## ۷ـ۱ـ مفاهیم پایهٔ پایگاه دادهٔ رابطه‌ای

**۱ــ۷ــ۱ــ محیط عملیاتی[۱] و تعریف موجودیت :** محیط عملیاتی،محیطی است که می‌خواهیم یک سیستم ذخیره و بازیابی برای آن ایجاد کنیم. مانند : مدرسه یا دانشگاه، بانک، تعمیرگاه، فروشگاه و...

در هر محیط عملیاتی تعدادی موجودیت وجود دارد. موجودیت، هر مفهوم و یا شیء در محیط عملیاتی است. مثلاً دانش‌آموز، معلم، کارمند مدرسه، کلاس و درس موجودیت‌های محیط عملیاتی مدرسه هستند. تمام موجودیت‌های یک محیط عملیاتی مهم نیستند بلکه موجودیت‌هایی برای ما مهم هستند که می‌خواهیم در مورد آن‌ها اطلاع داشته باشیم. به‌عنوان یک مثال ساده در محیط عملیاتی مدرسه موجودیت‌های دانش‌آموز، معلم و درس برای سیستم فرضی ثبت‌نام مهم و قابل‌توجه هستند.

در هر محیط عملیاتی مجموعه‌ای از موجودیت‌های مناسب و مرتبط براساس مرز و محدودهٔ سیستم مورد طراحی، انتخاب می‌شوند.

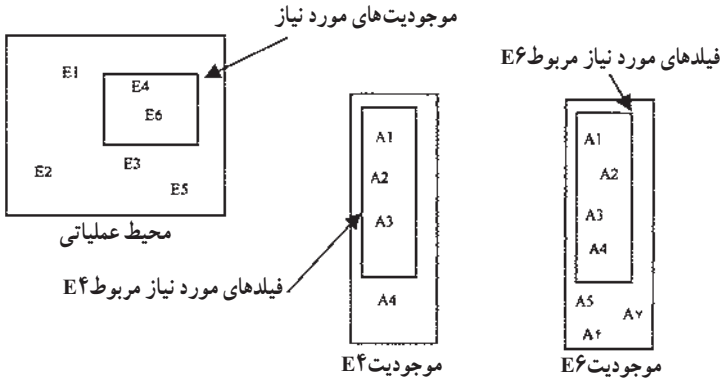
☑ **تمرین ۱ـ۱ :** در محیط‌های عملیاتی بانک، خانواده و باشگاه ورزشی چه موجودیت‌هایی قابل تشخیص است؟

**پرسش :** در چه مواقعی از فیلدهای خود موجودیت برای تعریف کلید استفاده نمی‌شود؟

**۲ــ۷ــ۱ــ ویژگی‌های هر موجودیت :** صفات و ویژگی‌های هر موجودیت درواقع همان فیلدها هستند. در مورد انتخاب فیلدهای یک موجودیت نیز مانند انتخاب موجودیت‌های محیط عملیاتی فقط صفات مورد نیاز و مرتبط به‌عنوان فیلد درنظر گرفته می‌شوند و تمام صفات یک موجودیت درنظر گرفته نمی‌شوند. به‌عنوان مثال، در محیط عملیاتی مدرسه در مورد اطلاعات مربوط به دانش‌آموز ویژگی‌هایی نظیر قد، وزن، رنگ چشم درنظر گرفته نمی‌شود چرا که در محیط عملیاتی موردنظر به آن‌ها نیاز نیست، ولی ممکن است در محیط عملیاتی دیگر مناسب و مفید باشند.

---

۱ــ Operational Environment




شکل ۱ــ۱ــ زیر مجموعه موجودیت‌ها و فیلدهای انتخاب شده برحسب نیاز

**کنجکاوی**

**درنظر گرفتن محدودهٔ محیط عملیاتی قبل از تشخیص موجودیت و ویژگی‌ها چه فایده‌هایی دارد؟**

**تمرین ۲ـ۱:** در سه محیط عملیاتی تمرین ۱ــ۱ موجودیت‌های مرتبط به همراه فیلدهای موردنیاز را شناسایی کنید.

**مثال ۱ـ۱**

محیط عملیاتی مدرسه را درنظر بگیرید. موجودیت‌ها و ویژگی‌های هر موجودیت را تعریف کنید.

***موجودیت‌ها:*** دانش‌آموز، معلم و درس است.

***ویژگی‌های موجودیت دانش‌آموز:*** شماره دانش‌آموزی، نام، نام‌خانوادگی، تلفن، آدرس.

***ویژگی‌های موجودیت معلم:*** نام، نام‌خانوادگی، تلفن، آدرس.

***ویژگی‌های موجودیت درس:*** شماره درس، نام درس، تعداد واحد، ساعت کلاس و نام کلاس.

* دقت کنید نام معلم ویژگی درس نیست زیرا تداخل اطلاعاتی رخ می‌دهد، بعدها خواهیم دید چرا مشخصه معلم به درس اضافه می‌شود.

**۳ــ۷ــ۱ــ ارتباط**[۱] : پیش از این عنوان شد که در محیط عملیاتی، موجودیت‌هایی وجود دارند. معمولاً بین موجودیت‌ها ارتباط یا ارتباط‌هایی وجود دارد که قابل شناسایی است. در محیط عملیاتی مدرسه، بین موجودیت‌های درس، دانش‌آموز و معلم ارتباط‌های زیر قابل تعریف است :

۱ــ معلم درس ارایه می‌کند.

۲ــ دانش‌آموز هر ترم چندین درس ثبت‌نام می‌کند.

۳ــ دانش‌آموز برای هریک از درس‌های ثبت‌نام شده نمره می‌گیرد.

همان‌طورکه ملاحظه می‌شود :

- در هر ارتباط موجودیت‌هایی شرکت دارند.
- هر ارتباط دارای یک مفهوم یا عملکرد است. مثلاً در ارتباط اول «ارایه کردن» عملکرد می‌باشد.
- هر ارتباط دارای «ماهیتی» خاص است. مثلاً در ارتباط دوم ماهیت ارتباط می‌تواند یک به چند باشد. یعنی یک دانش‌آموز چند درس اخذ می‌کند.
- برای ایجاد ارتباط، از کلید خارجی استفاده می‌شود. کلید خارجی فیلدی است مشترک بین دو رابطه که در یک رابطه کلید اصلی است و در رابطه‌ای که کلید خارجی محسوب می‌شود، به تنهایی کلید اصلی نیست.
- ارتباط مانند موجودیت می‌تواند ویژگی‌هایی داشته باشد.

**کنجکاوی**

**آیا بین سه موجودیت نیز می‌توان ارتباط برقرار کرد؟ مثال بزنید.**

**۴ــ۷ــ۱ــ انواع ماهیت در ارتباط**[۲] : تناظر بین موجودیت‌ها، ماهیت ارتباط است و به یکی از سه نوع زیر وجود دارد :

۱ــ یک به یک ۱ : ۱

۲ــ یک به چند ۱ : N

۳ــ چند به چند M : N

---

۱ــ Relation ۲ــ Cardinality

در ارتباط یک به یک، یک نمونه از موجودیت اول فقط با یک نمونه از موجودیت دیگر ارتباط دارد. مثلاً، هر مدرسه یک مدیر دارد. در ارتباط یک به چند، یک نمونه موجودیت با چندین نمونه از موجودیت دیگر مرتبط است. مثلاً یک معلم چند درس ارایه می‌کند. در حالت سوم، یک نمونه از موجودیت اول با چندین نمونه از موجودیت دوم مرتبط است و برعکس. مثلاً یک دانش‌آموز چند درس اخذ می‌کند و یک درس به‌وسیلهٔ چند دانش‌آموز اخذ می‌شود.

**تمرین۳ـ۱:** مثالی از حالت چند به چند ارایه کنید.

**حالتی را مثال بزنید که موجودیت با خودش مرتبط باشد.**

**مثال ۲ـ۱**

در محیط عملیاتی مدرسه، ماهیت ارتباط‌ها به‌صورت زیر است :

۱ــ یک معلم چند درس ارایه می‌کند و هر درس به‌وسیلهٔ یک معلم ارایه می‌شود. (۱ : n)

۲ــ دانش‌آموز (در هر ترم) چندین درس ثبت‌نام می‌کند و یک درس به‌وسیلهٔ چندین دانش‌آموز انتخاب می‌شود. (m: n)

۳ــ یک دانش‌آموز برای هر درس ثبت‌نام شده، یک نمره می‌گیرد. یک نمره به یک درس داده می‌شود. (۱ : n)

**۵ــ۷ــ۱ــ نمودار ارتباط ــ موجودیت /ER**[1] : برای نشان دادن ارتباط بین موجودیت‌ها و بیان عملکرد ارتباط، از نمودار ER استفاده می‌شود. به عبارت دیگر، یک مدل‌سازی معنایی از داده‌هاست.

اجزای نمودار ER عبارت است از :

۱ــ موجودیت‌ها

۲ــ عملکرد ارتباط

۱ــ Entity Relationship Diagram

۳ــ ماهیت ارتباط

۴ــ ویژگی‌های موجودیت

برای ترسیم نمودار، از شکل‌هایی به‌صورت زیر استفاده می‌شود[۱] :

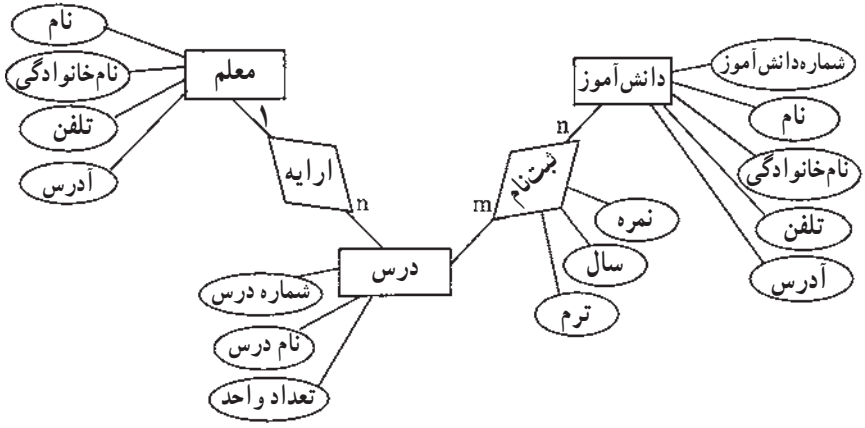
موجودیت‌ها از جنس «اسم» هستند مانند دانش‌آموز و درس. عملکرد ارتباط از جنس «فعل» یا «مصدر» است مانند اخذ کردن. ماهیت ارتباط تناظر بین موجودیت‌ها را نشان می‌دهد، ویژگی‌های موجودیت همان صفات قابل شناسایی هر موجودیت است.

**نکته**

ماهیت ارتباط روی خطوط متصل بین موجودیت‌ها در دو طرف شکل لوزی ذکر می‌گردد.

**مثال ۳ ـ ۱**

نمودار ER محیط عملیاتی مدرسه به‌صورت زیر است :




۱ــ شکل‌ها به‌صورت قراردادی است و در منابع مختلف متفاوت است.

**دقت:** یک عملکرد خود می‌تواند مشخصه‌هایی داشته باشد. در عملکرد ثبت‌نام مشخصه‌های سال، ترم و نمرهٔ درس وجود دارد.

**۶ــ۷ــ۱ــ رابطه / جدول :** مدل رابطه‌ای بر مبنای یک مفهوم ریاضی به‌نام «رابطه»[1] تعریف شده است. استحکام نظریه مجموعه‌ها و سهولت پیاده‌سازی آن باعث رواج این مدل داده‌ای شده است. بر این اساس هر بانک اطلاعاتی از تعدادی رابطه یا همان جدول تشکیل می‌شود.

هر جدول می‌تواند نشان‌دهندهٔ یک موجودیت و یا ارتباط بین موجودیت‌ها باشد. ER مثال ۱ــ۱ را دوباره ملاحظه نمایید. طبق مدل داده‌ای رابطه‌ای جدول‌های زیر را می‌توان درنظر گرفت :

- جدول دانش‌آموز (Student)
- جدول درس (Course)
- جدول معلم (Teacher)

**Course**

| شمارهٔ درس | نام درس | تعداد واحد |
|---|---|---|
| | | |

**Teacher**

| نام | نام‌خانوادگی | تلفن | آدرس |
|---|---|---|---|
| | | | |

**Student**

| شمارهٔ دانش‌آموزی | نام | نام‌خانوادگی | تلفن | نشانی |
|---|---|---|---|---|
| | | | | |

هر جدول با نام خود و مجموعهٔ نام ستون‌هایش مشخص می‌شود. دقت کنید بهتر است در نرم‌افزار اکسس نام جدول و نام ستون‌ها به‌صورت لاتین باشد ولی در این‌جا فارسی آورده شده است.

---

۱ــ رابطه، زیرمجموعه‌ای از حاصل‌ضرب دکارتی مجموعه‌ها است. مفاهیم مرتبط در این نظریه به‌ترتیب عضو، مجموعه حاصل‌ضرب دکارتی، رابطه و تابع است.

**نکته**

محاسن مدل داده‌ای رابطه‌ای :

۱ــ از نظر نمایش، ساده و قابل فهم است و تنها از یک مفهوم اصلی به‌نام رابطه یا جدول استفاده می‌کند.

۲ــ انواع ماهیت‌های ارتباط را پشتیبانی می‌کند.

۳ــ مبنای ریاضی دارد.

هنگامی که طراح، موجودیت‌های مورد نیاز و ویژگی‌های آن‌ها را به همراه ارتباط بین موجودیت‌ها شناسایی کرد، کافی است نام و نوع آن‌ها را در حالت جدولی به پایگاه داده معرفی نماید.

**۷ــ۷ــ۱ــ تبدیل ER به جدول**

ــ هر موجودیت مستقل به یک جدول تبدیل می‌شود.

ــ صفات موجودیت‌ها به فیلدهای جدول مربوطه تبدیل می‌شوند.

ــ موجودیت‌هایی که ارتباط ۱ : ۱ دارند به یک جدول تبدیل می‌شوند. [اگرچه ممکن است تداخل اطلاعاتی رخ دهد]. در مواردی نیز مانند ۱ : n یکی از جدول‌ها به دیگری منتقل می‌شود.

ــ در ارتباط ۱ : n، کلید اصلی جدول طرف ۱ به جدول طرف n اضافه می‌شود.

ــ ارتباط n: m، تبدیل به یک جدول می‌شود که کلید اصلی دو موجودیت را شامل می‌شود.

## ۸ ـ ۱ ـ قدم‌های طراحی بانک اطلاعاتی

در طراحی هر سیستم اطلاعاتی، صرف‌نظر از روش و متدولوژی طراحی باید ابتدا محیط سیستم شناخته و مطالعه شود، سپس نیازسنجی صورت می‌گیرد یعنی تعیین نیازهای اطلاعاتی ــ پردازشی، تشخیص محدودیت‌ها و قواعد حاکم بر محیط.

روش‌های مختلفی برای طراحی وجود دارد. با بیان ساده می‌توان گفت که در طراحی بانک داده مدل‌سازی معنایی داده صورت می‌گیرد یعنی ابتدا طراحی مفهومی انجام می‌شود و نمودار ER رسم می‌گردد و در طراحی سیستم اطلاعاتی، معمولاً ابتدا تحلیل فرایندی صورت می‌گیرد و پردازش‌ها و گردش اطلاعات شناسایی می‌شوند. با مشخص شدن پردازش‌ها، داده‌هایی که از طریق فرم‌ها رد و بدل می‌شوند و تغییراتی که روی آن‌ها انجام می‌شود، مشخص می‌گردند و به این ترتیب ذخایر داده‌ای شکل می‌گیرند و برنامهٔ کاربردی طراحی می‌شود یعنی واسط‌های کاربری مثل فرم‌ها و گزارش‌ها و کنترل‌های

مورد نیاز تعیین می‌شود. در این روش‌ها، پس از انجام طراحی تفصیلی، جدول‌ها، فیلدها و کلید اصلی آن‌ها از نمودار ER و یا ذخایر داده‌ای تعیین می‌شوند. روش تبدیل موجودیت‌ها و رابطه‌ها (نمودار ER) را به جدول‌ها، فیلدها و کلید قبلاً توضیح داده‌ایم.

### برای مطالعهٔ بیشتر

در هر روش و متدولوژی طراحی، مراحل انجام طراحی، اصول و مفروضات مورد استفاده، ابزار و نکات کاربردی و محصولات و نتایج حاصل در هر مرحله و هم‌چنین ارتباط بین مراحل، چگونگی عبور از هر مرحله، روش کنترل مستندات حاصل در هر مرحله، و در برخی متدولوژی‌ها ممیزی، کنترل و حساب‌رسی کیفیت مراحل و محصولات هر مرحله مشخص است. به‌طورکلی سه روش وجود دارد :
فرایندگرا (Process Oreinted)، داده‌گرا (Data Oriented) و شیءگرا (Object Oriented). روش فرایندگرا مانند SSADM، Jackson، Urdon و غیره. داده‌گرا مانند روش Information Engineering (IE) و روش شیءگرا مانند RUP (UML).

### کنجکاوی

**یک روش و متدولوژی طراحی نام برده و خصوصیات آن را به‌طور خلاصه بیان کنید.**

بحث شناخت سیستم و طراحی فرم‌ها و گزارش‌ها به مبحث تجزیه و تحلیل سیستم‌های اطلاعاتی مرتبط است که خود محدودهٔ وسیعی دارد و از بحث ما خارج است.

با این سرآغاز طراحی یک بانک داده را به قدم‌های زیر ساده می‌کنیم :

**۱ــ ابتدا هدف از طراحی پایگاه داده را مشخص کنید.**

مشخص کنید چه نوع اطلاعاتی باید از بانک دریافت شود. با افرادی که از بانک استفاده خواهند کرد صحبت کنید و در مورد فرم‌ها و گزارش‌های مورد نیاز، اطلاعات جمع‌آوری کنید.

**۲ــ نمودار ER را طراحی کنید.**

موجودیت‌ها، ویژگی‌ها و ارتباط بین آن‌ها را شناسایی کنید و نمودار ارتباط ــ موجودیت‌ها

(ER) را ترسیم کنید.

برای هر موجودیت کلید اصلی را تعیین یا ایجاد کنید.

**نکته**

**به‌خاطر بسپارید که یک جدول نباید اطلاعات افزونه داشته باشد یعنی هر جدول باید اطلاعات یک موضوع را نگهداری کند و از اختلاط اطلاعاتی پرهیز شود.**

**۳ــ جدول‌ها و فیلدهای آن‌ها را مشخص کنید.**

نمودار ER ترسیم شده را به جدول‌ها، فیلدهای مربوطه تبدیل کنید. بعدها خواهیم دید که جدول ایجاد شده از نظر سطوح نرمال بررسی شده و ممکن است خود به جدول‌های جدید تجزیه شوند.

**۴ــ فرم‌های مورد نیاز را تعریف کنید.**

فرم‌های ورود، ویرایش و حذف داده را رسم کنید و از وجود فیلدهای فرم در جدول‌های بانک مطمئن شوید.

**۵ــ گزارش‌های مورد نیاز را تعریف کنید.**

گزارش‌های موردنیاز کاربران بانک را معین کرده، قالب نمایش را تعیین کنید. وجود داده‌های لازم و موردنیاز نمایش گزارش در جدول‌های بانک را بررسی نمایید.

**۶ــ طراحی خود را اصلاح و بازنگری کنید.**

یافته‌های خود را با افرادی که از بانک استفاده می‌کنند مطرح کنید و بار دیگر نیازها را بررسی و طراحی خود را اصلاح و بازنگری نمایید.

**کنجکاوی**

**تفاوت‌های طرح خود را با طراحی مثال بعد بررسی و تحلیل کنید.**

**نکته**

دقت کنید که ممکن است طراحی شما با جدول‌های زیر متفاوت باشد. این مسأله نشان‌دهندهٔ غلط بودن طراحی نیست. بلکه برحسب مفروضات، سطح نرمال بودن جدول‌ها[1] و محدودهٔ سیستم کاملاً طبیعی است.

۱ــ با نرمال‌سازی در فصل ۷ آشنا خواهید شد.

مثال ۴-۱

محیط عملیاتی مدرسه را طراحی کنید.

نمودار ER رسم شده است.

۱ــ نمودار ER به جدول‌های زیر تبدیل می‌شود :

شکل [ ] نشان‌دهندهٔ کلید اصلی و شکل ( ) نشان‌دهندهٔ کلید خارجی اضافه شده از جدول دارای ارتباط است.

**جدول معلم Tbl–Teacher**

| Teacher Code | Teacher FName | Teacher LName | Teacher Tel | TeacherAddress |
|---|---|---|---|---|
| کد معلم | نام | نام‌خانوادگی | تلفن | آدرس |

**جدول دانش‌آموز Tbl–Student**

| Student Code | Student FName | Student LName | Student Tel | StudentAddress | Year |
|---|---|---|---|---|---|
| شمارهٔ دانش‌آموزی | نام | نام‌خانوادگی | تلفن | آدرس | سال |

* در جدول درس، فیلد کد معلم را که کلید اصلی در جدول معلم است به عنوان کلید خارجی اضافه می‌کنیم.

**جدول درس Tbl–Course**

| Course Code | Course FName | Course Unit | Teacher Code |
|---|---|---|---|
| شمارهٔ درس | نام درس | تعداد واحد | کد معلم |

* ارتباط ۱ : n بین درس و معلم، کلید جدول معلم (کد معلم) به جدول درس اضافه می‌شود. دقت کنید که کد معلم در جدول درس بخشی از کلید اصلی نیست و کلید خارجی به‌حساب می‌آید.

**جدول ثبت‌نام Tbl–Register**

| Year | Term | Course Code | Student Code | CourseGrade |
|---|---|---|---|---|
| سال | ترم | شمارهٔ درس | شمارهٔ دانش‌آموز | نمرهٔ درس |

* ارتباط n: m بین دانش‌آموز و درس به جدول ثبت‌نام تبدیل می‌شود و کلید اصلی جدول درس (شمارهٔ درس) و کلید اصلی جدول دانش‌آموز (شمارهٔ دانش‌آموز) به آن اضافه می‌شود.

چون ارتباط n: m مفروض خود دارای ویژگی‌هایی است پس این دو کلید به‌تنهایی کلید اصلی آن جدول نیستند و ترکیب آن دو با فیلد سال و ترم جدول کلید اصلی شده است.

با توجه به اصول و مفروضات طراحی که در بخش نرمال‌سازی خواهیم دید، جدول ثبت‌نام به دو جدول زیر تبدیل می‌شود :

**جدول ثبت‌نام Tbl–Register**

| Register Year | Term | Student Code | Register Code |
|---|---|---|---|
| سال | ترم | شمارهٔ دانش‌آموز | سریال ثبت‌نام |

* ما شمارهٔ سریال ثبت‌نام را به‌عنوان کلید اصلی اضافه می‌کنیم. می‌توانستیم سال و ترم را هم کلید بگیریم ولی بهتر است از کلید ترکیبی به علت پیچیدگی‌هایی که ایجاد می‌کنند استفاده نکنیم.

**جدول دروس ثبت‌نام شده Tbl–Register**

| Register Code | Course Code | Greade |
|---|---|---|
| سریال ثبت‌نام | شمارهٔ درس | نمرهٔ درس |

* دقت کنید که در جدول دروس ثبت‌نام شده شمارهٔ درس بخشی از کلید اصلی است.

مثال ۴-۱

۲ــ فرم‌های مورد نیاز :

**فرم تعریف معلم**

کد معلم :
نام :
نام‌خانوادگی :
تلفن :
آدرس :

**فرم تعریف درس**

شمارهٔ درس :
نام درس :
تعداد واحد :
کد معلم :

**فرم تعریف دانش‌آموزان**

شمارهٔ دانش‌آموز :
نام :
نام‌خانوادگی :
تلفن :
آدرس :
سال :

**فرم ورود نمره**

سال : ترم : نام دانش‌آموز :

| شمارهٔ درس | نام درس | تعداد واحد | نمرهٔ نهایی |
|---|---|---|---|
| | | | |
| | | | |
| | | | |
| | | | |

**فرم ثبت‌نام**

سال : ترم : نام دانش‌آموز :

| شمارهٔ درس | نام درس | تعداد واحد |
|---|---|---|
| | | |
| | | |
| | | |
| | | |

۳ـ گزارش‌های مورد نیاز :

**گزارش لیست معلمان مدرسه**

| کد معلم | نام معلم | تلفن | آدرس |
|---|---|---|---|
| | | | |
| | | | |
| | | | |
| | | | |

**گزارش ثبت‌نام**

سال : ترم :

نام دانش‌آموز : شماره دانش‌آموز :

دروس ثبت‌نام شده

| شمارهٔ درس | نام درس | تعداد واحد | نام معلم |
|---|---|---|---|
| | | | |
| | | | |
| | | | |

**گزارش کارنامه دانش‌آموزان**

نام و نام‌خانوادگی :

تاریخ گزارش :

رشته : سال : ترم :

| شمارهٔ درس | نام درس | تعداد واحد | نمره |
|---|---|---|---|
| | | | |
| | | | |
| | | | |

جمع واحد :

معدل :

**گزارش لیست حضور و غیاب**

سال : ترم :

نام معلم : درس :

تعداد واحد : ساعت تشکیل کلاس :

| شمارهٔ دانش‌آموز | نام دانش‌آموز | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |

## خلاصهٔ فصل

سیستم ذخیره و بازیابی از مفهوم سادهٔ فایلینگ توسعه پیدا کرده است. فیلد کوچک‌ترین واحد ذخیرهٔ داده است. رکورد مجموعهٔ فیلدهای مرتبط است و مجموعهٔ رکوردها فایل را تشکیل می‌دهند. این مفاهیم پایه‌ای در هر سیستم اطلاعاتی دیگر نیز مطرح‌اند. فیلد یا زیرمجموعه‌ای از فیلدها که باعث یکتایی رکورد شوند را کلید گویند.

در سیستم‌های ذخیره و بازیابی، عملیات مشخصی روی رکوردها انجام می‌گیرد که عبارت است از بازیابی، درج، حذف، به‌هنگام‌سازی و تغییر ساختار. این عملیات را در محیط Access بیشتر بررسی می‌کنیم.

برای جلوگیری از درگیر شدن کاربر با مسایل برنامه‌نویسی و انجام سریع‌تر و دقیق‌تر تغییرات و در مجموع برای برقرار کردن ارتباط بین حجم بالای داده‌های ذخیره شدهٔ روی رسانه و کاربران معماری سه لایه‌ای مطرح شده است : سطح خارجی، سطح مفهومی، سطح داخلی.

نرم‌افزاری که این ارتباط را برقرار، مدیریت و اداره می‌کند، سیستم مدیریت بانک اطلاعاتی نام دارد که برای مدیریت دستیابی به داده‌ها و پنهان کردن آن از دید کاربران از یک مدل مفهومی به‌نام مدل داده‌ای استفاده می‌کند. مدل رابطه‌ای از جمله مدل‌های داده‌ای مهم است که بر پایه مفهوم ریاضی مجموعه‌ها بنا شده است و بیشتر سیستم‌های مدیریت بانک اطلاعاتی رایج از آن پشتیبانی می‌کنند.

در این فصل مفاهیم پایه مدل دادهٔ رابطه‌ای مانند : رابطه/جدول، ارتباط، ماهیت ارتباط، نمودار ارتباط ــ موجودیت (ER) و روش تبدیل این نمودار به جدول‌های پایگاه داده مطرح شده است.