

شماره دانشجویی: 99243100

اسم و اسم خانوادگی: عبدالقدیر فرتاش

تمرین سری اول

جواب مسئله (1)

1) solution (انزو): سیستم فایل‌ها مکانیزم جداسازی تراکنش‌ها را ندارند و ممکن است باعث تداخل و خرابی داده‌ها شوند، در حالی که DBMS با مکانیزم‌های ACID از تداخل جلوگیری می‌کند.

2) Data Access (دسترسی به داده‌ها): دسترسی به داده‌ها در سیستم فایل‌ها غیرساخت‌یافته و پیچیده است و نیاز به نوشتن یک برنامه جدید، برای انجام هر کار جدید است، اما DBMS با زبان‌هایی مثل SQL دسترسی سریع و ساخت‌یافته را فراهم می‌کند.

3) Redundancy (افزونی): فرمت‌های مختلف فایل داده‌ها باعث می‌شود که بعضی از داده‌ها را تکراری داشته باشیم که سیستم فایل‌ها این مازد یا افزونی داده‌ها را کنترل نمی‌کنند، اما DBMS با تکنیک‌هایی مثل نرمال‌سازی از تکرار غیرضروری جلوگیری می‌کند.

به طور کلی، DBMS امنیت، کارایی و یکپارچگی بهتری نسبت به سیستم فایل‌ها ارائه می‌دهد.

جواب مسئله (2)

الف) استقلال داده در DBMS یعنی امکان تغییر در ساختار فیزیکی یا منطقی داده‌ها بدون تأثیر بر سایر سطوح. اگر وجود نداشته باشد، تغییرات در ذخیره‌سازی یا ساختار داده‌ها به اصلاحات گسترده در برنامه‌ها و سطوح دیگر نیاز دارد، که باعث افزایش پیچیدگی و هزینه‌های نگهداری می‌شود.

ب) دو نوع استقلال داده وجود دارد:

استقلال داده فیزیکی: تغییرات در ساختار فیزیکی ذخیره‌سازی داده‌ها بدون تأثیر بر سطح منطقی.

استقلال داده منطقی: تغییرات در ساختار منطقی داده‌ها (مثل افزودن ستون) بدون نیاز به تغییر برنامه‌های کاربردی.

پ) طراحی بد یک پایگاه داده باعث مشکلاتی همچون (1) داده‌های تکراری (2) اساین نشدن مقدار null به یک ویژگی می‌شود.

ج) در طراحی اسکیمای پایگاه داده، ستون ID به دلیل سادگی در ارجاع، بهبود عملکرد جستجو، کاهش پیچیدگی و سازگاری با ابزارهای مدیریت حتی با وجود ویژگی‌های یکتا تعریف می‌شود. چون بعضی موقع‌ها ویژگی یکتا چنین امکانات را ندارد و باعث می‌شود که کارایی سیستم پایین بیاید.

جواب مسئله (3)

- رابطه (Relation):
The department table

- اسکیمای (Schema):
department = (dept_name, building, budget)

- ویژگی (Attribute):
{dept_name, building, budget}

dept_name	building	budget
Comp. Sci.	Taylor	100000
Biology	Watson	90000
Elec. Eng.	Taylor	85000
Music	Packard	80000
Finance	Painter	120000
History	Painter	50000
Physics	Watson	70000

(b) The department table

- نمونه (Relation Instance):
یعنی مجموعه‌ای از مقادیر واقعی است که در هر لحظه خاص در جدول وجود دارد. مثل خود جدول

- دامنه ویژگی (Domain of Attribute):
dept_name **varchar(20),**
building **varchar(20),**
budget **numeric(8,2)**

- چندتایی (Tuple):
[Comp. Sci, Taylor, 100000],
[Music, Packard, 8500],

جواب مسئله 4

از اول همه باید بدانیم مدیریت تراکنش چیست؟ مدیریت تراکنش عبارت از مجموعه‌ای از عملیات است که یک تابع نوع منطقی را در یک برنامه پایگاه داده انجام می‌دهد یا به عبارت ساده‌تر همیشه گفت انجام عملیات بصورت اتمیک یعنی اگر یک عملیات اجرا شود باید تمام آن بخش اجرا شود در غیر اینصورت اصلاً اجرا نشود مثلاً انتقال دادن پول از یک حساب به حساب دیگر شامل دو مرحله است.

- 1) پول از حساب کاربر اولی کم میشود
- 2) پول به حساب کاربر دومی اضافه میشود

اگر مدیریت تراکنش نباشد و کاربر بخواد پول انتقال بدهد پول از حسابش کم می‌شود اگر مشکل فنی مثل قطع شد برق یا اختلال اینترنت رخ دهد باعث می‌شود که پول به حساب کاربر دومی واریز نشود و این باعث برهم ریختی سیستم و پایگاه داده می‌شود به همین خاطر مدیریت تراکنش در پایگاه داده خیلی مهم است.

جواب مسئله 5

بخش الف)

$$\Pi_{name}(\sigma_{salary > 5000}(instructor))$$

بخش ب)

$$\Pi_{salary}(Instructor \bowtie (\sigma_{Budget \geq 5000}(department)))$$

بخش پ)

$$\Pi_{name}(Instructor \bowtie (Advisor \bowtie (Student \bowtie (\sigma_{grade > 17}(Takes))))))$$

بخش ت)

$$\Pi_{name}(Instructor \bowtie (teaches \bowtie (section \bowtie (takes \bowtie (\sigma_{ID=25}(student)))))$$

$$\Pi_{title}(course \bowtie (\Prereq \bowtie (\sigma_{title='Physics'}(Course))))$$

$$\Pi_{title}(\sigma_{id=(\Pi_{id}(\sigma_{title='physics'}(Course)))}(\Prereq))$$

جواب مسئله 6 بله امکان Natural join در بخش ت وجود دارد. بخاطر که با استفاده از ID دانشجو میتواند به درس های که دانشجو برداشته دسترسی داشت و مجددا با استفاده از Natural join بین رابطه بدست اومد و section زد و چون course_id در هر دوش مشترک است میتواند با استفاده course_id به Teaches دسترسی پیدا کرده و بالاخر به اون استاد که به این دانشجو تدریس کرده است دسترسی یافت.