

به نام یکتا



طراحی پایگاه داده‌ها

تمرین 4

استاد

مهدی آخی

تهیه و تدوین

تیم دستیاران درس - بخش تمرین

بهار ۱۴۰۳

فهرست

2.....	ضوابط تمرین
2.....	سیاست‌های جزیی تمرین ۴
2.....	سوال اول (۱۵ نمره)
3.....	سوال دوم (۲۰ نمره)
3.....	سوال سوم (۴۵ نمره)
3.....	سوال چهارم (۳۵ نمره)
3.....	سوال پنجم (۳۵ نمره)

ضوابط تمرین

- پرسش و پاسخ

- برای هر تمرین یک پست در کوئرا ایجاد خواهد شد که می‌توانید سوالات و ابهامات احتمالی خود را در زیر همان پست بپرسید.

- سیاست تاخیر

- تاخیر به صورت ساعتی محاسبه می‌شود، هر تمرین را تا حداکثر ۲ روز پس از ددلاین می‌توانید ارسال کنید.
- در مجموع هر فرد 5 روز تاخیر مجاز دارد.
- به ازای هر ساعت تاخیر غیر مجاز ۲ درصد از نمره‌ی آن تمرین کم خواهد شد.

- سیاست تقلب

- در صورت مشاهده هرگونه مشابهت نامتعارف، بار اول نمره‌ی کل تمرین صفر شده و بار دوم ادامه‌ی درس میسر نخواهد بود. (برای هر دو طرف درگیر)
- مشورت و ایده گرفتن از یکدیگر در صورتی که راه حل دیگری را مشاهده نکنید ایرادی ندارد اما باید در پاسختان صراحتاً ذکر شود.

سیاست‌های جزئی تمرین ۲

- ددلاین ارسال پاسخ: 10 خرداد
- نمره تمرین: 1.5 نمره

این تمرین مربوط به اسلایدهای Normalization درس می‌باشد.

سوال اول (۱۵ نمره)

ابتدا مشخص کنید هر یک از رابطه های زیر در چه سطح نرمالی قرار دارند، سپس سطح به سطح تا BCNF نرمال سازی را انجام دهید.

(الف)

$$R = (X, Y, Z, S, T, U, V)$$

$$F = \{S \rightarrow X, S \rightarrow V, T \rightarrow Y, X \rightarrow Y, XY \rightarrow TUZ\}$$

(ب)

$$R = (A, B, C, D, E)$$

$$F = \{A \rightarrow BC, BC \rightarrow AD, D \rightarrow E\}$$

پاسخ سوال ۱:

در این سوال فرض کرده ایم که همه ستون ها Atomic هستند.

الف

$$R = (X, Y, Z, S, T, U, V)$$

$$FDs = \begin{cases} S \rightarrow X \\ S \rightarrow V \\ T \rightarrow Y \\ X \rightarrow Y \\ XY \rightarrow TUZ \end{cases} = \begin{cases} S \rightarrow X \\ S \rightarrow V \\ T \rightarrow Y \\ X \rightarrow Y \\ XY \rightarrow T \\ XY \rightarrow U \\ XY \rightarrow Z \end{cases} = \begin{cases} S \rightarrow X \\ S \rightarrow V \\ T \rightarrow Y \\ X \rightarrow T \\ X \rightarrow U \\ X \rightarrow Z \end{cases}$$

همانطور که مشخص است، ستون S می تواند باقی ستون ها را تعیین کند و از دیگر ستون ها نمی توان به S رسید. پس حتماً S داخل کلید کاندید ما موجود است و چون باقی ستون ها از آن بدست می آیند، پس کلید کاندید ما فقط $\{S\}$ است.

همه ستون های دیگر به کلید کاندید وابستگی مستقیم یا غیرمستقیم دارند، چون کلید کاندید ما تک عضوی است، پس این دیتابیس $2NF$ است. اما دیتابیس از نوع $3NF$ نیست زیرا که به عنوان مثال ستون Y به ستون T که کلید کاندید نیست وابستگی دارد.

حال با شکستن روابط تعدی موجود، دیتابیس را به نرمال مرتبه سوم تبدیل می کنیم:

$$R_1 = \{\underline{T}, Y\}$$

$$R_2 = \{\underline{S}, X, V\}$$

$$R_3 = \{\underline{X}, T, U, Z\}$$

حال چون فقط یک کلید کاندید داریم و دیتابیس از نوع $3NF$ هم هست، پس قطعاً از نوع $BCNF$ هم هست.

ب

$$R = (A, B, C, D, E)$$

$$FDs = \begin{cases} A \rightarrow BC \\ BC \rightarrow AD \\ D \rightarrow E \end{cases} = \begin{cases} A \rightarrow B \\ A \rightarrow C \\ BC \rightarrow D \\ BC \rightarrow A \\ D \rightarrow E \end{cases}$$

کلیدهای کاندید دیتابیس ما $\{A, BC\}$ هستند.

همه ستون‌های غیر کلید به کل کلید اصلی وابستگی تابعی دارد، پس از نوع $1NF$ نرمال شده است. اما چون ستون غیر کلیدی پیدا می‌شود که به ستون غیر کلید دیگری وابستگی تابعی دارد، پس از نوع $3NF$ نرمال شده نیست. به عنوان مثال ستون E به ستون D وابسته است.

پس جدول را به این شکل می‌شکنیم تا در هر رابطه‌ی تابعی، ستون سمت چپ یک کلید کاندید باشد. پس:

$$R_1 = \{\underline{A}, B, C\}$$

$$R_2 = \{\underline{B}, C, D\}$$

$$R_3 = \{\underline{D}, E\}$$

با توجه به این که در تمامی FD ها، ستون سمت چپ یک کلید کاندید است، پس دیتابیس ما از نوع $BCNF$ نرمال شده است.

سوال دوم (۲۰ نمره)

با توجه به رابطه مشخص شده، هر یک از موارد زیر را تحلیل کنید و درستی و نادرستی آن‌ها را به همراه استدلال مشخص نمایید.

$$R = (X, Y, Z)$$

$$F = \{Y \rightarrow X, XZ \rightarrow Y, X \rightarrow Z\}$$

الف) XZ ، یک candidate key است.

ب) FD سوم اضافه و قابل حذف است.

ج) وجود X در FD دوم اضافه است.

د) این رابطه در سطح نرمال $BCNF$ قرار دارد.

ه) رابطه حاصل از cartesian product این رابطه با رابطه زیر، ۴ candidate key دارد.

$$R = (A, B, C, D, E)$$

$$F = \{A \rightarrow B, AB \rightarrow CD, D \rightarrow ABC\}$$

(candidate key های هر یک روابط گفته شده و رابطه حاصل را مشخص کنید)

پاسخ سوال ۲:

$$R_1 = (X, Y, Z)$$

$$FD_{S_1} = \begin{cases} Y \rightarrow X \\ XZ \rightarrow Y \\ X \rightarrow Z \end{cases}$$

ابتدا مجموعه FDs را ساده می‌کنیم:

$$\rightarrow FD_{S_1} = \begin{cases} Y \rightarrow X \\ X \rightarrow Y \\ X \rightarrow Z \end{cases} \rightarrow FD_{S_1} = \begin{cases} Y \rightarrow X \\ X \rightarrow YZ \end{cases}$$

الف

این عبارت نادرست است. زیرا Z خودش از X بدست می‌آید و از آن نمی‌توان چیزی را بدست آورد.

ب

این عبارت نادرست است. اگر رابطه $X \rightarrow Z$ را حذف کنیم، از روابط باقی‌مانده نمی‌توان آن را بدست آورد.

ج

این عبارت نادرست است. زیرا خود Z از X بدست می‌آید. پس وجود Z اضافی است و باید حذف شود.

د

این عبارت درست است. با توجه به این که دیتابیس ما $1NF$ است و CK های ما همگی تک‌عضوی هستند، پس $2NF$ هم هست. با توجه به این که X, Y کلیدهای کاندید ما هستند و ستون Z به ستونی به جز کلیدهای کاندید وابسته نیست، پس دیتابیس $3NF$ هم هست. همچنین در سمت چپ تمامی FD های موجود، فقط کلید کاندید وجود دارد، پس دیتابیس ما در نهایت $BCNF$ است.

ه

این عبارت درست است.

$$R_2 = (A, B, C, D, E)$$

$$FD_{S_2} = \begin{cases} A \rightarrow B \\ AB \rightarrow CD \\ D \rightarrow ABC \end{cases}$$

واضح است که کلیدهای کاندید این دیتابیس، DE, AE هستند. همچنین دیتابیس اول هم دارای کلیدهای کاندید X, Y هستند. پس کلید کاندید دیتابیس حاصل $Cartesian Product$ این دو، باید شامل یک کلید کاندید از اولی و یک کلید کاندید از دومی باشد. پس در کل 2×2 حالت داریم:

$$CK = \{YAE, YDE, XAE, XDE\}$$

سوال سوم (۴۵ نمره)

یک پایگاه داده به صورت زیر را در نظر بگیرید که نشان‌دهنده حساب کاربری دانشجویان دانشکده کامپیوتر است.

UserAccounts(userID: int, name: string, email: string, age: int, city: string)

UserEducation(userID: int, school : string, startYear: int, finishYear: int)

UserInterests(userID: int, interest: string)

UserPartner(userID: int, partnerUserID: int, marriedToPartner: boolean)

ConnectedTo(userID: int, friendUserID: int, privilegeLevel : int)

فرض کنید یک ویژگی جدید به این محیط قرار است اضافه کنیم که برای صندوق دریافت ایمیل تعریف می‌شود. این ویژگی یک پیام دریافتی به صورت "private" است. همچنین قرار است یک دیوار (wall) به صورت عمومی تعریف شود که دیگران نیز بتوانند در آن پست‌های خود را قرار دهند (در این دیوار پیام‌ها برای تمامی دوستان شما نیز قابل مشاهده است).

هر پیام یک شناسه و یک سطح مشاهده (public یا private بودن) دریافت می‌کند. هر کلمه‌ای نیز یک شناسه دریافت می‌کند و ممکن است در هر پیام چند بار ظاهر شود. در این محیط نیاز است که ما از روابط زیر پیروی کنیم:

R(userID, msgID, visibility, wordID, wordText, wordPosition)

همچنین functional dependency های زیر نیز داده شده است:

userID, msgID \rightarrow visibility

wordID \rightarrow wordText

wordText \rightarrow wordID

msgID, wordPosition \rightarrow wordID

msgID \rightarrow userID

msgID, wordID \rightarrow visibility

msgID, wordPosition \rightarrow wordText

الف) یک الگوریتم کلی (در حد توضیحات مرحله به مرحله) ارائه دهید که بتوان یک Minimal Cover برای functional dependency ها یافت.

ب) با توجه به قسمت قبل برای functional dependency های فوق یک [Minimal Cover](#) ارائه دهید.

ج) آیا FD زیر در [closure](#) آن FD های داده شده قرار دارد؟ توضیح دهید.

$\text{wordID, msgID, visibility} \rightarrow \text{userID}$

د) یک شمای رابطه‌ای به صورت 3NF برای دامنه داده شده ارائه دهید و همچنین کلیدهای آن را مشخص کنید.

آیا تجزیه شما Lossless است؟

آیا وابستگی‌ها را حفظ می‌کند؟

ه) یک عبارت بر اساس جبر رابطه‌ای بنویسید که بتواند تمام رابطه‌های نرمالیز شده شما را با join کردن، به همان رابطه اولیه R برساند.

پاسخ سوال ۳:

1. Specify a minimal cover for the functional dependencies.

Answer:

$\text{msgID} \rightarrow \text{visibility}$

$\text{wordID} \rightarrow \text{wordT ext}$

$\text{wordT ext} \rightarrow \text{wordID}$

$\text{msgID, wordPosition} \rightarrow \text{wordID}$

$\text{msgID} \rightarrow \text{userID}$

2. Is the FD $\text{wordID, msgID, visibility} \rightarrow \text{userID}$ in the closure of the functional dependencies?

Answer:

Yes: $\text{msgID} \rightarrow \text{userID}$.

3. Provide a relational schema in 3NF for the domain, using relational schema notation as above, and indicating keys with underlines under the attributes. Is your decomposition lossless? Dependency preserving?

Answer:

MsgVisibility(msgID, visibility)

Lexicon(wordID \rightarrow wordText)

Occurs(msgID, wordPosition, wordID)

Owner(msgID, userID)

Yes, this is given by the properties of 3NF normalization.

4. Write a relational algebra expression, putting parentheses around each pair of expressions being joined, that re-joins all of your normalized relations to build the single original relation R.

Answer:

((MsgVisibility \bowtie Owner) \bowtie Occurs) \bowtie Lexicon

9 Finding the Minimal Cover

Given a set of functional dependencies F :

1. Start with F
2. Remove all trivial functional dependencies
3. Repeatedly apply (in whatever order you like), until no changes are possible
 - Union Simplification (it is better to do it as soon as possible, whenever possible)
 - RHS Simplification
 - LHS Simplification
4. Result is the minimal cover

Small example Applying to algorithm to EGS with

1. $E \rightarrow G$
2. $G \rightarrow S$
3. $E \rightarrow S$

Using the union rule, we combine 1 and 3 and get

1. $E \rightarrow GS$
2. $G \rightarrow S$

Simplifying RHS of 1 (this is the only attribute we can remove), we get:

1. $E \rightarrow G$
2. $G \rightarrow S$

Done!

سوال چهارم (۳۵ نمره)

تصور کنید که وظیفه باز طراحی پایگاه داده برای سیستم مدیریت کتابخانه دانشگاه به شما محول شده است. این سیستم باید بتواند انواع مختلفی از امانت ها، فرمت های مختلف و پیگیری تراکنش های خاص کاربران را مدیریت کند. جدول اولیه ای که برای تجزیه و تحلیل و بهینه سازی به شما داده شده است به شرح زیر است:

Library(LoanID, MemberID, MemberName, MemberAddress, MemberType, ItemID, ItemTitle, ItemType, ItemFormat, AuthorID, AuthorName, DueDate, LoanDate, ReturnDate, LateFee)

وابستگی های عملکردی به شرح زیر است:

1. **LoanID** → MemberID, ItemID, LoanDate, DueDate, ReturnDate, LateFee
2. **MemberID** → MemberName, MemberAddress, MemberType
3. **ItemID** → ItemTitle, ItemType, ItemFormat, AuthorID
4. **AuthorID** → AuthorName
5. **MemberType, ItemType** → LateFee

1. انواع آنومالی ها (درج، بروزرسانی، حذف) که ممکن است با طرح فعلی رخ دهند را شناسایی کنید.
2. جدول Library را تا فرم نرمال BCNF نرمال سازی کنید. هر مرحله از فرآیند نرمال سازی را شامل ایجاد جدول های جدید و بازتوزیع صفات توضیح دهید.
3. ارزیابی کنید که طراحی چگونه تغییرات آینده را پشتیبانی می کند، مانند افزودن انواع جدید موارد یا تغییر سیاست های امانت بر اساس انواع اعضا.

پاسخ سوال ۴:

وابستگی‌های عملکردی و آنومالی‌ها

1. $\text{LoanID} \rightarrow \text{MemberID}, \text{ItemID}, \text{LoanDate}, \text{DueDate}, \text{ReturnDate}, \text{LateFee}$

2. $\text{MemberID} \rightarrow \text{MemberName}, \text{MemberAddress}, \text{MemberType}$

3. $\text{ItemID} \rightarrow \text{ItemTitle}, \text{ItemType}, \text{ItemFormat}, \text{AuthorID}$

4. $\text{AuthorID} \rightarrow \text{AuthorName}$

5. $\text{MemberType}, \text{ItemType} \rightarrow \text{LateFee}$

آنومالی‌های ممکن

درج: نمی‌توان اطلاعات اعضای جدید یا آیتم‌های جدید را بدون وجود یک تراکنش امانت ثبت کرد.
بروزرسانی: تغییرات در اطلاعات اعضا یا نویسندگان باید در تمامی ردیف‌ها اعمال شود، که می‌تواند منجر به ناسازگاری داده‌ها شود.
حذف: حذف آخرین تراکنش یک عضو یا آیتم منجر به حذف اطلاعات مربوط به آن عضو یا آیتم می‌شود.

نرمال‌سازی تا فرم نرمال BCNF

1. فرم نرمال اول (1NF)

جدول Library به صورت زیر در 1NF است، زیرا تمامی مقادیر اتمیک هستند:

Library(LoanID, MemberID, MemberName, MemberAddress, MemberType, ItemID, ItemTitle, ItemType, ItemFormat, AuthorID, AuthorName, DueDate, LoanDate, ReturnDate, LateFee)

2. فرم نرمال دوم (2NF)

برای رسیدن به 2NF، جداول را به گونه‌ای تجزیه می‌کنیم که هیچ وابستگی جزئی وجود نداشته باشد:
1. جدول امانت‌ها:

(Loans(LoanID, MemberID, ItemID, LoanDate, DueDate, ReturnDate, LateFee

2. جدول اعضا:

(Members(MemberID, MemberName, MemberAddress, MemberType

3. جدول آیتم‌ها:

(Items(ItemID, ItemTitle, ItemType, ItemFormat, AuthorID

4. جدول نویسندگان:

(Authors(AuthorID, AuthorName

3. فرم نرمال سوم (3NF)

در این مرحله، تمامی وابستگی‌های ترانزیتی را حذف می‌کنیم:

5. جدول هزینه تأخیر:

(LateFees(MemberType, ItemType, LateFee

4. فرم نرمال بویس-کد (BCNF)

برای اطمینان از BCNF، باید هر وابستگی تابعی $X \rightarrow Y$ را بررسی کنیم:

1. جدول Loans: LoanID کلید اصلی است و تمامی صفات به آن وابسته هستند. بنابراین، این جدول در BCNF است.

2. جدول Members: MemberID کلید اصلی است و تمامی صفات به آن وابسته هستند. بنابراین، این جدول در BCNF است.

3. جدول Items: ItemID کلید اصلی است و تمامی صفات به آن وابسته هستند. بنابراین، این جدول در BCNF است.

4. جدول Authors: AuthorID کلید اصلی است و AuthorName به آن وابسته است. بنابراین، این جدول در BCNF است.

5. جدول LateFees: کلید ترکیبی MemberType و ItemType است و LateFee به این کلید وابسته است. بنابراین، این جدول نیز در BCNF است.

طراحی نهایی جداول

1. جدول امانت‌ها:

(Loans(LoanID, MemberID, ItemID, LoanDate, DueDate, ReturnDate, LateFee

2. جدول اعضا:

(Members(MemberID, MemberName, MemberAddress, MemberType

3. جدول آیتم‌ها:

(Items(ItemID, ItemTitle, ItemType, ItemFormat, AuthorID

4. جدول نویسندگان:

(Authors(AuthorID, AuthorName

5. جدول هزینه تأخیر:

(LateFees(MemberType, ItemType, LateFee

ارزیابی و انعطاف‌پذیری

افزودن انواع جدید موارد:

می‌توان به راحتی انواع جدید موارد (ItemType و ItemFormat) را با افزودن ردیف‌های جدید به جدول Items مدیریت کرد.

تغییر سیاست‌های امانت:

تغییر سیاست‌های امانت بر اساس انواع اعضا (MemberType) و انواع آیتم‌ها (ItemType) با به‌روزرسانی جدول LateFees قابل انجام است.

این طراحی نرمال‌سازی شده نه تنها باعث کاهش آنومالی‌های داده‌ها می‌شود، بلکه انعطاف‌پذیری و قابلیت نگهداری داده‌ها را نیز افزایش می‌دهد.

سوال پنجم (۳۵ نمره)

یک کتابفروشی اخیراً تصمیم به راه‌اندازی سیستم قرض دادن کتاب کرده است. در این سیستم هر شخصی که در کتابفروشی ثبت نام کرده باشد می‌تواند کتابی را قرض بگیرد و تا مهلت تعیین شده آن را مطالعه کرده و درباره آن نظری ثبت کند. مدل رابطه‌ای سیستم به صورت زیر می‌باشد:

BookBorrow (Borrower_id , Book_id , Borrow_id , Borrower_name , Due_date , Writer , Genre , Feedback)

همچنین بر روی این مدل Functional Dependency های زیر برقرار است :

$Borrower_id \rightarrow Borrower_name$

$Borrow_id \rightarrow Due_date$

$Book_id \rightarrow Genre, Writer$

$Writer \rightarrow Genre$

$(Borrower_id, Book_id) \rightarrow Feedback$

$Feedback \rightarrow Borrower_id$

الف) این مدل را تا سطح 3NF نرمال‌سازی کنید. پیش از نرمال‌سازی در هر سطح، آنومالی‌های INSERT, UPDATE و DELETE را ذکر کرده و مثالی برای آن بزنید.

ب) در هر سطح بررسی کنید که آیا تجزیه انجام شده Dependency Preserving است یا خیر. در صورت جواب منفی ذکر کنید که کدام Dependency از دست می‌رود.

ج) این مدل را تا سطح BCNF نرمال‌سازی کنید و مانند موارد قبل آنومالی‌های پیش از نرمال‌سازی را ذکر کرده و Dependency Preserving بودن تجزیه را نیز بررسی کنید. در صورت نقض آن ذکر کنید که کدام Dependency از دست می‌رود.

د) تجزیه‌ای برای این مدل مثال بزنید که Lossless نباشد و پس از Join کردن رابطه‌های تجزیه شده در آن داده‌هایی از دست رفته یا تولید شوند.

پاسخ سوال ۵:

الف:

در ابتدا در 1NF قرار داریم و هر سه نوع آنومالی وجود دارند:

- درج: اگر بخواهیم یک کتاب جدید اضافه کنیم که کسی آنرا امانت نگرفته باشد، باید یک سطر اضافه کنیم که اطلاعات کتاب را داشته باشد و بقیه را خالی بگذاریم.
- حذف: اگر یک کتاب فقط توسط یک کاربر امانت گرفته شده باشد با حذف آن کاربر، اطلاعات کتاب نیز حذف می‌شود.
- بروزرسانی: اگر بخواهیم نام یک کاربر را تغییر دهیم، باید این تغییر را در تمام سطرهایی که این کاربر امانت گرفته است اعمال کنیم.

با توجه به FD های داده شده CK این رابطه ترکیب $\{BorrowerId, BookId, BorrowId\}$ است. با توجه به اینکه وابستگی $BorrowerId \rightarrow BorrowerName$ داریم که سمت چپ آن بخشی از CK است، پس در 2NF قرار نداریم.

در مرحله اول وابستگی های $BorrowerId \rightarrow BorrowerName$ و $BorrowId \rightarrow DueDate$ و $BookId \rightarrow Genre, Writer$ را تجزیه می‌کنیم و روابط زیر را می‌سازیم:

$$\begin{cases} R_1 = \{\underline{BorrowerId}, BorrowerName\} \\ R_2 = \{\underline{BorrowId}, DueDate\} \\ R_3 = \{\underline{BookId}, Genre, Writer\} \\ R_4 = \{\underline{BorrowerId}, \underline{BookId}, \underline{BorrowId}, Feedback\} \end{cases}$$

در مرحله بعد وابستگی $(BorrowerId, BookId) \rightarrow Feedback$ را تجزیه می‌کنیم و روابط زیر را می‌سازیم:

$$\begin{cases} R_1 = \{\underline{BorrowerId}, BorrowerName\} \\ R_2 = \{\underline{BorrowId}, DueDate\} \\ R_3 = \{\underline{BookId}, Genre, Writer\} \\ R_4 = \{\underline{BorrowerId}, \underline{BookId}, Feedback\} \\ R_5 = \{\underline{BorrowerId}, \underline{BookId}, \underline{BorrowId}\} \end{cases}$$

در این مرحله در 2NF قرار داریم و در 3NF نیستیم چون $BookId \rightarrow Writer \rightarrow Genre$ در R_3 داریم.

در این سطح هر سه نوع آنومالی وجود دارد:

- درج: اگر بخواهیم یک نویسنده بدون کتاب اضافه کنیم باید یک سطر در R_3 اضافه کنیم که $BookId$ آن خالی است.
- حذف: اگر بخواهیم یک نویسنده فقط یک کتاب داشته باشد در صورت حذف آن کتاب، نویسنده نیز حذف خواهد شد.
- بروزرسانی: اگر بخواهیم ژانر یک نویسنده را تغییر دهیم باید این تغییر را در تمام سطرهایی که این نویسنده کتاب دارد اعمال کنیم.

برای تبدیل به 3NF روابط زیر را می‌سازیم:

$$\begin{cases} R_1 = \{\underline{BorrowerId}, BorrowerName\} \\ R_2 = \{\underline{BorrowId}, DueDate\} \\ R_3 = \{\underline{BookId}, Writer\} \\ R_4 = \{Writer, Genre\} \\ R_5 = \{\underline{BorrowerId}, \underline{BookId}, Feedback\} \\ R_6 = \{\underline{BorrowerId}, \underline{BookId}, BorrowId\} \end{cases}$$

ب:

تمام تجزیه‌های انجام شده Preserving Dependency هستند چون تمام وابستگی‌های اولیه در تجزیه‌ها حفظ شده‌اند.

ج:

چون که داریم $Feedback \rightarrow BorrowerId$ پس R_5 در BCNF نیست، چون که $Feedback$ یک super-key از این رابطه نیست.

آنومالی‌ای که در این سطح وجود دارد این است که دو کاربر نمیتوانند فیدبک یکسان ثبت کنند.

برای حل این مشکل رابطه‌ی R_5 را به دو رابطه تجزیه می‌کنیم:

$$R_5 \longrightarrow \begin{cases} R_7 = \{\underline{BookId}, Feedback\} \\ R_8 = \{\underline{Feedback}, BorrowerId\} \end{cases}$$

با این تجزیه وابستگی $(BorrowerId, BookId) \rightarrow Feedback$ از بین رفت و دیگر Dependency-Preservation نداریم.

د:

فرض کنید در بخش قبل R_3 و R_4 را به دو رابطه به این شکل تجزیه کرده بودیم:

$$\{R_3, R_4\} \longrightarrow \begin{cases} R_9 = \{\underline{BookId}, Genre\} \\ R_{10} = \{Genre, \underline{Writer}\} \end{cases}$$

در این حالت اگر دو کتاب با نویسنده‌های متفاوت ولی ژانر یکسان داشته باشیم، پس از جوین کردن رابطه‌های R_9 و R_{10} تناظر کتاب به نویسنده از بین می‌رود و رابطه حاصل به ازای ۲ کتاب، ۴ سطر خواهد داشت.