

-۱

a

- جدول اول **users table** است که شامل userid , name و phoneNumber است و به صورت زیر تعریف میشود:

```
userid INT NOT NULL
name VARCHAR(30)
phoneNumber VARCHAR(30)
PRIMARY KEY(userid)
```

-جدول دوم **buyer table** است :

```
userid INT NOT NULL
,PRIMARY KEY(userid)
FOREIGN KEY(userid) REFERENCES users(userid)
```

- جدول سوم **seller table** است:

```
userid INT NOT NULL
PRIMARY KEY(userid)
FOREIGN KEY(userid) REFERENCES Users(userid)
```

-جدول چهارم **address table** است:

```
addrid INT NOT NULL
userid INT NOT NULL
name VARCHAR(50)
contactPhoneNumber VARCHAR(20)
province VARCHAR(100)
city VARCHAR(100)
streetaddr VARCHAR(100)
postCode VARCHAR(12)
PRIMARY KEY(addrid)
FOREIGN KEY(userid) REFERENCES Users(userid)
```

-جدول پنجم **product table** است:

pid INT NOT NULL
name VARCHAR(100)
type VARCHAR(50)
modelNumber VARCHAR(50)
color VARCHAR(50)
amount INT
price INT
PRIMARY KEY(pid)

-جدول ششم **order item table** است:

itemid INT NOT NULL
pid INT NOT NULL
price INT
creationTime DATE
PRIMARY KEY(itemid)
FOREIGN KEY(pid) REFERENCES Product(pid)

b

- ۱- در جدول users کلید اصلی userid است.
- ۲- در جدول buyer کلید اصلی userid و کلید خارجی نیز userid است.
- ۳- در جدول seller کلید اصلی userid و کلید خارجی نیز userid است.
- ۴- در جدول address کلید اصلی addrid و کلید خارجی userid است.
- ۵- در جدول product کلید اصلی pid که همان id محصول است میباشد.
- ۶- در جدول order item کلید اصلی itemid و کلید اصلی pid از جدول product است.

-۲

Concurrent access by multiple users

دسترسی همزمان برای افزایش کارایی نیاز است اما دسترسی های همزمان کنترل نشده میتواند منجر به ناهماهنگی شود پس DBM برای حل این مشکل رکورد را قفل میکند.

-۳

a

زمانی که یک مقداری هنوز وارد نشده است یا مقدار آن نامشخص است به جای آن مقدار NULL قرار میگیرد در کل NULL به معنای یک مقدار ناشناخته یا عدم وجود یک مقدار است. که باعث میشود ما مجبور نباشیم از ابتدا مقدار بعضی موارد را مشخص کنیم.

b

فرض کنید در جدول Student چنین رکوردی وجود دارد :

مثال برای delete

(1122, Mahsa, computer, 200)

و این رکورد هم در جدول takes وجود دارد:

(1122, m-1,11, winter, 2021, B)

اگر رکورد جدول Student را حذف کنیم چون در جدول takes به آن ارجاع داده ایم و id ما برابر ۱۱۲۲ است به مشکل بر میخوریم. در واقع در جدول takes حالا id فردی وجود دارد که حذف شده است و این درست نیست.

مثال برای insert

اگر رکوردی را به جدول takes اضافه کنیم در حالی که id ای که در قسمت id جدول takes وارد کرده ایم در جدول student دانشجویی با چنین id ای وجود نداشته باشد به مشکل بر میخوریم به عنوان مثال این رکورد را اضافه میکنیم:

(1234, m-1,11, winter, 2021, B)

در حالی که در جدول student دانشجویی با id ۱۲۳۴ وجود ندارد.

subject:
Page()

1	BIO-101
2	BIO-201
3	CS-101
4	CS-101
5	CS-190
6	CS-190
7	CS-210
8	CS-219

$\rightarrow \Pi_{course_id}(section)$

CS-101
CS-190
CS-190
CS-210
CS-219

$\sigma_{semester=Spring}(section)$

حال این رور از هم کم می کنیم :

BIO-101
BIO-201
CS-101

①

1	BIO-101
2	BIO-201
3	BIO-219
4	CS-101
5	CS-190
6	CS-210
7	CS-219
8	CS-24V

②

BIO-101
BIO-201
CS-101

جواب \rightarrow

subject:

Page()

.a

$\Pi_{\text{Book.Title, Borrow.ReturnDate}} (\sigma_{\text{memberID} = 1659, \neg \text{Is Return}} (\text{Borrow})) \bowtie_{\text{Borrow.BookID} = \text{Book.BookID}} \text{Book}$

.b

$\Pi_{\text{Member.Name}} (((\sigma_{\text{category.categoryName} = \text{'Drama'}} (\text{Category})) \bowtie_{\text{book.categoryID} = \text{category.categoryID}} \text{Book}) \bowtie_{\text{Borrow.BookID} = \text{Book.BookID}} \text{Borrow}) \bowtie_{\text{member.MemberID} = \text{borrow.MemberID}} \text{Member})$

$\Pi_{\text{Member.Name}} ((\sigma_{\text{category.categoryName} = \text{'Drama'}} (\text{Category})) \bowtie_{\text{book.categoryID} = \text{category.categoryID}} \text{Member})$

.c

$\Pi_{\text{Member.Name, Book.Title}} (\sigma_{\text{Book.categoryID} = \text{Member.categoryID}} (\text{Book} \times \text{Member}))$

$\Pi_{\text{member.Name, Book.Title}} ((\sigma_{\text{Book.categoryID} = \text{Member.categoryID}} (\text{Book} \times \text{Member})))$

$\bowtie_{\text{Book.BookID} = \text{Borrow.BookID} \wedge \text{Member.MemberID} = \text{Borrow.MemberID}} \text{Borrow}$

\square member.Name, Book.Title ((categoryName ^{physics} categoryID = category.categoryID Book)
 \star Book.BookID = Borrow.BookID ((Return Date - today - to 1
 - is Return Borrow)

\star member.memberID = Borrow.memberID (Member)

ع. $numdays * penalty$ برابر چندان جریمه ای است که باید پرداخته شود
 در این عبارت لیست نام عضوها و کتابهایی که جریمه ای آن ها
 بیش از مساوی ۱۰۰۰۰۰ را نشان می دهد

f. \star لیست نام کتابهایی است که از دسته فلسفه هستند و نویسنده
 آن ها plato نیست و کسی آن ها را به اجازت گرفته و هنوز برگردانده
 نشده اند