فصل ۷

نرمال‌سازی

پس از شناسایی موجودیت‌ها و رسم ER، صفات مورد نیاز و ارتباط بین موجودیت‌ها، می‌باید جدول‌های (روابط) مورد نیاز طراحی شوند. یک تصور نادرست این است که هرچه صفات جدول بیشتر باشد، از اکمال و جامعیت بالاتری برخوردار است، حال آن که گزینش درست صفات، از بروز برخی مشکلات در بانک‌های اطلاعاتی اجتناب می‌کند. نارسایی‌ها در طراحی و یک طراحی نامناسب، باعث ایجاد پدیدهٔ «آنومالی»[۱] می‌گردد.

آنومالی، در سه محور قابل طرح و بررسی است:

الف ـ «انجام‌ناپذیری» یکی از عملیات در بانک.

ب ـ «بروز تبعات نامطلوب» در پی انجام یک عملیات مبنایی.

ج ـ «فزون‌کاری» برای انجام یک عملیات مبنایی.

قبل از شرح موارد آنومالی، ویژگی‌های رابطه و چند مفهوم مقدماتی دیگر ذکر می‌شود، سپس در قالب یک مثال آنومالی و راه‌های رفع آن گفته خواهد شد.

هدف‌های رفتاری: پس از آموزش این فصل هنرجو می‌تواند:

- **ویژگی‌های رابطه را تعریف کند.**
- **وابستگی تابعی و وابستگی تابعی کامل را شرح دهد.**
- **آنومالی را تعریف کند.**
- **ویژگی‌های تجزیهٔ مناسب را شرح دهد.**
- **مفهوم سطوح نرمال 1NF، 2NF و 3NF را توضیح دهد.**

۱ـ Anomaly

۱ـ۷ـ ویژگی‌های رابطه

رابطه که در بانک‌های اطلاعاتی رابطه‌ای[۱]، به‌صورت جدول پیاده‌سازی می‌شود دارای ویژگی‌های زیر است :

الف ــ منظم نیست، به این معنی که ترتیب صفات، در هنگام تعریف بی‌اهمیت است و سطرهای جدول[۲] نظم ندارد.

ب ــ دارای صفات تک‌مقداری است یعنی به اجزای معنی‌دار تقسیم نمی‌شود. به‌عنوان مثال اگر مقدار شمارهٔ دانش‌آموزی برابر ۸۵۱۷۴۳ باشد و ۸۵ نشان‌دهندهٔ سال تحصیلی باشد و در پرس‌وجوها بخواهیم دانش‌آموزانی که در سال ۱۳۸۵ ثبت‌نام کرده‌اند را مشخص کنیم، فیلد «شمارهٔ دانش‌آموزی» یک فیلد تک‌مقداری نیست.

ج ــ جدول، سطرهای تکراری ندارد.

۲ـ۷ـ وابستگی تابعی[۳]

شبیه تعریف تابع در ریاضیات، در رابطهٔ R، صفت y با صفت x[۴] وابستگی تابعی دارد اگر به‌ازای هر مقدار x، تنها یک مقدار y وجود داشته باشد.

وابستگی تابعی y با x را به‌صورت $x \longrightarrow y$ نشان می‌دهیم[۵].

مثال ۱ـ۷

در رابطهٔ R۱، وابستگی تابعی $A \longrightarrow B$ و $B \longrightarrow C$ را بررسی کنید :

R۱

A	B	C
a_1	b_1	c_1
a_2	b_2	c_3
a_1	b_1	c_2
a_3	b_4	c_2
a_5	b_1	c_1

۱ــ Relational Data Base

۲ــ به سطرهای جدول در این مدل داده‌ای اصطلاحاً تاپل گفته می‌شود.

۳ــ Functional Dependency

۴ــ x و y می‌توانند صفات خاصهٔ ساده یا مرکب باشند.

۵ــ A را اصطلاحاً دترمینان گویند.

مثال ۱-۷

پاسخ: وابستگی $A \longrightarrow B$ برقرار است. زیرا به‌ازای هیچ مقدار مساوی از A، دو مقدار متفاوت از B وجود ندارد. اما وابستگی $B \longrightarrow C$ برقرار نیست. زیرا که به‌ازای b_1، دو مقدار c_1 و c_2 در R وجود دارد. $(B \longrightarrow C)$

وابستگی تابعی بیان‌کنندهٔ قواعد محیط عملیاتی است. این قواعد را می‌توان در نمودار «وابستگی تابعی» یا نمودار FD نمایش داد.

مثال ۲-۷

فرض کنید قواعد زیر در یک محیط عملیاتی برقرار است. نمودارFD متناظر را رسم کنید. یک جدول با مقادیر دلخواه متناظر و با نامFIRST ایجاد کنید.

قاعدهٔ ۱ : هر ناشر، تعدادی کتاب منتشر می‌کند.

قاعدهٔ ۲ : هر ناشر از یک کتاب، شمارگان مشخصی منتشر می‌کند.

قاعدهٔ ۳ : هر ناشر در یک شهر دفتر دارد.

قاعدهٔ ۴ : هر ناشر دارای یک رتبهٔ صنفی است.

قاعدهٔ ۵ : ناشران یک شهر دارای یک رتبهٔ صنفی هستند.

پاسخ : با توجه به قواعد بالا داریم :

(P#= شماره ناشر، Book#= شماره کتاب، Qty=شمارگان، City= شهر و Grade= رتبه صنفی ناشر)

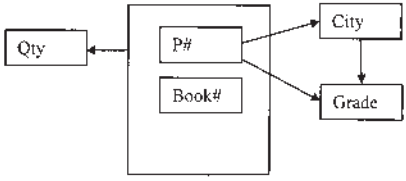
1) $P\# \not\rightarrow Book\#$

2) $(P\# , Book\#) \rightarrow Qty$

3) $P\# \rightarrow City$

4) $P\# \rightarrow Grade$

5) $City \rightarrow Grade$



P#	Book#	Qty	City	Grade
P_1	b_1	1000	c_1	40
P_2	b_4	3000	c_2	60
P_3	b_3	2000	c_1	40
P_1	b_2	3000	c_1	40
P_4	b_3	4000	c_3	50
P_4	b_5	3500	c_3	50

تمرین ۱ـ۷ : در یک محیط عملیاتی چهار قاعده بیان کنید و نمودار FD متناظر را رسم کنید.

نکته

تمام صفات یک موجودیت با صفت کلید آن، وابستگی تابعی دارند.

مثال ۳ـ۷

اگر $A \longrightarrow B$ و $B \longrightarrow C$ بر قرار باشد، آیا $A \longrightarrow C$ برقرار است؟

پاسخ : فرض می‌کنیم $A \not\rightarrow B$ در این صورت داریم :

$a_1 \ldots c_1$

$a_1 \ldots c_2$

دو حالت روبه‌رو قابل تصور است : $a_1\ b_1\ c_1$ $\quad$ $a_1\ b_1\ c_1$

$a_1\ b_1\ c_2$ $\quad$ $a_1\ b_2\ c_2$

در هر دو حالت با تناقض در فرض مواجه می‌شویم. در قسمت (الف) $B \not\rightarrow C$ و در قسمت (ب) $A \not\rightarrow B$ درنتیجه طبق برهان خلف داریم : $A \longrightarrow C$.

۳ـ۷ـ وابستگی تابعی کامل[۱]

صفت y، با صفت x وابستگی تابعی کامل دارد ($x \Longrightarrow y$) اگر :

۱ـ y با x وابستگی تابعی داشته باشد.

۲ـ y با هیچ زیرمجموعهٔ x وابستگی تابعی نداشته باشد.

۱ـ Fully Functional Dependency

مثال ۲ـ۷

با توجه به R2 تحقیق کنید، آیا وابستگی تابعی کامل بین C و (A و B) برقرار است؟

$R^{۲}$

A	B	C
a_1	b_1	c_1
a_2	b_1	c_3
a_1	b_2	c_2

پاسخ : شرط اول برقرار است. یعنی : $(A,B) \longrightarrow C$

شرط دوم نیز برقرار است. زیرا : $A \not\rightarrow C\ ((a_۱ , c_۱) , (a_۱ , c_۲))$

$B \not\rightarrow C\ ((b_۱ , c_۱) , (b_۱ , c_۲))$

درنتیجه : $(A,B) \Longrightarrow C$

نکته

اگر x صفت ساده باشد و وابستگی تابعی بین x و y برقرار باشد $(x \longrightarrow y)$، وابستگی تابعی کامل $(x \Longrightarrow y)$ همواره برقرار است.

۴ـ۷ ـ شرح آنومالی‌ها

همان‌طورکه گفتیم آنومالی سه وجه دارد. اینک با درنظرگرفتن رابطهٔ مثال ۲ـ۷ آنومالی را توضیح می‌دهیم.

۱ـ درج : در آنومالی ناشی از درج، تمام یا بخشی از کلید اصلی تعریف‌نشده و نامعین است. طبق یکی از قواعد عام بانک اطلاعاتی، مقدار کلید اصلی باید کاملا مشخص باشد.

فرض کنید می‌خواهیم رکورد $<P_{10}, 1000, c_4, 10>$ را درج کنیم. این درج امکان‌پذیر نیست. زیرا مشخص نیست ناشر چه کتابی را منتشر کرده است.

۲ـ حذف : در آنومالی نوع دوم پس از انجام یک عمل، عوارض نامطلوب داریم. فرض کنید می‌خواهیم رکورد $<P_2, b_4, 3000>$ را حذف کنیم. این حذف اگرچه شدنی است اما رتبهٔ صنفی ناشران

شهر c_2 (مقدار ۶۰) ناخواسته حذف می‌شود (دقت کنید فعلًا تنها یک ناشر ساکن شهر c_2 است).

۳ــ به‌هنگام‌سازی : این نوع آنومالی، موجب فزون‌کاری می‌شود. در فزون‌کاری به‌ازای یک عمل مبنایی چندین عملیات صورت می‌گیرد. فرض کنید قرار است رتبهٔ صنفی ناشران ساکن شهر c_3 از ۵۰ به ۷۰ تغییر کند. این عمل موجب به‌هنگام‌سازی منتشرشونده[۱] (فزون کاری) در سیستم می‌شود، یعنی در بیش از یک سطر جدول باید تغییر صورت گیرد.

همان‌طورکه مشاهده می‌شود، رابطهٔ FIRST (مثال۲ــ۷) دارای آنومالی است. در یک بیان غیردقیق، علت آنومالی‌ها، «اختلاط اطلاعاتی» است. به این معنی که اطلاعات نشر و اطلاعات پایه‌ای ناشر در یک رابطه (جدول) با هم جمع شده است. برای ایجاد یک طراحی مناسب از رابطه‌ها باید آن‌ها را از لحاظ سطوح نرمال بررسی کنیم و در صورت نرمال نبودن در آن سطح، آن رابطه را با «تجزیه» اصلاح نماییم. در این صورت از بروز آنومالی جلوگیری می‌شود. اینک مناسب است سطوح نرمال رابطه را با بیان دقیق‌تر مورد بررسی قرار دهیم.

۵ـ۷ـ سطوح نرمال

روابط از سطح غیرنرمال تا سطح نرمال ۵ قابل تبیین و بررسی است. سطوح نرمال عبارتند از :

۱ــ سطح نرمال اول 1NF

۲ــ سطح نرمال دوم 2NF

۳ــ سطح نرمال سوم 3NF

۴ــ سطح نرمال BCNF[۲]

۵ــ سطح نرمال چهارم 4NF

۶ــ سطح نرمال پنجم 5NF

در این کتاب تا سطح نرمال 3NF را بررسی می‌کنیم. چرا که رابطه در این سطح، از بسیاری آنومالی‌ها جلوگیری می‌کند.

۱ــ۵ــ۷ــ سطح نرمال اول : رابطهٔ FIRST در مثال ۲ــ۷ را در نظر بگیرید. تمام فیلدها به نحوی هستند که مقادیر آن‌ها به‌صورت منطقی قابل تقسیم نیست. به هر یک از این فیلدها، یک فیلد تک‌مقداری گوییم. اگر تمام فیلدها تک‌مقداری باشد، رابطه در سطح نرمال اول قرار دارد.

۱ــ Propagated Update

۲ــ به احترام نام Boyce و Codd نامیده شده است. Boyce-Codd Normal Form(BCNF)

رابطهٔ R در سطح نرمال اول (1NF) است، اگر تمام فیلدهای آن در هر سطر جدول، تک‌مقداری باشد.

مثال ۵ ـ ۷

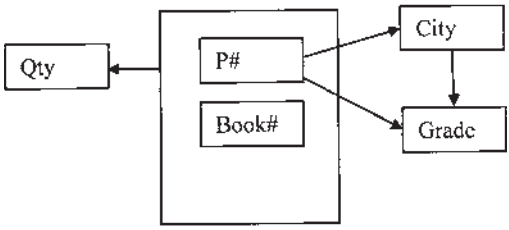
رابطهٔ کارمند را در نظر بگیرید، آیا این رابطه در سطح 1NF است؟

کد پرسنلی	میزان تحصیلات	شماره شناسنامه	نام
۱۱۰	دیپلم	۲۴۳	علی علوی
۱۴۰	کارشناسی	۲۷۱۹	ستایش یمقانی
۱۳۰	کاردانی	۵۹۳	رضا قاسمیان
۱۷۰	کارشناسی	۹۰۹	اسماعیل میرزایی
۱۹۰	کارشناسی	۷۱۴	علیرضا مسچی

پاسخ : فیلد نام، قابل تجزیه به دو فیلد نام و نام خانوادگی است. اگر بخواهیم نام‌خانوادگی کارمندان را از رابطه به‌دست آوریم، فیلد نام را تجزیه کرده‌ایم؛ بنابراین رابطه غیرنرمال است و در سطح 1NF قرار ندارد. شکل نرمال 1NF آن به‌صورت زیر است :

کد پرسنلی	میزان تحصیلات	شماره شناسنامه	نام خانوادگی	نام
۱۱۰	دیپلم	۲۴۳	علوی	علی
۱۴۰	کارشناسی	۲۷۱۹	یمقانی	ستایش
۱۳۰	کاردانی	۵۹۳	قاسمیان	رضا
۱۷۰	کارشناسی	۹۰۹	میرزایی	اسماعیل
۱۹۰	کارشناسی	۷۱۴	مسچی	علیرضا

۲ــ۵ــ۷ــ سطح نرمال دوم : در قسمت‌های قبل با تعریف وابستگی تابعی کامل آشنا شدیم.
رابطه وقتی در سطح 1NF باقی می‌ماند که وابستگی تابعی کامل بین حداقل یک فیلد با کلید اصلی نقض شود. به‌عنوان مثال نمودار FD برای رابطهٔ FIRST را در نظر بگیرید :



کلید اصلی در این رابطه، صفت مرکب (P#, Book#) است. از آن‌جایی که تمام فیلدها، با کلید FD دارند، داریم : (P# , Book#) ⟶ City

برای برقراری وابستگی تابعی کامل باید داشته باشیم :

P# ↛ City

Book# ↛ City

اما همان‌طورکه در نمودار FD قابل مشاهده است، داریم : P# ⟶ City

درنتیجه FD کامل بین City و (Book#, P#) نقض شده است.

اینک می‌توانیم دلیل آنومالی‌های رابطهٔ FIRST را بهتر دریابیم. نقض FD کامل.

رابطهٔ R در سطح نرمال دوم (2NF) است، اگر:

اولاً: 1NF **باشد.**

ثانیاً: صفت غیرکلید، ویژگی وابستگی تابعی کامل را با کلید نقض نکند.

برای رفع آنومالی و افزایش سطح نرمال رابطه، می‌باید رابطهٔ FIRST را تجزیه نماییم. برای این کار FIRST را به‌صورت زیر تجزیه می‌کنیم :

SECOND (P#, City, Grade)

PB (P#, Book#, Qty)

SECOND

P#	City	Grade
P_1	c_1	40
P_2	c_2	60
P_3	c_1	40
P_4	c_3	50

PB

P#	Book#	Qty
P_1	b_1	1000
P_2	b_4	3000
P_3	b_3	2000
P_1	b_2	3000
P_4	b_3	4000
P_4	b_5	3500

ملاک تجزیهٔ رابطه : تجزیهٔ رابطه R به رابطه‌های R_1 و R_2 می‌باید به نحوی باشدکه پیوند[1] دو رابطهٔ R_1 و R_2 رابطهٔ R را ایجاد کند و تاپلی (رکورد یا سطری) کم و زیاد نشود. از طرف دیگر تجزیهٔ R می‌باید، وابستگی‌های تابعی را حفظ کند.

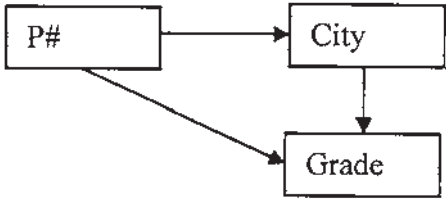
تمرین۲ـ۷ : آیا معیارهای تجزیهٔ مناسب در تجزیهٔ FIRST لحاظ شده است؟

رابطهٔSECOND ، در سطح دوم نرمال است. زیرا اولاً : 1NF است و ثانیاً : وابستگی تابعی کامل نقض نشده است. اما این رابطه نیز دارای آنومالی می‌باشد.

تمرین۳ـ۷ : آنومالی‌های SECOND را ذکر کنید.

۳ــ۵ــ۷ــ سطح نرمال سوم : رابطهٔ SECOND را در نظر می‌گیریم. نمودار FD آن به‌صورت زیر است :

1. P# ⟶ City
2. City ⟶ Grade
3. P# ⟶ Grade



همان‌طورکه در مثال ۳ــ۷ دیدیم، از دو وابستگی تابعی اول و دوم، می‌توان وابستگی تابعی سوم را نتیجه گرفت، یعنی Grade وابستگی تابعی با واسطه با P# از طریق City دارد. علت بروز آنومالی‌های SECOND نیز همین عامل یعنی وابستگی تابعی باواسطه است.

رابطهٔ R در سطح نرمال سوم (3NF) است. اگر:
اولاً : 2NF **باشد.**
ثانیاً : هر صفت غیرکلید با کلید اصلی، وابستگی تابعی باواسطه نداشته باشد.

۱ــ Join

برای رفع آنومالی‌های آن، می‌باید این رابطه تجزیه شود. بنابراین SECOND را به رابطه‌های PC و CG تجزیه می‌کنیم :

PC(P#,City)

CG (City, Grade)

PC

P#	City
P_1	c_1
P_2	c_2
P_3	c_1
P_4	c_3

CG

City	Grade
c_1	40
c_2	60
c_3	50

با حذف آنومالی‌های SECOND، روابط جدید نه‌تنها در سطح 3NF قرار می‌گیرند بلکه در سطح بالاتر یعنی BCNF نیز قرار می‌گیرند که ما به بررسی آن نمی‌پردازیم.

تمرین ۴ـ۷:

۱ــ آیا تجزیهٔ انجام‌شده، ضوابط یک تجزیهٔ مناسب را دارا می‌باشد؟

۲ــ حالت‌های دیگر تجزیهٔ SECOND را در نظر بگیرید. آیا این تجزیه‌ها مناسب هستند؟

اگرچه نرمال‌سازی، آنومالی‌ها را از بین می‌برد اما در پرس‌وجوها، به علت نیاز به پیوند رابطه‌ها (جدول‌ها) سرعت انجام پرس‌وجو کاهش می‌یابد.

کنجکاوی

پس از اعمال سطوح نرمال و تجزیهٔ جدول‌ها، سرعت بازیابی اطلاعات مورد نیاز کاربران بررسی می‌شود. در صورت عدم کارایی پرس‌وجوهای مورد نظر، و بنابر ملاحظاتی ممکن است جدول‌ها مجدداً ترکیب گردند. چه ملاحظاتی در این مورد مطرح است؟

ادگار فرانک تد کاد (۲۳ آگوست ۱۹۲۳ــ ۱۸ آوریل ۲۰۰۳)

کاد یک دانشمند علوم کامپیوتر بود. وی نقش به سزایی در بنیان‌گذاری تئوری بانک رابطه‌ای داشت و می‌دانیم که این مدل رایج‌ترین مدل داده‌ای بانک اطلاعاتی است. زمانی که برای IBM فعالیت می‌کرد، مدل رابطه‌ای را برای مدیریت بانک اطلاعاتی بنا کرد. اگر چه نقش ارزشمند بسیاری در سایر پیشرفت‌های علوم کامپیوتر داشت ولی مدل رابطه‌ای وی از تئوری‌های مهم و به یادماندنی او است.

کاد در پورتلند انگلستان متولد شد. او ریاضی و شیمی را در دانشکدهٔ اکستر و آکسفورد گذراند. در سال ۱۹۴۸، به نیویورک رفت و به عنوان برنامه‌نویس با شرکت IBM کار کرد. کاد دکترای علوم کامپیوتر خود را از دانشگاه میشیگان دریافت کرد و دو سال بعد مجدداً در دفتر پژوهشی IBM فعالیت خود را آغاز کرد.

دو دهه (۱۹۶۰ و ۱۹۷۰) را بر روی تئوری‌های مدیریت داده کار کرد و در سال ۱۹۷۰ مقالهٔ خود را در مورد مدل رابطه‌ای برای بانک‌های دادهٔ مشترک و بزرگ[۱] منتشر کرد.

کاد بر توسعه و بسط مدل رابطه‌ای خود ادامه داد و با کریس دیت[۲] همکاری خود را شروع کرد. بر این اساس پس از کاد، یکی از مدل‌های نرمال را به نام Boyce-Codd نامیدند که به BCNF معروف است. پس از دهه ۱۹۸۰ که مدل رابطه‌ای به اوج شهرت رسید، بحث‌های کاد با تولیدکنندگان بانک اطلاعاتی منجر به پیشنهاد اصول ۱۲ گانهٔ وی شد که یک پایگاه دادهٔ رابطه‌ای باید از آن‌ها پیروی کند. با ایجاد زبان SQL، به نظر وی تئوری رابطه‌ای به نادرستی پیاده‌سازی شد و مخالفت‌هایش، پست و مقام شغلی وی را در IBM با مشکل مواجه کرد که شرکت را ترک گفت. پس از آن وی با کمک کریس دیت و دیگر همراهانش، شرکتی مشاوره‌ای راه‌اندازی کرد.

کاد جایزهٔ تورینگ را در سال ۱۹۸۱ دریافت کرد و در سال ۱۹۹۴ وی به‌عنوان عضو افتخاری انجمن ماشین‌های محاسب منصوب شد.

۱ــ "A Relational Model of Data for Large Shared Data Banks"

۲ــ Chris Date

کریستوفر جی دیت ۱۹۴۱

دیت یکی از نویسندگان، محققان، مشاوران و متخصصان مستقل در زمینهٔ فن‌آوری بانک دادهٔ رابطه‌ای است. زمانی که برای شرکت IBM کار می‌کرد، در زمینهٔ برنامه‌ریزی و طراحی فنی محصولات شرکت چون SQL/DS و DB2 همکاری داشت. او زمانی با کاد در مدیریت پایگاه داده‌ها و مدل رابطه‌ای همکاری می‌کرد. در سال ۱۹۸۳ شرکت IBM را ترک کرد و یکی از توسعه‌دهندگان و پایه‌گذاران مدل رابطه‌ای شناخته می‌شود.

کتاب وی به عنوان مقدمه‌ای بر سیستم‌های بانک داده، تاکنون به هشتمین ویرایش خود رسیده است و کتابی مرجع در زمینهٔ بانک‌های داده‌ای است و در صدها دانشکده و دانشگاه در سراسر دنیا تدریس می‌شود. وی همچنین نویسندهٔ کتاب‌های بسیاری در زمینهٔ مدیریت داده است.

خلاصهٔ فصل

پس از طراحی مفهومی بانک اطلاعاتی و رسم نمودار ER، جداول، فیلدها و کلید اصلی آن‌ها تعیین می‌شوند. در برخی موارد جدول‌های ایجادشده دارای آنومالی هستند. یعنی برخی عملیات حذف، اضافه، ویرایش روی آن‌ها غیرممکن می‌شود، یا در صورت انجام اطلاعات مهم از دست می‌رود و یا با دشواری همراه است.

یکی از دلایل ایجاد آنومالی اختلاط اطلاعاتی است. شش سطح نرمال داریم که سه سطح آن را بررسی کرده‌ایم. بیشتر جدول‌های نرمال در سطح سوم، در سطح بالاتری نیز قرار می‌گیرند. پس از تعیین جدول‌ها، هر جدول از نظر سطح نرمال بررسی شده و در صورت لزوم تجزیه می‌شود. هر سطح نرمال دارای شرایطی است.

هر جدول غیرنرمال برای نرمال‌شدن باید تجزیه شود. عمل تجزیه ملاک‌هایی دارد که به برخی از آن‌ها در فصل اشاره کرده‌ایم. تجزیه‌شدن جدول به جدول‌های جدید که حافظ اطلاعات جدول قبلی باشد، آنومالی‌ها را از بین می‌برد ولی در مواردی سرعت بازیافت اطلاعات در پرس‌وجوها را کاهش می‌دهد.

خودآزمایی

۱ـ آنومالی به چه معنی است؟

۲ـ وابستگی تابعی و وابستگی تابعی کامل را با ذکر مثال توضیح دهید.

۳ـ سطح نرمال اول، دوم و سوم چه ویژگی‌هایی دارند؟

۴ـ محاسن و معایب نرمال‌سازی چیست؟

۵ـ دلیل دیگری ذکر کنید که نشان دهد وابستگی تابعی کامل بین صفت خاصهٔ غیرکلید و کلید اصلی در FIRST ، نقض شده است.

۶ ـ آیا می‌توان از وابستگی تابعی $A \longrightarrow D$ عبارت $(A,C) \longrightarrow D$ را نتیجه گرفت؟

۷ـ رابطه Z را درنظر بگیرید و آن را به دو رابطهٔ $X(A,B)$ و $Y(A,C)$ تجزیه کنید. آیا پیوند (JOIN) آن‌ها رابطهٔ Z را ایجاد می‌کند؟

Z

A	B	C
$a_۱$	$b_۱$	$c_۱$
$a_۱$	$b_۲$	$c_۲$
$a_۲$	$b_۱$	$c_۳$
$a_۳$	$b_۳$	$c_۴$

۸ـ فرض کنید در رابطهٔ $(a_۱, a_۲, \ldots, a_۶)$، وابستگی‌هایی تابعی زیر وجود دارد. به نظر شما کلید اصلی چیست؟ چرا؟

$a_۱ \longrightarrow a_۲$

$a_۱ \longrightarrow a_۳$

$a_۱ \longrightarrow a_۴$

$a_۱ \longrightarrow a_۵$

$a_۱ \longrightarrow a_۶$

۹ـ در رابطهٔ $R(A,B,C,D)$ وابستگی‌های تابعی زیر وجود دارد و (A,B) کلید است سطح نرمال رابطه را مشخص کنید.

$C \longrightarrow D$

$A \longrightarrow C$

فصل ۸

مطالعهٔ موردی

در فصل‌های قبل با امکانات محیط بانک اطلاعاتی آشنا شدیم. حال می‌خواهیم با یک مثال از این امکانات برای ایجاد یک سیستم بانک اطلاعاتی استفاده کنیم.

مهدکودکی علاقه‌مند به ایجاد یک پایگاه داده برای مدیریت اطلاعات کودکان تحت حمایت خود است. والدین، نگهداری روزانهٔ کودک خود را به این مؤسسه محول کرده‌اند. روزانه غذاهای متنوعی توزیع می‌گردد، با این حال باید بدانیم آیا کودکی وجود دارد که نتواند از این تنوع غذایی استفاده کند و در این صورت، باید تدارک این گونه استثنائات نیز دیده شود. بنابراین اطلاعات نیاز روزانهٔ غذایی و رژیمی کودکان باید ذخیره گردد، مشخصات فردی کودکان و والدین آن‌ها باید نگهداری شود و همچنین برای بهبود فعالیت مهدکودک، می‌توانیم زمان واکسیناسیون، پایش رشد (کنترل قد و وزن) را نیز نگهداری و برآورد کنیم و به والدین اطلاع دهیم و یادآوری کنیم.

هدف‌های رفتاری: پس از آموزش این فصل هنرجو می‌تواند:

- **یک بانک اطلاعاتی مطابق تجزیه و تحلیل انجام شده را ایجاد کند.**
- **در ایجاد بانک سطح نرمال‌سازی 3FN را رعایت کند.**
- **بانک اطلاعاتی تحلیل شده را توجیه کند.**

۸ـ۱ ـ اطلاعات کودکان Children

زمانی که یک کودک برای اولین بار به مهد کودک آورده می‌شود،یک مشخصه منحصر به فرد می‌گیرد. هر کودک در یک تخت مخصوص به خود و در اتاقی خاص نگهداری می‌شود.

کودکان نوپا می‌توانند از انواع غذاها استفاده کنند ولی نوزادان به برنامهٔ غذایی خاص نیاز دارند. همچنین برخی کودکان ممکن است به غذاهای خاصی حساسیت داشته باشند و نتوانند برخی انواع موادغذایی مثل توت‌فرنگی، تخم‌مرغ، گوجه‌فرنگی و ... را مصرف کنند. همچنین اطلاعات واکسیناسیون و آخرین تاریخ پایش رشد کودکان نیز باید نگهداری شود.

۸ـ۲ ـ اطلاعات والدین Parents

هر یک از والدین می‌تواند مراقبت یک یا چند کودک خود را به مهد کودک بسپارد. برخی والدین علاقه‌مند هستند کودک خود را در اتاق غیرخصوصی و مشترک نگهداری کنند تا با کودکان دیگر مراوده داشته باشند و همچنین هزینهٔ آن ارزان‌تر باشد.

۸ـ۳ ـ تحلیل

مهارت و دانشی که در زمان تحلیل موردنیاز است دانش کامپیوتری فرد طراح نیست، بلکه دانش درک مسأله است. توجه ما در این کتاب، تحلیل و شناخت سیستم نیست، بنابراین فرض می‌کنیم تحلیل قبلاً انجام شده است تا بتوانیم کاربرد Access را با هم بیاموزیم.

۱ـ۳ـ۸ـ موجودیت‌ها Entity : این اصطلاح برای هر چیزی به‌کار می‌رود که باید اطلاعاتی در مورد آن ذخیره کنیم. موجودیت‌های دنیای واقع به جدول‌های بانک تبدیل می‌شوند.

در مثال، دو جدول Parents و Children قابل شناسایی است.

۲ـ۳ـ۸ـ مشخصه‌ها Attribute : جزییات بیشتر یک موجودیت را در بر دارد.

ـ مشخصهٔ Parents عبارت است از : اطلاعات تماس والدین، شناسهٔ منحصر به فرد که به هر یک از اولیاء تخصیص می‌دهیم، نام و نام‌خانوادگی وی، همچنین اطلاعات آدرس، شهر، کشور، کدپستی و شماره تلفن وی که برای ما اهمیت دارد. به خاطر بسپارید که مشخصات کودک جزو مشخصهٔ والدین نیست و به صورت مستقل به عنوان یک موجودیت مطرح است.

ـ مشخصهٔ Children عبارت است از : شناسه کودک، نام، سن، جنسیت، تاریخ واکسیناسیون و نیازهای خاص رژیمی و ملاحظات تغذیه‌ای وی.

۳ــ۳ــ ۸ ــ ارتباط بین موجودیت Relationship : برای تعیین ارتباط Relation بین این دو موجودیت، شناسهٔ جدول Parents را در جدول Children اضافه می‌کنیم.

نکته

۱ــ در زمان طراحی بانک، بهتر است شناسه‌های منحصر به فرد را تعیین و یا ایجاد کنیم و از شناسه‌های ترکیبی استفاده نکنیم. مثلاً می‌توانیم از نام و آدرس به صورت مشترک برای کلید استفاده کنیم ولی این کار ایجاد ارتباط بین دو جدول را مشکل می‌سازد.

۲ــ شاید بتوان از شمارهٔ کارت‌ملی و یا شماره حساب سیبا که منحصر به فرد است استفاده کرد.

مشخصه‌ها به فیلدهای جدول تبدیل می‌شوند. فیلدهای دو جدول مزبور در زیر نمایش داده شده‌اند. ارتباط بین آن‌ها نیز به‌وسیلهٔ ParentsId تعریف شده است. (Id نمایانگر عبارت Identifier به معنی شناسه است.)

یادآوری: Table، روش ساخت یافتهٔ نمایش داده‌ها است. فیلدها در Access به صورت عنوان ستون‌ها نمایش داده می‌شوند. جدول، مقادیر داده‌ها را برای نام، نام‌خانوادگی و ... نگهداری می‌کند. این مقادیر به صورت ردیف نمایش می‌یابند که در اصطلاح بانک داده، رکورد Record نامیده می‌شوند.

جدول ۱ــ ۸ــ فیلدهای جدول Parents

ParentsId	FirstName	LastName	Address	City	Country	PostCode	PhoneNumber
شناسه	نام	نام‌خانوادگی	آدرس	نام شهر	نام کشور	کد پستی	شماره تلفن تماس

شرح فیلد :

جدول ۲ــ ۸ ــ فیلدهای جدول Children

ChildId	ParentsId	FName	Age	Sex	VacDate	RunNo	DietNotes
شناسه	شناسه والدین	نام	سن	جنسیت	تاریخ واکسیناسیون	وضعیت‌حضور	ملاحظات تغذیه‌ای

شرح فیلد :

۸ـ۴ ـ استاندارد نام‌گذاری اشیاء بانک

اعـمال یک استاندارد بـرای نـامگـذاری مناسب فـایل‌ها و اشیاء بـانک، کار را قابل اعتماد می‌سازد، خصوصاً زمانی که فایل به اشتراک گذارده می‌شود. مثلاً تمام اشیاء، توصیف‌کننده‌ای داشته باشند که در شروع نام قرار داده شود. استاندارد Access را می‌توانید در آدرس زیر و راهنمای Leszynski/Reddick پیدا کنید. مثلاً این که از کاراکتر فاصله در اسامی استفاده نکنید، ارتباط بین جدول‌ها با استفاده از اسامی مشابه فیلدهای متناظر ایجاد گردد و ...

http: //www.microsoft_accesssolutions.co.uk/naming_conv.htm

۸ـ۵ ـ ایجاد پایگاه داده

فایل پایگاه داده‌ای که در این قسمت ایجاد می‌کنید، در واقع ظرف نگهدارندهٔ (Container) کلیهٔ اشیای بانک خواهد بود مثل : جدول‌ها، فرم‌ها و ...

ـ برنامهٔ Microsoft Access را اجرا کنید.

ـ گزینهٔ New را از منوی اصلی برنامه انتخاب کنید.

ـ در پانل سمت راست نام پایگاه داده را به CaseExample.accdb تغییر دهید، محل ذخیره‌سازی آن را نیز انتخاب نموده، دکمهٔ Create را کلیک کنید.

حال که فایل نگهدارنده ایجاد شد، می‌توان اشیاء (جدول‌ها، فرم‌ها، پرس‌وجوها و گزارش‌ها) را به آن اضافه کرد.

۸ـ۶ ـ ایجاد جدول‌ها

۱ـ۶ـ۸ ـ ایجاد جدول به‌وسیلهٔ Table Templates : چون اشیاء فرم‌ها، گزارش‌ها و پرس‌وجوها بر مبنای جدول‌ها ایجاد می‌شوند. ابتدا باید دو جدول را ایجاد کنیم.

ـ از زبانهٔ Create نوار ابزار، روی TableTemplates کلیک کرده و Contacts را انتخاب نمائید.

ـ جدول ایجاد شده جدید را به‌نام TableParents ذخیره نمایید.

شکل ۱ــ ۸

۸ـ۷ـ اصلاح طراحی جدول‌ها

جدول Table Parents را در نمای Design باز کنید.

تغییرات زیر را روی ساختار فیلدهای جدول اعمال کنید.

ــ فیلد ID را به Parents Id تغییر نام دهید.

ــ جای فیلدهای First Name و Last Name را با هم عوض کنید.

ــ فیلدهای Company، Emaid Address، Job Title، Business Phone، Home Phone و Mobile Phone را حذف نمایید.

ــ فیلد Fax Number را به Phone Number تغییر نام دهید.

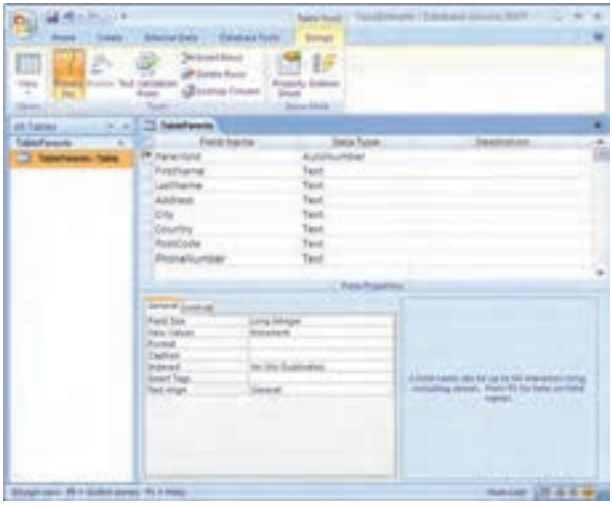
ــ نوع داده‌ای فیلد Address را از Memo به Text تغییر دهید.

ــ فیلد Zip/PostalCode را به PostCode تغییر نام دهید.

ــ فیلد State/Pravince را به Country تغییر نام دهید.

ــ چهار فیلد آخر را از جدول حذف نمایید.

ــ فیلد Phone Number را به انتهای لیست منتقل نمایید.



شکل ۲ــ ۸

در نمای Design، می‌توانید نام، نوع فیلد، توضیحات و دیگر خصوصیات فیلد را تغییر دهید.

توجه شود که ParentsId از نوع داده AutoNumber است، یعنی هر زمان داده‌ای (رکورد هر یک از والدین جدید) اضافه شود، Access به‌طور خودکار، عددی منحصر به فرد به عنوان ParentsId ایجاد می‌کند و بدین‌ترتیب هر رکورد، کلید منحصر به فردی خواهد داشت.

همچنین در Access می‌توان مشخصه Input Mask فیلد را تعریف کرد. این مشخصه کمک می‌کند کاربر داده غلط وارد نکند. مثلاً جا افتادن رقمی از شماره تلفن و ...

ـ کلیک روی PostCode و انتخاب آن.

ـ کلیک در بخش Field Properties و انتخاب ردیف Input Mask. (باید زبانه general انتخاب شده باشد)

ـ روی دکمه ... سمت راست آن کلیک کنید. پیغام ذخیرهٔ جدول را تأیید کنید.

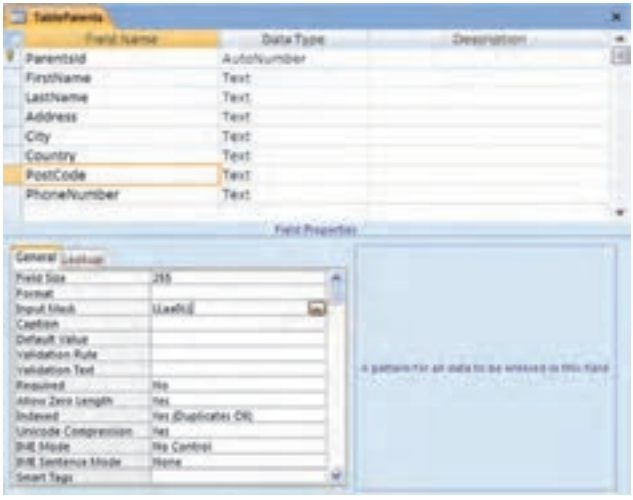
ـ زمانی که Mask Wizard شروع شد، PostCode را انتخاب نمایید.

ـ در قسمت Try It، عبارت LLaoLL را تایپ کنید تا عبارت باید شامل دو حرف سپس دورقم/ حرف و ارقام صفر تا ۹ و سپس دو حرف دیگر باشد (شکل ۳ـ ۸).

ـ روی Finish کلیک کنید.

ـ از منوی اصلی Save را انتخاب کنید.

ـ در صورتی که کار با طراحی جدول تمام شده باشد، آن را Close کنید. ولی ما به حالت Datasheet می‌رویم تا داده‌ها و رکوردهایی را در جدول اضافه کنیم.



شکل ۳ـ ۸

عبارت‌های مجاز در Input Mask

کاراکتر	توضیحات
0	رقم (صفر تا ۹ و علامت [+] یا [−] مجاز نیست).
9	رقم یا خالی (ورود اطلاع ضروری نیست و علامت مثبت یا منفی مجاز نیست).
#	رقم یا فضای خالی (ورود اطلاع ضروری نیست. در زمان ویرایش علامت Space به صورت فضای خالی نمایش داده می‌شود ولی در زمان ذخیره‌سازی حذف می‌گردد. علامت مثبت و منفی مجاز است).
L	حروف (فارسی از الف تا ی و انگلیسی از A تا Z. ورود اطلاع ضروری است).
?	حروف (فارسی از الف تا ی و انگلیسی از A تا Z. ورود اطلاع ضروری نیست).
A	حروف یا رقم (ورود اطلاع ضروری است).
a	حروف یا رقم (ورود اطلاع ضروری نیست).
&	هر کاراکتری و یا فضای خالی (ورود اطلاع ضروری است).
C	هر کاراکتری و یا فضای خالی (ورود اطلاع ضروری نیست).
; : , . / −	جداساز (برای محل رقم هزارگان، تاریخ، زمان و استفاده از کاراکترهای جداساز استاندارد در محیط ویندوز).
<	باعث می‌شود تمام کاراکترها به حروف کوچک تبدیل شوند.
>	باعث می‌شود تمام کاراکترها به حروف بزرگ تبدیل شوند.
!	باعث می‌شود، عبارت از راست به چپ نمایش یابد. کاراکترهایی که در این فیلد تایپ می‌شوند، فیلد را از چپ به راست پر می‌کنند. این علامت در هر جایی از عبارت می‌تواند قرار گیرد.
\	باعث می‌شود کاراکترها به صورت کاراکتر ثابت نمایش داده شوند (مثلاً \A به صورت A نمایش داده می‌شود).

۸ ـ ۸ ـ نمای Datasheet جدول

پس از تغییرات طراحی جدول، نمای Design را ترک کرده، به نمای Datasheet می‌رویم تا بتوانیم در این نما، داده‌هایی را به جدول اضافه کنیم (شکل ۴ــ ۸).

شکل ۴ــ ۸

ورود داده به جدول : مطابق شکل ۵ ــ ۸، داده‌ها را به جدول اضافه کنید. با کمک کلیدهای TAB روی فیلدها و از رکوردی به دیگری جابه‌جا شوید و یا بالعکس، از Shift+TAB استفاده کنید. پس از اتمام ورود داده‌ها، پنجره مربوطه را ببندید.

شکل ۵ــ ۸

۹ ـ ۸ ـ استفاده از فرم‌ها

۱ــ۹ــ ۸ ــ ایجاد فرم جدید ورود اطلاعات : با کمک فرم می‌توان اطلاعات وارد شده را کنترل کرد و از ثبت اطلاعات غلط جلوگیری کرد. مثلاً زمان ورود اطلاعات کودکان مهدکودک، می‌توانیم از جدول والدین استفاده و لیستی از آن‌ها ارایه کنیم. بدین‌ترتیب شناسهٔ والدین را به‌طور صحیح به‌دست آورده و در جدول Children به همراه سایر اطلاعات کودک ذخیره نماییم. ابتدا فرمی بدون هیچ‌گونه کنترلی ایجاد می‌نماییم.

۲ــ۹ــ ۸ ــ ایجاد فرم :

برای ایجاد فرم ابتدا جدول TableParents را انتخاب کنید سپس از زبانه Create نوار ابزار گزینهٔ Form را انتخاب نمایید.

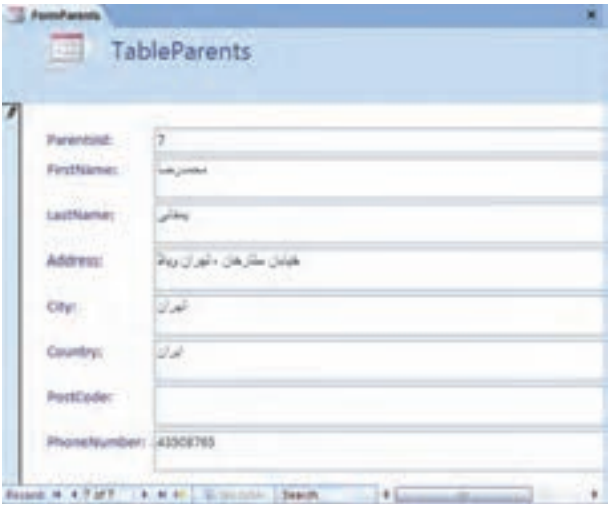
فرم جدید ایجاد شده را به نام FormParents ذخیره نمایید.

۳ــ۹ــ ۸ ــ ورود اطلاعات به‌وسیلهٔ فرم : فرم FormParents را در نمای FormView باز کنید، برای درج اطلاعات جدید می‌توانید از نوار موجود در قسمت پایین فرم استفاده نمایید و یا با فشردن کلیدهای ترکیبی (+) + Ctrl+Shift در وضعیت ثبت اطلاعات جدید قرار بگیرید.

ــ اطلاعات نام و آدرس بیشتری را مطابق شکل ۶ ــ ۸ وارد نمایید.

شکل ۶ــ ۸

برای این کار می‌توانید از فرم شکل ۷ــ ۸ نیز استفاده کنید.



شکل ۷ــ ۸

ــ با دکمهٔ Close فرم را ببندید.

ــ جدول TableParents را باز کنید و بررسی کنید آیا اطلاعات اضافه شده است (شکل ۸ ــ ۸).

شکل ۸ــ ۸

یادآوری : به خاطر داشته باشید که فرم، یکی از راه‌های ورود داده به جدول است و تمام اطلاعات در جدول ذخیره می‌شود. زمانی که داده‌ای در فرم تایپ می‌شود، اطلاعات آن در جدول ذخیره می‌شود و نه در فرم مربوطه.

۱۰ـ ۸ـ روابط Relationship

انواع رابطه‌ها : سه نوع اصلی ارتباط وجود دارد :

● ***یک ــ به ــ یک (One-to-One)*** : یک رکورد در یک جدول با فقط یک رکورد از جدول دیگر مرتبط است.

● ***یک ــ به ــ چند (One-to-Many)*** : یک رکورد در یک جدول می‌تواند به رکوردهای زیادی از جدول دیگر مرتبط باشد.

● ***چند ــ به ــ چند (Many-to-Many)*** : یک رکورد در یک جدول می‌تواند به یک یا چند رکورد جدول دیگر مرتبط باشد و یک رکورد آن جدول می‌تواند به یک یا چند رکورد جدول اولی مرتبط گردد.

ارتباط یک ــ به ــ یک با استفاده از یک جدول به جای دو جدول حل می‌شود. ارتباط چند ــ به ــ چند معمولاً به جدول سومی نیاز دارد که به عنوان رابط بین دو موجودیت به کار رود. برای توضیحات بیشتر در این خصوص به کتاب‌های پایگاه داده و بخش مفاهیم اولیه مراجعه کنید.

۱۱ـ ۸ـ افزودن جدول دیگر

اکنون به جدول دیگری نیاز داریم که موجودیت Children را ایجاد کند. چون جدول‌های نمونه Table Templates مناسب به نظر نمی‌رسند، باید خودمان جدول را ایجاد و طراحی کنیم. باید نام فیلد، نوع فیلد و سایر خصوصیات را تعریف کنیم.

نام فیلدهای جدول عبارت‌اند از :

ChildId, ParentsId, Name, Age, Sex, VacDate , RunNo و DietNotes

از توصیف فارسی نام فیلدها به‌عنوان (Caption) فیلد استفاده می‌کنیم. در این‌جا انواع داده و خصوصیات فیلدها را بررسی می‌کنیم :

ــ ChildId از نوع AutoNumber است. چرا که Access خود به‌طور اتوماتیک به هر رکورد

جدید مقدار منحصر به فردی را تخصیص می‌دهد. این فیلد به عنوان کلید اصلی (Primary key) شناخته می‌شود.

ـ ParentsId از نوع دادهٔ Number است و اندازهٔ فیلد (Field size) آن Long Integer است. این موضوع بسیار مهم است چرا که باید بتوانیم Children و Parents را به هم مرتبط سازیم. ParentsId در جدول Parents، والدین را مشخص می‌کند و از نوع AutoNumber است. اگر بخواهیم این دو جدول را به هم مرتبط کنیم باید فیلد متناظر در جدول دیگر از نوع Long Integer باشد. در این حالت ParentsId در جدول Children، به عنوان کلید خارجی (Foreign key) شناخته می‌شود و کلید اصلی جدول Parents است.

ـ Age از نوع Number است. می‌توانیم اندازه فضایی که برای این فیلد ذخیره می‌شود را کاهش دهیم، بدین صورت که اندازهٔ فیلد را Byte قرار دهیم. این نوع، امکان ذخیرهٔ تا مقدار ۲۵۵ را فراهم می‌کند و بیشتر کودکان بیش از ۱۰۰ سال زندگی نمی‌کنند و حد مقدار حتی بیشتر از کفایت است.

ـ Sex از نوع Text است. در قواعد کنترلی (validation rule) مقدار In ("M" ; "F") قرار می‌دهیم. این قاعده به معنی آن است که مقدار وارد شده در این فیلد باید M یا F باشد. چون این مقدار یک حرف است، باید اندازهٔ فیلد ۱ قرار داده شود. (مقدار پیش‌فرض اندازهٔ فیلد text ، ۵۰ بوده است).

ـ VacDate از نوع Date/Time در نظر می‌گیریم. (به خاطر داشته باشید که این نوع فقط مقادیر تاریخ میلادی را ذخیره می‌کند و می‌توان محاسبات ریاضی روی آن اعمال کرد. درصورتی که‌بخواهید تاریخ شمسی را ذخیره نمایید باید از نوع text و اندازهٔ ۱۰ استفاده کنید ولی دیگر قادر به استفاده از توابع تاریخ و عملگرهای ریاضی روی آن نخواهید بود).

ـ RunNo عددی مثبت با اندازهٔ فیلد Byte است. Access به‌طور اتوماتیک مقدار پیش‌فرض (Default Value) صفر را برای آن لحاظ می‌کند. ما از این خصوصیت در زمان ایجاد Query استفاده خواهیم کرد.

ـ DietNotes از نوع فیلد Memo است. چون نوع text می‌تواند حداکثر تا ۲۵۵ کاراکتر را دربر گیرد. این مقدار برای توضیحات ما کافی نیست بنابراین از نوع Memo استفاده می‌کنیم و کاربرد آن را نشان دهیم.