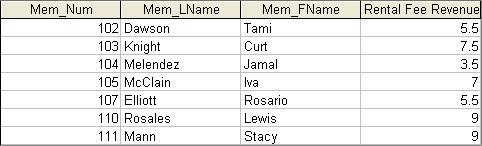
1. **Write a query to display the rental number, rental date, movie title, and detail fee for each movie that was returned on or before the due date (result shown in Figure P7.96).**

Figure P7.96 Actual rental fees charged



1. **Write a query to display the membership number, last name, and total rental fees earned from that membership (result shown in Figure P7.97). The total rental fee is the sum of all of the detail fees (without the late fees) from all movies that the membership has rented.**

Figure P7.97 Total rental fees paid by membership



1. **Write a query to display the movie number, movie genre, average movie cost of movies in that genre, movie cost of that individual movie, and the percentage difference between the average movie cost and the individual movie cost (result shown in Figure P7.98). Note: the percentage difference is calculated as the cost of the individual movie minus the average cost of movies in that genre, divided by the average cost of movies in that genre multiplied by 100. For example, if the average cost of movies in the “Family” genre is $25, if a given Family movie cost $26, then the calculation would be ((26 – 25) / 25 \* 100), which would work out to be 4.00%. This indicates that this movie costs 4% more than the average Family movie.**

Figure P7.98 Movie difference from genre average

