

طراحی پایگاه داده‌ها

نیم‌سال دوم ۰۳-۰۲

استاد: مهدی آخی

پاسخ‌دهنده: معین آعلی - ۴۰۱۱۰۵۵۶۱



دانشکده‌ی مهندسی کامپیوتر

تمرین چهارم

پاسخ مسئله‌ی ۱.

پاسخ مسئله‌ی ۲.

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

$$FDs_1 = \begin{cases} Y \rightarrow X \\ XZ \rightarrow Y \\ X \rightarrow Z \end{cases}$$

ابتدا مجموعه FDs را ساده می‌کنیم:

$$\rightarrow FDs_1 = \begin{cases} Y \rightarrow X \\ X \rightarrow Y \\ X \rightarrow Z \end{cases} \rightarrow FDs_1 = \begin{cases} Y \rightarrow X \\ X \rightarrow YZ \end{cases}$$

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.

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

$$FDs_2 = \begin{cases} A \rightarrow B \\ AB \rightarrow CD \\ D \rightarrow ABC \end{cases}$$

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:

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

پاسخ مسئله‌ی ۳.

الف

مراحل زیر را طی می‌کنیم:

۱. هر وابستگی تابعی در F را به طوری تجزیه کنید که در سمت راست فقط یک ستون وجود داشته باشد.
۲. صفات اضافی را با محاسبه بسته closure صفات سمت چپ به جز صفت مورد نظر، حذف کنید.
۳. وابستگی‌های تابعی زائد را حذف کنید.
۴. اطمینان حاصل کنید که سمت راست هر وابستگی تابعی فقط شامل یک صفت باشد.
۵. روابطی که با استفاده از تعدی به وجود آمده‌اند را حذف می‌کنیم.

ب

پس از اجرای الگوریتم فوق، به مجموعه FDs زیر می‌رسیم:

$$FDs = \begin{cases} msgID, wordPosition \rightarrow wordID \\ wordID \rightarrow wordText \\ wordText \rightarrow wordID \\ msgID \rightarrow visibility \\ msgID \rightarrow userID \end{cases}$$

ج

این عبارت درست است. زیرا:

$$\begin{cases} msgID, wordID \rightarrow visibility \\ msgID \rightarrow userID \end{cases} \longrightarrow \{ msgID, wordID, visibility \rightarrow userID \}$$

د

می‌دانیم که کلید کاندید ما $\{msgID, wordPosition\}$ است. در ابتدا *Partial Dependency* موجود در $wordID$ و سپس *Transitive Dependency* موجود بین $wordText$ و کلید کاندید را از بین می‌بریم. پس در کل به ۳ رابطه نیاز داریم:

$$\begin{aligned} R_1 &= \{ \underline{wordID}, wordText \} \\ R_2 &= \{ msgID, \underline{userID}, visibility \} \\ R_3 &= \{ \underline{msgID}, \underline{wordPosition}, wordID \} \end{aligned}$$

با توجه به اینکه که مجموعه کلیدهای کاندید ما تک عضوی است، پس دیتابیس ما *BCNF* هم هست. در نتیجه چون نرمال‌سازی ما در سطح *BCNF* است، پس *LossLess* است و *Dependency*‌ها را حفظ می‌کند.

ه

$$R = R_1 \bowtie_{R_1.wordID=R_3.wordID} R_3 \bowtie_{R_3.msgID=R_2.msgID} R_2$$