ARDA ANDIRIN-17921934994

EFE BERK ERGÜLEÇ-24191130360

HALİL ÖMER TEKİN-15430127586

TABLE OF CONTENTS

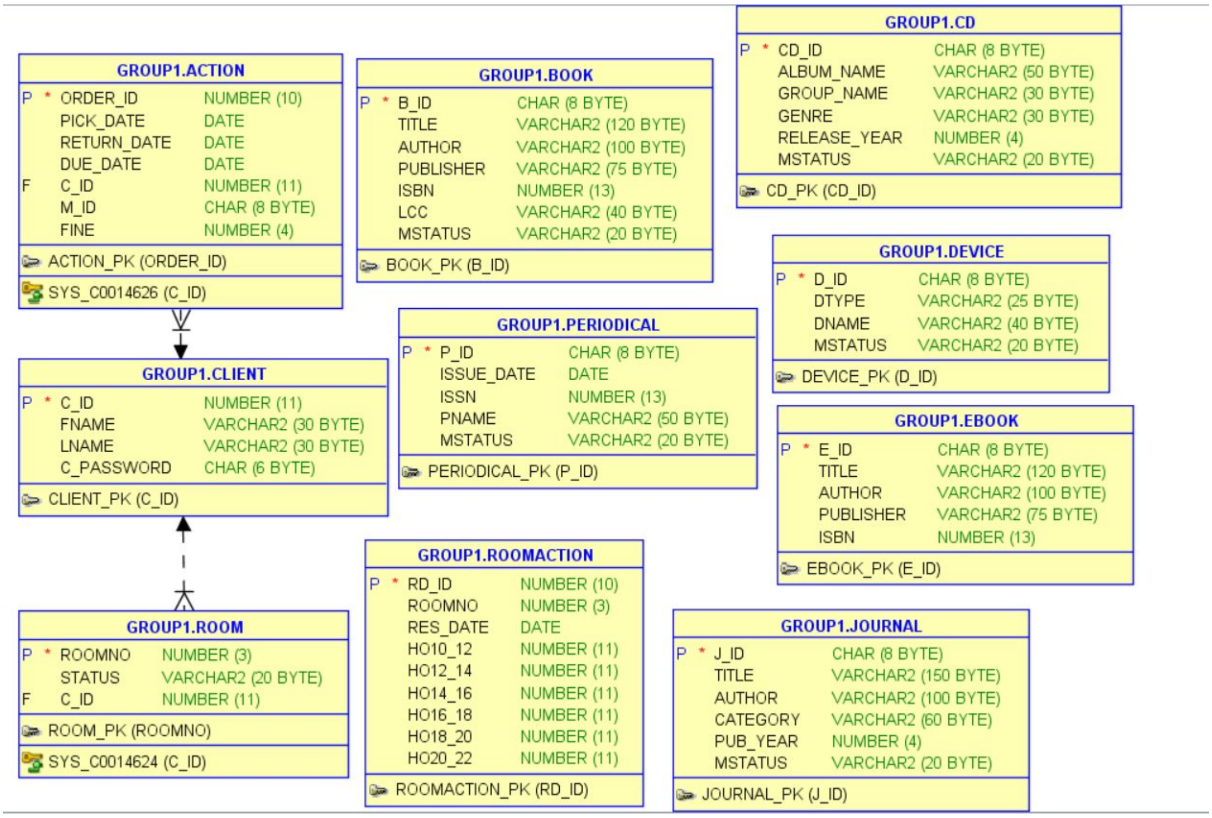
BRIEF INTRO

Before explanations and descriptions, we want to say that our project is a library management system for not only for users, but also a superuser called “Admin”. At this system our main goal is to make common activities with both users and Admin. Activities and their meanings will be explained in the progressive steps. User at our database can show its personal informations such as his/her name, also users can look at available materials in the library, they can borrow materials, view materials that they hold, make payment to pay all previous fines and lastly they can reserve rooms. Admin can show users’ informations, add user, show borrowed items show rooms, and add,delete or update materials. It’s precedence is more than normal user, so it’s operations are more general. Details will explain below.

UPDATES AND CHANGES AT PHASE 2

- At phase 1, we kept all materials together. However in phase 2, we thought that it isn’t the efficient way to keep data like that, so we separate Material entity set into specific entity sets like Book, Device, etc.
- Library entity set is unneeded, so we removed it.
- We changed Person entity set to Client, and we added password to it.
- We added new entity sets. These are Action, Book, CD, Device, Ebook, Journal, Periodical and RoomAction.

DESCRIPTIONS OF TABLES



GROUP1.ACTION

P	*	ORDER_ID	NUMBER (10)
		PICK_DATE	DATE
		RETURN_DATE	DATE
		DUE_DATE	DATE
F		C_ID	NUMBER (11)
		M_ID	CHAR (8 BYTE)
		FINE	NUMBER (4)

ACTION_PK (ORDER_ID)

SYS_C0014626 (C_ID)

GROUP1.BOOK

P	*	B_ID	CHAR (8 BYTE)
		TITLE	VARCHAR2 (120 BYTE)
		AUTHOR	VARCHAR2 (100 BYTE)
		PUBLISHER	VARCHAR2 (75 BYTE)
		ISBN	NUMBER (13)
		LCC	VARCHAR2 (40 BYTE)
		MSTATUS	VARCHAR2 (20 BYTE)

BOOK_PK (B_ID)

GROUP1.CD

P	*	CD_ID	CHAR (8 BYTE)
		ALBUM_NAME	VARCHAR2 (50 BYTE)
		GROUP_NAME	VARCHAR2 (30 BYTE)
		GENRE	VARCHAR2 (30 BYTE)
		RELEASE_YEAR	NUMBER (4)
		MSTATUS	VARCHAR2 (20 BYTE)

CD_PK (CD_ID)

GROUP1.DEVICE

P	*	D_ID	CHAR (8 BYTE)
		DTYPE	VARCHAR2 (25 BYTE)
		DNAME	VARCHAR2 (40 BYTE)
		MSTATUS	VARCHAR2 (20 BYTE)

DEVICE_PK (D_ID)

GROUP1.CLIENT

P	*	C_ID	NUMBER (11)
		FNAME	VARCHAR2 (30 BYTE)
		LNAME	VARCHAR2 (30 BYTE)
		C_PASSWORD	CHAR (6 BYTE)

CLIENT_PK (C_ID)

GROUP1.PERIODICAL

P	*	P_ID	CHAR (8 BYTE)
		ISSUE_DATE	DATE
		ISSN	NUMBER (13)
		PNAME	VARCHAR2 (50 BYTE)
		MSTATUS	VARCHAR2 (20 BYTE)

PERIODICAL_PK (P_ID)

GROUP1.EBOOK

P	*	E_ID	CHAR (8 BYTE)
		TITLE	VARCHAR2 (120 BYTE)
		AUTHOR	VARCHAR2 (100 BYTE)
		PUBLISHER	VARCHAR2 (75 BYTE)
		ISBN	NUMBER (13)

EBOOK_PK (E_ID)

GROUP1.ROOMACTION

P	*	RD_ID	NUMBER (10)
		ROOMNO	NUMBER (3)
		RES_DATE	DATE
		HO10_12	NUMBER (11)
		HO12_14	NUMBER (11)
		HO14_16	NUMBER (11)
		HO16_18	NUMBER (11)
		HO18_20	NUMBER (11)
		HO20_22	NUMBER (11)

ROOMACTION_PK (RD_ID)

GROUP1.ROOM

P	*	ROOMNO	NUMBER (3)
		STATUS	VARCHAR2 (20 BYTE)
F		C_ID	NUMBER (11)

ROOM_PK (ROOMNO)

SYS_C0014624 (C_ID)

GROUP1.JOURNAL

P	*	J_ID	CHAR (8 BYTE)
		TITLE	VARCHAR2 (150 BYTE)
		AUTHOR	VARCHAR2 (100 BYTE)
		CATEGORY	VARCHAR2 (60 BYTE)
		PUB_YEAR	NUMBER (4)
		MSTATUS	VARCHAR2 (20 BYTE)

JOURNAL_PK (J_ID)

Action: This table keeps the borrowing actions of a library material (Book / CD / Device etc.).Every action has a pick date, due date and the date returned. Thanks to these three dates we can follow a material's situation, we can calculate a client's fee and can follow payments. An action has a primary key Order_ID, a foreign key Client_ID of the Client who borrowed the material and the material_ID of the borrowed material.

Book: This table is for the hard-copy books in the library. A book has a primary key B_ID consisting of a string beginning with B and this key ends with the copy number of the book such as B0001-01. Every book has a title, an author, a publisher, ISBN, Library of Congress Classification code that is to find the shelf easily according to book's subject. The status is also can be accessed by this table.

CD: In this table, we keep the musical records in the library. It has CD_ID such as C00001-01 which is similar to Book Id. CD has an album name, group_name as its artist name, the year of its release, its genre and the status.

Client: This table keeps the data of the clients of the library. Every client has C_ID which is equal to national ID number, a password given, first and last name. We can reach their borrowed materials and calculate the debt from Action table through queries.

Device: This table consists of the devices in the library such as PC, tablet or calculator that can be borrowed by clients. A device has a type, name, Device ID like D00001-01 similar to book id and its status.

Ebook: This table keeps data of electronic books in library. It has E_ID, title, author, publisher and ISBN.

Journal: This table keeps data of academic journals in library. It has J_ID like J00001-01, title, author, category, publication year and the status.

Periodical: This table contains newspapers and magazines in library. It has P_ID like P00001-01, name, issue date, ISSN and status.

Room: This table keeps the data of the study rooms and their status. Every room has a number. We can assign them users for periods of time and we can follow their status.

RoomAction: This table keeps the reservation records of rooms with their dates. Every RoomAction has an RD_ID and date. There are 6 reservation time intervals per day, each having length of 2 hours. These hours are attached to Client IDs when reserved.

EXPLANATION OF JAVA APPLICATION

In our Java application, we designed a console-based app. At first, our program wants user to enter his/her id and password. There are 2 different user type. One of it is user enter. At user enter, if a person is in Client table and it's password is matching with itself, it can enter system. This user type cannot add or remove data, but it can borrow something from library. Other user type is super user called "Admin". At Admin mode, you can see and edit all of the data in the system.

Our interface is build at console and it's commands are given with numbers at PC's keyboard. You can apply commands by entering numbers which represents at console. We created 3 java classes: Driver, UserMenu and AdminMenu. Driver is our main class. It includes login screen and paths which goes to Admin and users. You have 2 choices: enter as Admin or enter as user. If you want to enter as Admin, you need to enter ID as "Admin"(it's case-insensitive, so you can enter whatever type you want) and password as "cmpe232"(it's case-insensitive). On the other side, if you want to enter as user, you need to know your id and your password. If you want to enter as user, we put an id and a password here. “23982758947” is id and “dblect” is password for our instructor Orkunt Sabuncu. When you type this id and password, you can enter your specific account.

There are 2 different menu in our application. One of them is owned by user and it’s completely specific. Other one is owned by Admin and only Admin can see this menu. We wrote login screen to Driver.java. Admin’s menu and actions are in AdminMenu.java. User’s menu and actions are in UserMenu.java We want to mention these 3 menu separately.

Driver class is our main class which gives way to our application. It’s features is to log in with account and directs it to menus. There are 2 menus which it can direct: Admin mode and User mode.

First of all, we want to mention Admin mode. At Admin mode, you can show all of the user’s informations, you can add user, you can show items which borrowed by people, you can show rooms, you can delete materials, you can update materials and you can add materials. When you show user’s features, output is a table of all clients’ info. When you want to add user, program will give you inputs to fill and if you give wrong inputs, it will print error, else user will be added. When you show items which borrowed, it will give a table as a result that borrowed and not gave back(currently in-hand for user). When you show rooms, it will print a table includes rooms and their status. When you want to delete a material, first of all you need to choose a material type. After that you need to enter id of the material which will be deleted after this operation. Same procedures also will apply for adding and updating materials. At updating, you can change material’s status and at adding, you can add new material by entering whole parameters such as Name, Title, ID, etc.

Second mode, which owned by user, can do only restricted operations such as showing its personal info, looking available materials, borrow a material for itself, looking for materials which is in-hand and making payment. When you want to show your personal info, you will get a table about your general informations like full name, password. When you want to look available materials, all available materials are shown as a result. When you want to borrow a material, the material you will take will write to you and this action will add to Actions table. When you want to check your owned materials, it will give your materials as result. When you want to make payment, it will deduct whole fine of the user.