

**اسم واسم خانوادگی: عبدالقدیر فرتاش شماره دانشجویی: 99243100**

**تمرین سری اول**

**جواب مسله 1)**   
**I) solation (1انزوا):** 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.

**) Data Access (2دسترسی 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.

**) Redundancy (3افزونگی**): فرمت های مختلف فایل داده ها باعث می شود که بعضی از داده ها را تکراری داشته باشیم که سیستم فایل‌ها این مازد یا افزونگی 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 مسله 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:

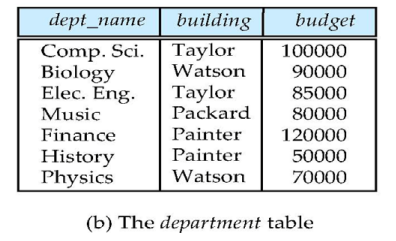
**استقلال داده فیزیکی:** تغییرات در ساختار فیزیکی ذخیره‌سازی داده‌ها بدون تأثیر بر سطح منطقی.

**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) داده های تکراری 2) اساین نشدن مقدار null به یک ویژگی می شود.

**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 موقع ها ویژگی یکتا چنین امکانات را ندارد و باعث می شود که کارایی سیستم پایین بیاید.

**جواب مسله 3)**

* رابطه (Relation):  
  **The department table**
* اسکمیا (Schema):  
  **department = (dept\_name, building, budget)**
* ویژگی (Attribute):  **{dept\_name, building, budget}**
* 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 مسله 4)**   
از اول همه باید بدانیم مدیریت تراکنش چیست؟ مدیریت تراکنش عبارت از مجموعه ای از عملیات است که یک تابع نوع منطقی را در یک برنامه پایگاه داده انجام می دهد یا به عبارت ساده تر میشه گفت انجام عملیات بصورت اتمیک یعنی اگر یک عملیات اجرا شود باید تمام آن بخش اجرا شود در غیر اینصورت اصلا اجرا نشود مثلا  
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 می شود اگر مشکل فنی مثل قطع شد برق یا اختلال انترینت رخ دهد باعث می شود که پول به حساب کاربر دومی واریز نشود و این باعث برهم ریختی سیستم و پایگاه داه می شود به همین خاطر مدیریت تراکنش در پایگاه داده خیلی مهم است.

**جواب مسله 5)**   
**بخش الف)**

**بخش ب)**

**بخش پ)**

**بخش ت)**

**بخش ث)**

**بخش ج)**

**جواب مسله 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 بلاخر به اون استاد که به این دانشجو تدریس کرده است دسترسی یافت.