

آشنایی با

Activex Data Objects (ADO)

هدف‌های رفتاری: پس از آموزش این فصل هنرجو می‌تواند:

- دلیل استفاده از ADO به عنوان متد دسترسی به داده‌ها را شرح دهد.
- هدف از SQL را شرح دهد.
- شکل کلی عبارات SQL ساده را تعریف کند (عبارت Select).
- رکوردها را در یک پرس‌وجو با استفاده از عبارات ساده SQL فیلتر کند.
- چگونگی اتصال کنترل ADO Data به منبع داده‌ها را شرح دهد.
- کنترل ADO Data را به جعبه ابزار ویژوال بیسیک اضافه کند.
- کنترل ADO Data را در یک برنامه کاربردی به کار گیرد.
- چگونگی ارتباط شیء Recordset به کنترل ADO Data را شرح دهد.
- رکوردهای جدیدی را به Recordset اضافه کند.
- رکوردهای موجود را به هنگام کند.
- رکوردها را از Recordset حذف کند.
- رکوردهای درون Recordset را جست‌وجو کند.

تقریباً تمام برنامه‌های کاربردی نیاز به دسترسی به داده‌ها دارند. برای برنامه‌های کاربردی، که در یک کامپیوتر اجرا می‌شوند، دسترسی به داده‌ها و پیاده‌سازی آن ساده است و نیاز به برنامه‌نویسی کمی دارد. مکانیزم دسترسی به داده‌ها تحت شبکه متفاوت است.

به عنوان یک برنامه‌نویس، مجبور خواهید بود که فناوری دسترسی به داده‌هایی که برای ایجاد برنامه به کار می‌برید را انتخاب کنید. فناوری‌های دسترسی به داده‌هایی که در ویژوال بیسیک ارایه شده‌اند، زمان ایجاد برنامه را کاهش داده و کد نویسی را ساده می‌کنند و کارایی بالایی را ارایه می‌دهند.

۱-۴- رابط‌های دسترسی به داده‌ها

یک رابط دسترسی به داده‌ها یک مدل شیء است که شکل‌های مختلف دسترسی به داده‌ها را ارایه می‌کند. در ویژوال بیسیک، سه نوع رابط دسترسی به داده‌ها وجود دارد که عبارتند از: Remote Data Objects (RDO)، Activex Data (ADO) و Data Access Objects (DAO). در ویژوال بیسیک می‌توان با برنامه‌نویسی اتصال به بانک اطلاعاتی، بازیابی رکوردها و تغییر مقدار رکوردها را کنترل کرد.

از هر کدام از سه فناوری دسترسی به داده‌ها برای تعامل با بانک اطلاعاتی می‌توان استفاده کرد، ولی ADO جدیدترین و قوی‌ترین آن‌ها و رابطی برای OLE DB است. OLE DB راهکار رابط سطح پایین برای تمام انواع داده‌هاست. این مفهوم را UDA^۱ نیز می‌نامند.

به عنوان مثال، OLE DB و ADO رابط یکسانی را نه تنها برای دسترسی به داده‌های بانک‌های اطلاعاتی رابطه‌ای و غیررابطه‌ای فراهم می‌کنند، بلکه به سایر منابع مثل پست الکترونیکی، سیستم‌های فایل، ابزارهای مدیریت پروژه، صفحه گسترده‌ها و شیء‌های کاری خاص را نیز ارایه می‌کنند.

در کتاب بانک اطلاعاتی با مفاهیم بانک‌های اطلاعاتی رابطه‌ای و عبارت‌های SQL آشنا شده‌اید.

تمرین: مفاهیم زیر را توضیح دهید:



جدول - رکورد - فیلد - کلید

تمرین: ۱ - هدف زبان SQL را بیان کنید.



۲- عبارت SELECT و اجزای آن را شرح دهید.

۴-۲- کنترل ADO Data

اگر چه می توان از ADO به طور مستقیم در برنامه های کاربردی استفاده کرد، ولی ADO Data دارای مزیت کنترل گرافیکی به همراه دکمه های پیمایش رکوردی است. همچنین این رابط ساده، امکان می دهد تا برنامه های کاربردی بانک اطلاعاتی را با حداقل کدنویسی ایجاد کنید.

کنترل ADO Data از Microsoft ADO برای ایجاد اتصال سریع بین کنترل های مقید به داده ها (data-bound) و ارایه کننده داده استفاده می کند. کنترل های Data-bound (مقید به داده ها)، کنترل هایی هستند که دارای مشخصه DataSource می باشند و شامل Image, ComboBox, CheckBox و PictureBox, Label, ListBox, و TextBox است. این کنترل ها بانکی از فیلدهای جداول بانک اطلاعاتی متصل شده و مقادیر آن فیلد را نمایش می دهند.

به علاوه ویژوال بیسیک دارای چندین کنترل Activex^۱ مقید به داده ها مثل DataGrid, DataCombo و Datalist نیز است. هنگامی که از کنترل ADO Data استفاده می کنید، می توانید هر فیلدی را به یک کنترل مقید کنید و هنگام پیمایش رکوردها، به طور خودکار محتوای فیلدها نمایش داده می شوند. این کار به طور داخلی به وسیله ویژوال بیسیک انجام می شود و نوشتن هیچ کدی نیاز نیست.

۴-۳- ایجاد فرم های مقید به بانک اطلاعاتی با Data Form Wizard

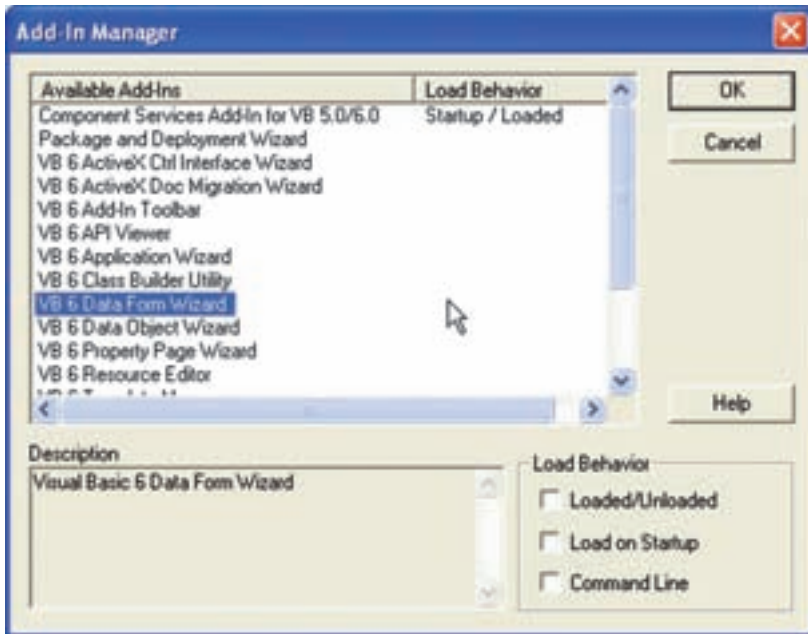
ویژوال بیسیک ابزاری ارایه می کند که امکان ایجاد فرم هایی که دارای کنترل های مقید به بانک اطلاعاتی هستند را فراهم می کند. می توان این ویزارد را از منوی Add-Ins فعال کرد. ویزارد Data Form فرمی را ایجاد می کند که امکان مرور بانک اطلاعاتی، کامل کردن کادرهای متن، برچسب ها و کنترل ADO Data را فراهم می کند.

هنگامی که ویژوال بیسیک را نصب می کنید، ویزارد Data Form و ویژوال بیسیک نصب نمی شود و باید آن را به منوی Add-Ins اضافه کنید.

۱- کنترل Activex یک مؤلفه نرم افزاری است که می تواند در صفحات وب، آفیس یا هر میزبانی که از کنترل های Activex پشتیبانی می کند، قرار داده شود و دارای متدهایی است که می تواند از طریق سایر برنامه های کاربردی، کتابخانه های پیوند پویا (ppl)، صفحات وب یا کنترل های دیگر فراخوانی شود.

۱-۳-۴- نصب Data Form Wizard :

- ۱- از منوی Add-Ins گزینه Add-In Manager را انتخاب کنید.
- ۲- در کادر محاوره‌ای Add-In Manager، گزینه Data Form Wizard، VB6 را از لیست انتخاب کرده و کادر علامت Loaded/Unloaded را فعال کنید (شکل ۴-۱) روی OK کلیک کنید.



شکل ۴-۱- اگر گزینه Load on Startup را انتخاب کنید، ویزارد Data Form بعد از عملیات نصب به منوی Add-Ins اضافه می‌شود.

اکنون که ویزارد را به IDE ویزوال بیسیک اضافه کردید، می‌توانید آن را در تمام پروژه‌ها به کار

ببرید.

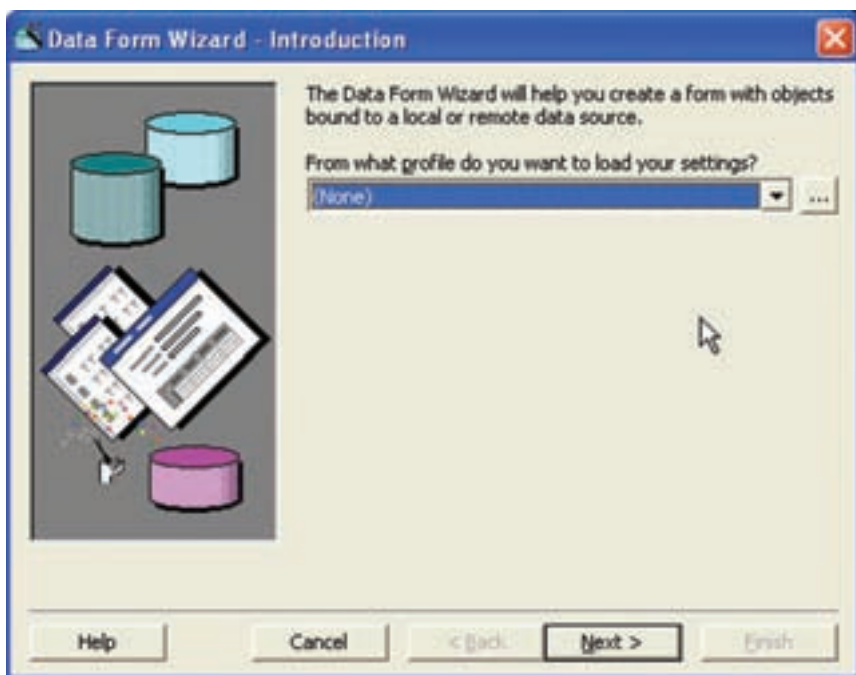


ایجاد فرم مقید به جدول بانک اطلاعاتی: از ویزارد VB Data Form به منظور ایجاد فرم

برای جدول tblWorks از بانک اطلاعاتی composer.mdb استفاده می‌کنیم.

۱- از منوی Add-Ins گزینه DataForm Wizard را انتخاب کنید.

۲- در کادر محاوره‌ای Introduction روی Next کلیک کنید (شکل ۴-۲).



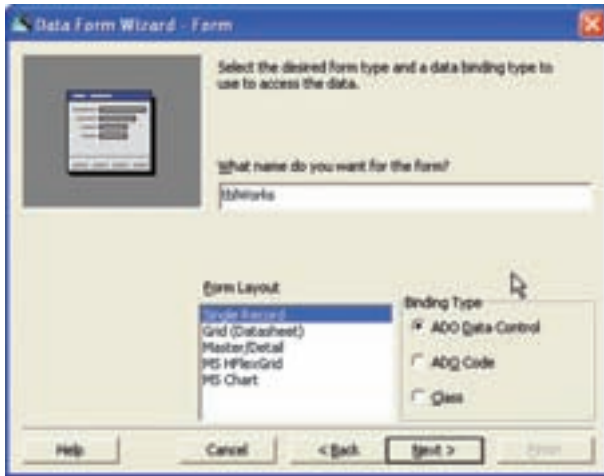
شکل ۲-۴- کادر محاوره‌ای Introduction امکان بارگذاری پرو فایل تنظیمات Data Form Wizard را فراهم می‌کند.

۳- در کادر محاوره‌ای Database Type، انتخاب Access را برگزیده و روی Next کلیک کنید.

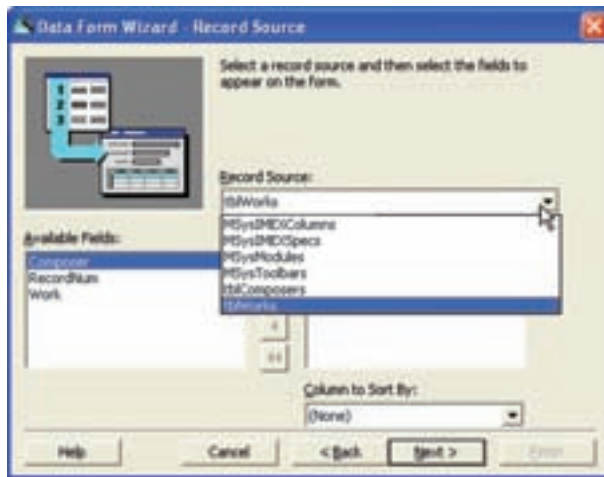
۴- در کادر محاوره‌ای Database روی دکمه Browse کلیک کنید تا کادر محاوره‌ای Access Database را مشاهده کنید. اکنون می‌توانید بانک اطلاعاتی مورد نظر برای ایجاد فرم را انتخاب کنید. Composer.mdb را پیدا کرده و روی Next کلیک کنید.

تحقیق: اگر فایل بانک اطلاعاتی در Access XP, ۲۰۰۳ ایجاد شده باشد، خطا رخ می‌دهد. برای رفع آن چه کاری باید انجام داد؟

۵- نام فرم را frmWorks قرار داده و از لیست Form Layout گزینه Single Record را انتخاب کنید. از گزینه‌های Binding Type گزینه ADO Data Control را انتخاب کرده و روی Next کلیک کنید (شکل ۳-۴).



شکل ۳-۴ طرح کلی فرم را با انتخاب گزینه‌ای از لیست Form Layout انتخاب کنید.



شکل ۴-۴ تمام جدول‌ها و پرس و جوهای بانک اطلاعاتی انتخاب شده در لیست بازشوی Record Source فهرست می‌شوند. بعد از انتخاب Record Source، فیلدهای آن در لیست Available fields ظاهر خواهند شد.

۶- جدول tblworks را از لیست بازشوی Record Source انتخاب کنید. لیست بازشوی

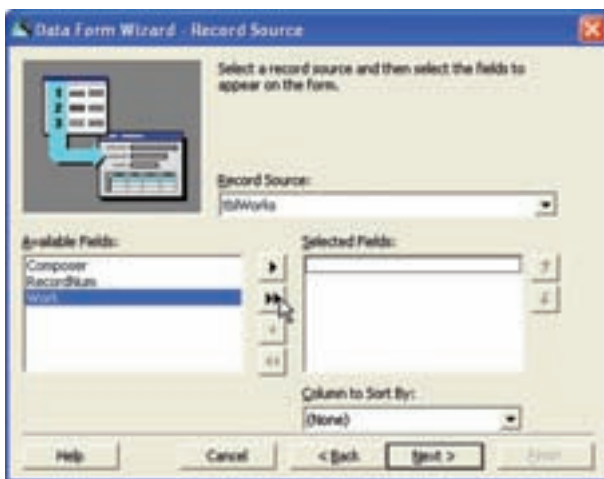
Record Source شامل تمام جداول و پرس‌وجوهای بانک اطلاعاتی انتخاب شده است.

۷- تمام فیلدهای جدول انتخاب شده از لیست Record Source در لیست Available Fields

ظاهر می‌شود از دکمه‌های «» و «» می‌توان به ترتیب برای انتخاب برخی از فیلدها یا همه فیلدها استفاده

کرد. روی دکمه «» و سپس Next کلیک کنید تا تمام فیلدها انتخاب شود. (شکل ۵-۴) در صورتی که

می خواهید رکوردها براساس فیلد خاصی مرتب شوند، از کادر Column to sort by نام فیلد مورد نظر را انتخاب کنید. روی Next کلیک کنید.



شکل ۴-۵ هر فیلدی که انتخاب کنید روی فرم داده‌ها ظاهر خواهد شد. در صورتی که بعضی از فیلدها را می‌خواهید، آن‌ها را انتخاب کرده و روی دکمه ▶ کلیک کنید. برای حذف فیلدها از فرم داده‌ها، روی دکمه‌های ◀ و ◀◀ کلیک کنید.

۸- در کادر محاوره‌ای Control Selection کادرهای علامت را از حالت انتخاب خارج نکنید، بنابراین تمام کنترل‌های کار کردن با داده‌ها روی فرم قابل دسترس خواهند بود (شکل ۴-۶) روی Next کلیک کنید.



شکل ۴-۶ می‌توان تعداد محدودی از دکمه‌ها را با انتخاب یا پاک کردن کادر علامت مربوطه روی فرم قرار داد.

۹- در کادر محاوره‌ای پایانی تنظیمات ایجاد شده را با کلیک کردن روی دکمه سه نقطه، ذخیره کنید (شکل ۷-۴). این تنظیمات در فایل‌ی با پسوند .twp ذخیره می‌شوند.

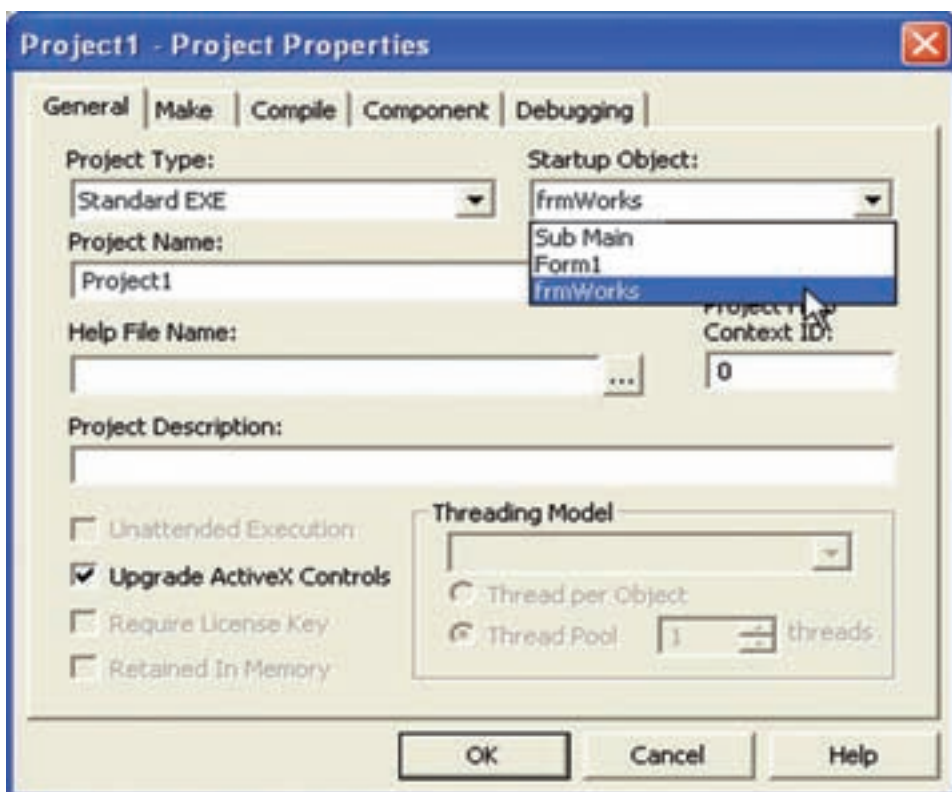


شکل ۷-۴- ذخیره تنظیمات در یک پروفایل می‌تواند سبب صرفه‌جویی در وقت برنامه بانک اطلاعاتی شود.

۱۰- روی Finish کلیک کنید. کادر محاوره‌ای نهایی Data Form Created خواهد شد. در صورتی که نمی‌خواهید این پیام تأیید دفعه بعد ظاهر شود، روی کادر علامت Don't show this Dialog in the Future کلیک کنید. روی OK کلیک کنید.

۱۱- کادر محاوره‌ای Project Properties را با انتخاب Project1 Properties از منوی Project باز کنید.

۱۲- فرم frmWorks را از لیست بازسوی Startup Object انتخاب کنید (شکل ۸-۴).



شکل ۸-۴ در صورتی که می‌خواهید فرم جدید را به صورت فرم آغازین در نظر بگیرید، باید شیء Startup را دوباره مقاردهی کنید.

۱۳- پروژه را ذخیره کرده و اجرا کنید (شکل ۹-۴).



شکل ۹-۴ فرم داده‌های ایجاد شده با ویزارد، امکان اضافه، به‌هنگام، حذف و نو کردن (refresh) داده‌های جدول را فراهم می‌کند.

کنترل دیگری به نام کنترل Data در جعبه ابزار ویژوال بیسیک وجود دارد که با کنترل ADO Data متفاوت است. برای کار با کنترل Data باید مشخصه Data BaseName آن را مساوی بانک اطلاعاتی موردنظر قرار دارد و در مشخصه Record Source آن یکی از جداول بانک اطلاعاتی را انتخاب نمود.

می‌توان از چندین کنترل Data روی فرم استفاده کرد. هر کنترل Data می‌تواند RecordSource متفاوتی از یک بانک اطلاعاتی یا از بانک‌های اطلاعاتی مختلف داشته باشد. همچنین می‌توان مشخصه‌های کنترل Data را در زمان اجرا تغییر داد.

کنترل Data امکان کار کردن با بانک اطلاعاتی ساده را فراهم می‌کند ولی دارای محدودیت است. برای برنامه‌نویسی حرفه‌ای با بانک اطلاعاتی از فناوری پیشرفته‌ای به نام ActiveX Data Objects (ADO) استفاده کنید. کنترل Data برای برنامه‌های کاربردی ساده‌تر، مناسب است.

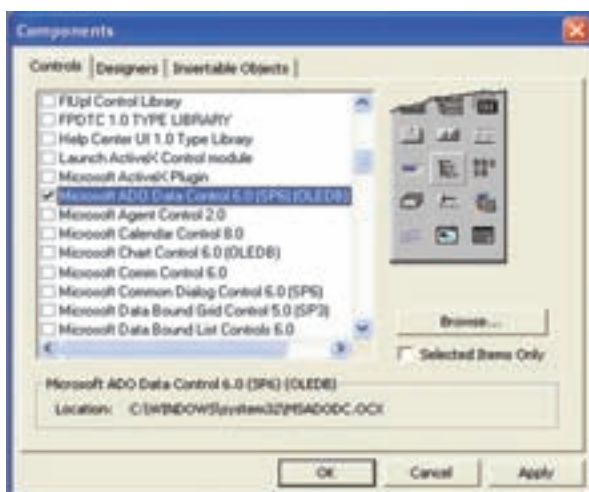
۴-۴- اضافه کردن کنترل ADO Data به جعبه ابزار

برای اضافه کردن کنترل ADO Data به جعبه ابزار، مراحل زیر را انجام دهید :

۱- از منوی Project گزینهٔ Components را انتخاب کنید.

۲- در کادر محاوره‌ای Components، روی Microsoft ADO Data Control 6.0 (OLE DB) کلیک کنید (شکل ۱-۴).

۳- روی OK کلیک کنید.



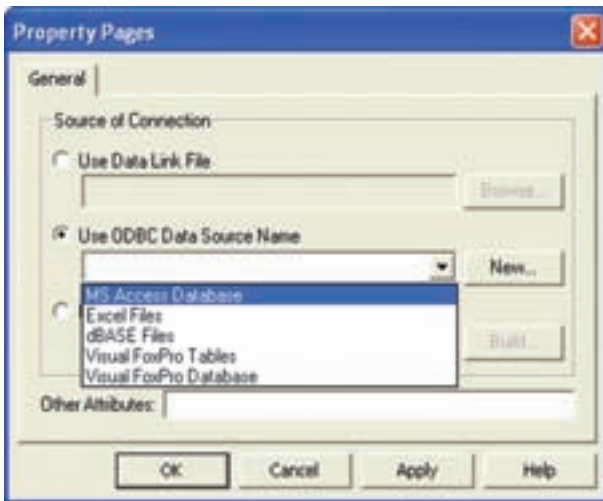
شکل ۱-۴- کادر محاوره‌ای Components

۴-۵- اتصال به منبع داده‌ها

در زمان طراحی، می‌توان با مقدار دهی مشخصه `ConnectionString` کنترل `ADO Data` با یک رشته اتصال معتبر، اتصال به منبع داده‌ها را برقرار کرد. برای مقداردهی این مشخصه روی شیء راست کلیک کرده و گزینه `ADODC Propertic` را انتخاب کنید (یا در پنجره `Properties` روی علامت سه نقطه روی مشخصه `ConnectionString` کلیک کنید) کادر محاوره‌ای `property Page` باز می‌شود (شکل ۴-۱۱). هنگامی که مشخصه `ConnectionString` از کنترل `ADO Data` را می‌خواهید مقداردهی کنید، سه انتخاب خواهید داشت :

● **Use Data Link File**: این گزینه تعیین می‌کند که شما می‌خواهید از یک فایل با پسوند `UDL` خاص استفاده کنید. هنگامی که این گزینه انتخاب شود، می‌توانید روی `Browse` کلیک کنید تا به کادر محاوره‌ای `Select Data Link File` دسترسی پیدا کنید و از آنجا پرونده `Data Link` را انتخاب نمایید.

● **Use ODBC Data Source Name**: این گزینه تعیین می‌کند که شما می‌خواهید از نام منبع داده‌های تعریف شده (`DSN`) برای رشته اتصال استفاده کنید. می‌توان به فهرستی از `DSN`های تعریف شده سیستم از طریق کادر ترکیبی دسترسی پیدا کرد و `DSN` موردنظر را انتخاب نمود و یا (شکل ۴-۱۱). روی `New` کلیک کرد و از طریق کادر محاوره‌ای `Create New Data Source Wizard` برای ایجاد `DSN` جدید و یا اصلاح `DSN`های موجود، استفاده کرد.



شکل ۴-۱۱- به‌کارگیری
گزینه نام منبع داده‌های ODBC

● Use Connection String: این گزینه تعیین می‌کند که شما می‌خواهید از یک رشته

اتصال برای دسترسی به داده‌ها استفاده کنید. روی Build کلیک کنید تا به کادر محاوره‌ای Data Link Properties دسترسی پیدا کنید. با استفاده از این کادر محاوره‌ای می‌توان اتصال، مجوزهای دسترسی و اطلاعات اضافی مورد نیاز برای دسترسی به داده‌ها را با استفاده از آرایه‌کننده OLE DB تعیین کرد.

۴-۶- تعیین رشته اتصال

در روال زیر، ما روی استفاده از رشته اتصال به منبع داده‌ها تمرکز می‌کنیم. در این فرآیند، یک آرایه‌کننده OLE DB را انتخاب، نام و محل بانک اطلاعاتی را تعیین و اتصال را آزمایش خواهید کرد.

مقداردهی مشخصه Connection String را به صورت زیر انجام دهید:

۱- یک کنترل ADO Data روی فرم قرار دهید.

۲- در پنجره Properties مربوط به این کنترل، مقابل مشخصه Connection String

روی ... کلیک کنید.

۳- گزینه Use Connection String را انتخاب و روی Build کلیک کنید.

۴- در کادر محاوره‌ای Data Link Properties و زبانه Provider به دلیل این که بانک

اطلاعاتی که ایجاد کرده‌ایم، در Access بوده است، گزینه Microsoft jet 4.0 OLE DB Provider را انتخاب می‌کنیم.

۵- در زبانه Connection با کلیک روی ... شماره ۱ نام بانک اطلاعاتی مورد نظر (در این

مثال، Nwind.mdb) را انتخاب کنید. در شماره ۲ این کادر محاوره‌ای می‌توانید نام کاربری و گذر واژه خاصی را برای دسترسی به بانک اطلاعاتی تعیین کنید.

۶- روی Test Connection کلیک کنید تا مطمئن شوید که اتصال برقرار شده است. یک

کادر پیام ظاهر می‌شود و اعلان می‌کند که اتصال موفقیت‌آمیز بوده است یا نه؟

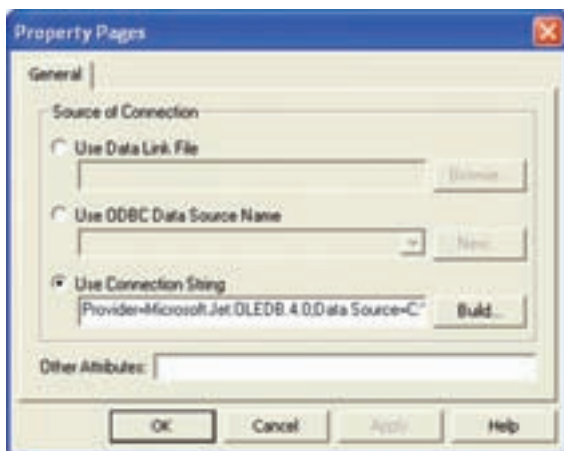
۷- در صورتی که پیام موفقیت‌آمیز بودن ایجاد اتصال را دریافت کردید، روی OK کلیک کنید

و در کادر محاوره‌ای Data Link Properties نیز روی OK کلیک کنید.

یک مقدار رشته‌ای به‌طور خودکار برای Connection String تولید خواهد شد

(شکل ۱۲-۴).

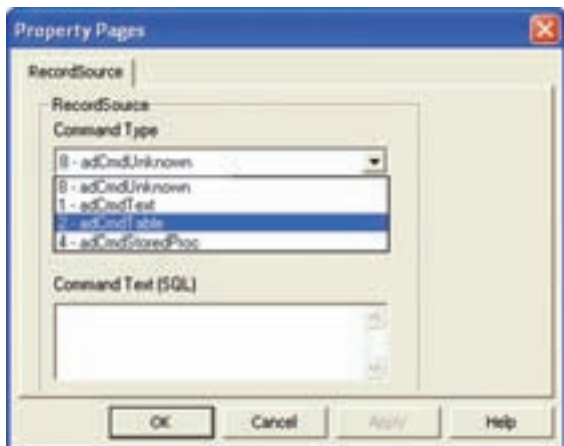
۸- روی OK کلیک کنید تا کادر محاوره‌ای Property Pages بسته شود.



شکل ۱۲-۴ یک مقدار رشته‌ای اتصال

۱-۶-۴- تعیین مشخصه RecordSource: بعد از مقداردهی مشخصه ConnectionString

برای اتصال به بانک اطلاعاتی، می‌توانید مشخصه RecordSource را برای به دست آوردن رکوردها، مقداردهی کنید. مشخصه RecordSource می‌تواند با نام جدول، پرس‌وجوی ذخیره شده یا یک عبارت SQL مقداردهی شود. برای بهبود کارایی، از مقداردهی این مشخصه با یک جدول کامل پرهیز کنید. مقدار این مشخصه را برابر با یک رشته SQL قرار دهید تا فقط رکوردهای مورد نیاز را بازیابی کند. با توجه به اینکه شکل دستورهای SQL در برنامه‌های Microsoft Access و Microsoft SQL Server متفاوت است. بنابراین باید برای هر بانک اطلاعاتی خاص، از شکل دستور مناسبی استفاده کرد. مشخصه Record Source می‌تواند در زمان طراحی و با استفاده از Property Pages مقداردهی شود (شکل ۱۳-۴).



شکل ۱۳-۴ صفحه مشخصه

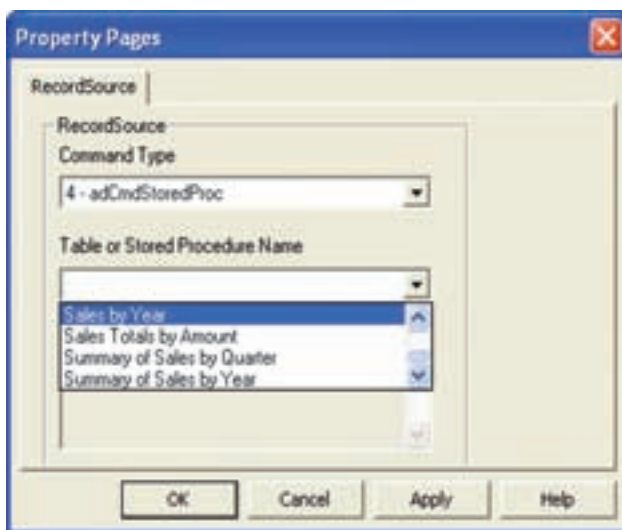
Record Source

در کادر محاوره‌ای فوق، پارامتر نوع فرمان را تعیین کنید که به ADO بیان می‌کند که نوع شیء فرمان مورد استفاده کدام است. جدول ۴-۱، گزینه‌های نوع فرمان مختلف را شرح می‌دهد.

جدول ۴-۱- انواع فرمان ADO

شرح	مقدار
نوع فرمان در مشخصه Command Text ناشناخته است. مقدار پیش فرض است.	adCmdUnknown
Command Text را به عنوان یک تعریف متنی از یک فرمان یا فراخوانی روال ذخیره شده در نظر می‌گیرد.	adCmd Text
Command Text را به عنوان نام یک جدول در نظر می‌گیرد که تمام ستون‌های آن به وسیله یک پرس‌وجوی SQL که به طور داخلی تولید می‌شود، برگردانده شده است.	adCmd Table
متن فرمان را بنام یک روال ذخیره شده در نظر می‌گیرد. این می‌تواند یک روال ذخیره شده در بانک اطلاعاتی SQL Server یا یک پرس‌وجو در Access باشد.	adCmd StoredProc

اگر adCmd Table یا adCmd StoredProc را انتخاب کنید، از کادر لیست زیرین، می‌توانید نام جدول یا روال ذخیره شده را انتخاب کنید (شکل ۴-۱۴).



شکل ۴-۱۴- انتخاب یک روال ذخیره شده به عنوان RecordSource

اتصال کنترل ADO Data به منبع داده ها: در این تمرین، کنترل ADO Data را به فرم اضافه خواهید کرد و سپس آن را به منبع داده ها متصل می کنید. منبع داده ها، بانک اطلاعاتی نمونه Northwind (Nwind.mdb) است که درون ویژوال بیسیک قرار دارد. با چگونگی اضافه کردن کنترل ADO Data به جعبه ابزار قبلاً آشنا شدید.

۱- کنترل ADO Data را به Form1 اضافه کنید.

۲- روی کنترل ADO Data کلیک راست و ADODC Properties را انتخاب کنید.

کادر محاوره ای Property Pages ظاهر می شود. در زبانه General، مطمئن شوید که گزینه Connection String انتخاب شده باشد.

۳- برای تعیین رشته اتصال، روی Build کلیک کنید.

۴- Microsoft jet OLE DB Provider را انتخاب و روی Next کلیک کنید.

به دلیل این که بانک اطلاعاتی از نوع Access است این گزینه انتخاب شده است.

۵- از زبانه Connection برای انتخاب یا تایپ نام بانک اطلاعاتی استفاده کنید. روی ... کلیک کنید.

۶- در کادر محاوره ای SelectAccessDatabase، روی Nwind.mdb و سپس open کلیک کنید.

۷- در کادر محاوره ای Data Link Properties روی Test Connection کلیک کنید.

یک کادر پیام ظاهر شده و بیان می کند که اتصال موفقیت آمیز بوده است یا نه؟

۸- روی OK کلیک کنید.

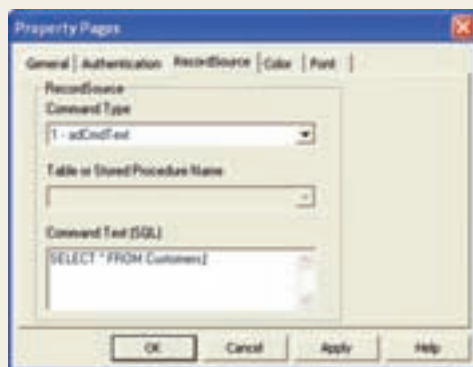
۹- تا این جا، مقدار رشته اتصال ایجاد شده است، روی زبانه RecordSource از کادر محاوره ای Property Page کلیک کنید.

۱۰- برای Command Type گزینه adCmd Text را انتخاب کنید.

۱۱- برای مقدار Command Text (SQL)، عبارت SQL زیر را تایپ کنید (شکل ۱۵-۴).

SELECT * FROM Customers

- ۱۲- روی OK کلیک کنید تا کادر محاوره‌ای Property Pages بسته شود.
 اکنون کنترل ADO Data مقید به منبع داده‌هاست و می‌توانید اطلاعات را از جدول Customers در بانک اطلاعاتی Nwind.mdb بازیابی کنید.
- ۱۳- از منوی File گزینه Save Project را انتخاب کنید و پروژه را با نام PrjADO.vbp ذخیره کنید تا در تمرین بعدی نیز از آن استفاده کنید.



شکل ۱۵-۴- کادر محاوره‌ای
 Property Pages

۴-۷- مقیدسازی کنترل‌ها

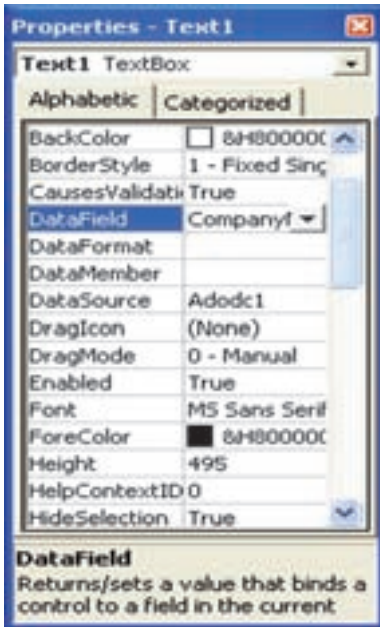
بعد از تعیین مشخصه‌های ConnectionString و RecordSource برای کنترل ADO Data، که به ترتیب بانک اطلاعاتی و یکی از جداول یا Queryهای بانک اطلاعاتی را مشخص می‌کند، می‌توانید یک کنترل مقید برای نمایش داده‌ها روی فرم اضافه کنید.

هنگامی که با کنترل ADO Data بین رکوردها جابه‌جا شوید، کنترل‌های مقید نیز داده‌های فیلدها را در رکوردها نمایش می‌دهند. با تغییر مقدار یک فیلد و جابه‌جایی بین رکوردها، این تغییر به طور خودکار به بانک اطلاعاتی اعمال می‌شود. مزیت استفاده از کنترل‌های مقید، به حداقل رساندن مقدار کدنویسی است. به دلیل این که مقدار کنترل مقید به طور خودکار از بانک اطلاعاتی بازیابی و در آن نوشته می‌شود، به برنامه‌نویسی کمی نیاز دارد.

۴-۷-۱- مقداردهی مشخصه‌های DataSource و DataField: برای این که کنترلی را به

کنترل ADO Data مقید کنید، باید این دو مشخصه را تنظیم نمایید. مشخصه DataSource منبع داده را از طریق کنترلی که به بانک اطلاعاتی مقید است، تعیین می‌کند (در این مثال، کنترل ADO Data). شیء RecordSet شامل مجموعه‌ای از رکوردهای جدول یا Query است که به وسیله مشخصه

RecordSource تعیین شده است. برای نمایش مقادیر فیلدهای این رکوردها در کنترل‌های مقید از مشخصه DataField کنترل‌های مقید استفاده می‌کنیم.



مشخصه DataField نام فیلد معتبری که در شیء Recordset به وسیله منبع داده‌ها ایجاد شده است را تعیین می‌کند. این مقدار تعیین می‌کند که کدام فیلد در کنترل مقید، نمایش داده شود. این دو مشخصه را می‌توان در زمان طراحی و از طریق پنجره Properties مقداردهی کرد (شکل ۱۶-۴).

هم چنین می‌توان این دو مشخصه را در زمان اجرا نیز مقداردهی کرد. اگر می‌خواهید مشخصه DataSource را در زمان اجرا با استفاده از کد مقداردهی کنید، باید از کلید ویژه set استفاده کنید، زیرا این مشخصه یک شیء است. مثال زیر، این دو مشخصه را برای کنترل Text box مقداردهی می‌کند.

شکل ۱۶-۴ - تنظیم مشخصه‌های DataSource و Datafield

```
set txt1.DataSource = Adodc1
txt1.DataField = "CompanyName"
```

تمرین:

مقید سازی کنترل‌ها

در این تمرین از پروژه ایجاد شده در تمرین قبلی استفاده خواهیم کرد:

۱- پروژه PrjADO.vbp را باز کنید.

۲- دو کادر متن در بالای کنترل ADO Data اضافه کنید.

۳- مشخصه DataSource هر دو کادر متن را Adodc1 قرار دهید.

۴- مشخصه DataField کادر متن اول (Text1) را برابر با CompanyName

قرار دهید.

۵- مشخصه DataField کادر متن دوم (Text2) را برابر با Phone قرار دهید.

۶- پروژه را اجرا کنید.

شما می‌توانید از طریق کلیک کردن روی فلش‌های کنترل ADO Data، اسامی شرکت‌ها و شماره تلفن‌های آن‌ها را مشاهده کنید. ظاهر فرم شبیه شکل ۱۷-۴ خواهد بود.

۷- روی End کلیک کنید.

۸- پروژه را ذخیره کنید.



شکل ۱۷-۴- نام شرکت و شماره تلفن در کنترل‌های مقید نمایش داده می‌شوند.

۸-۴- برنامه‌نویسی بانک اطلاعاتی با کنترل‌های ADO

همان‌طور که می‌دانید از کنترل ADO Data می‌توان برای پیوند خودکار با یک بانک اطلاعاتی استفاده کرد. کنترل ADO Data روش خوبی برای مرور رکوردها و ویرایش یک رکورد در هر لحظه است. علاوه بر مقیدسازی داده‌ها با کنترل ADO Data، می‌توان یک فرم ورود داده‌های ساده را به سرعت ایجاد کرد.

۱-۸-۴- کدنویسی کنترل ADO Data: بعد از این که چگونگی تنظیم مشخصه‌های کنترل

ADO Data را برای اتصال به بانک اطلاعاتی و مقیدسازی کنترل‌ها را آموختید. در این قسمت، درباره کار کردن با داده‌ها از طریق این کنترل، آشنا خواهید شد. این کنترل، امکان مشاهده و ویرایش رکوردها بدون نوشتن کد را فراهم می‌کند. با این وجود، برای پشتیبانی ویژگی‌های پیشرفته‌تر، نیاز به نوشتن کد خواهید داشت. با استفاده از این کنترل و مشخصه‌ها، متدها و رویدادهای کنترل مقید، می‌توان کنترل پیش‌تری روی چگونگی تعامل با داده‌های خارجی اعمال کرد.

کنترل ADO Data دارای مشخصه‌ای به نام Recordset است که گروهی از رکوردها می‌باشد.

مشخصه Recordset این کنترل، خودش یک شیء است و دارای مشخصه‌ها و متدهای خاص خودش می‌باشد. شیء Recordset رکوردهایی را از جدول پایه یا حاصل اجرای یک پرس‌وجو ارائه می‌دهد. شیء‌های Recordset با استفاده از رکوردها (سطرها) و فیلدها (ستون‌ها) ساخته می‌شوند. می‌توان از این شیء‌ها برای کار کردن با داده‌های بانک اطلاعاتی استفاده کرد. در هر لحظه، شیء Recordset فقط به یک رکورد اشاره می‌کند که رکورد جاری نامیده می‌شود.

برای بازیابی مجموعه‌ای از رکوردها، مشخصه RecordSource کنترل ADO Data را مقدار دهی کنید. مشخصه RecordSource یک مقدار رشته‌ای است که می‌تواند نام جدول یا پرس‌وجو در یک بانک اطلاعاتی باشد.

همچنین می‌توان مشخصه Filter از شیء Recordset را برای انتخاب رکوردهای خاصی به کار برد. به عنوان مثال، می‌توان تعیین کرد که مجموعه رکورد شامل فقط رکوردهایی باشد که CustomerID آن‌ها بزرگتر از ده است.

```
adoCustomers.Recordset.Filter = "CustomerID>10"
```

بعد از این که مشخصه RecordSource مقداردهی شد، می‌توانید مشخصه‌های EOF و BOF شیء Recordset را بررسی کنید. این مشخصه‌ها تعیین می‌کنند که در ابتدا یا انتهای Recordset هستید. اگر رکورد جاری اولین رکورد باشد، مقدار BOF برابر با True خواهد بود و اگر اشاره‌گر در انتهای رکوردها باشد، EOF برابر True است. در صورتی که هیچ رکوردی در Recordset نباشد، مقدار هر دو مشخصه EOF و BOF برابر با True خواهد بود.

۲-۸-۴- ویرایش رکوردها در Recordset:

خودکار رکوردها را بدون نیاز به اضافه کردن کد، ویرایش و به‌هنگام کند، ولی می‌توان متد Update را برای ویرایش داده‌ها به جای استفاده از فلش‌های کنترل ADO، به کار گرفت.

کنترل ADO Data به‌طور خودکار رکوردهای بانک اطلاعاتی را در مواقع زیر تغییر می‌دهد:

- ۱- جابه‌جایی به رکوردی که می‌خواهید ویرایش کنید.
- ۲- تغییر هر نوع اطلاعاتی که در کنترل‌های مقید نمایش داده می‌شوند.
- ۳- کلیک کردن روی هر کدام از فلش‌های کنترل ADO Data برای جابه‌جایی به رکورد دیگر. با این وجود، به‌کارگیری دکمه فرمان امکان اضافه کردن کد به رویداد Click برای انجام کارهایی مثل بررسی صحت فیلدها روی فرم، قبل از به‌هنگام سازی را فراهم می‌کند. برای انجام عمل به‌هنگام سازی رکورد جاری، از متد Update استفاده کنید. به عنوان مثال، در رویداد Click دکمه‌ای

مانند Update، می‌توان کد زیر را اضافه کرد :

```
adoCustomers.Recordset.Update
```

در صورتی که می‌خواهید از تغییرات اعمال شده در رکورد جاری؛ صرف‌نظر کنید یا رکورد جدیدی را قبل از فراخوانی متد Update درج کنید، می‌توانید از متد Cancel Update استفاده کنید. شکل کلی متد Cancel Update در مثال زیر نشان داده شده است.

```
adoEmployees.Recordset.CancelUpdate
```

۳-۸-۴- اضافه کردن رکوردها به Recordset: رکوردهای جدید را با فراخوانی متد AddNew می‌توان به Recordset اضافه کرد. متد AddNew کنترل‌های مقید را مقداردهی اولیه می‌کند و رکورد جدید، رکورد جاری می‌شود. اگر هم‌زمان با ویرایش رکورد دیگری، AddNew را فراخوانی کنید، ADO به‌طور خودکار متد Update را فراخوانی می‌کند تا تغییرات را ذخیره کند و سپس یک رکورد جدید ایجاد می‌کند. کد مثال زیر، رکورد جدیدی را به مجموعه رکورد adoCustomers اضافه خواهد کرد :

```
adoCustomers.Recordset.AddNew
```

برای ذخیره تغییرات در رکورد جدید، می‌توان متد Update شیء Recordset را فراخوانی کرد یا می‌توان روی یکی از دکمه‌های پیمایش کنترل ADO Data کلیک کرد.

۴-۸-۴- حذف رکوردها از Recordset: با استفاده از متد Delete می‌توان رکورد جاری یا گروهی از رکوردها را در مجموعه رکورد حذف کرد. متد Delete دارای یک پارامتر AffectRecords است که برای تعیین تعداد رکوردهایی که این متد تحت تأثیر قرار خواهد داد، مورد استفاده قرار می‌گیرد.

جدول ۲-۴- مقادیر پارامتر Affect Records

مقدار AffectRecords	شرح
adAffectCurrent	این گزینه فقط رکورد جاری را حذف خواهد کرد و گزینه پیش‌فرض است.
adAffectGroup	این گزینه می‌تواند حذف تمام رکوردهایی که از مشخصه Filter عبور کرده‌اند را دربرگیرد. برای استفاده از این گزینه باید ابتدا مشخصه Filter را مقداردهی کرد.

بازیابی مقادیر فیلد از رکوردهای حذف شده، خطایی را تولید می‌کند. بعد از حذف رکورد جاری، رکورد حذف شده جاری باقی می‌ماند تا زمانی که به رکورد دیگری جابه‌جا شوید. بعد از این که از رکورد حذف شده جابه‌جا شدید، در دسترس نخواهد بود. برای به‌کارگیری متد Delete از شکل کلی زیر استفاده کنید :


Recordset.Delete AffectRecords

برای مشاهده این که آخرین رکورد حذف شده است یا نه، مشخصه EOF را بررسی کنید. اگر EOF برابر با True بود، به آخرین رکورد منتقل شوید.

```

حذف رکورد جاری /
adoCustomers.Recordset.Delete
جابه‌جایی به رکورد بعدی /
adoCustomers.Recordset.MoveNext
آیا آخرین رکورد حذف شده است؟ /
If adoCustomers.Recordset.EOF=True Then
انتقال به آخرین رکورد جدید /
adoCustomers.Recordset.MoveLast
End IF

```

 **نکته :** بانک اطلاعاتی Northwind دارای قواعد جامعیت ارجاعی

(referential integrity) تعریف شده است که از حذف رکوردهای Recordset

جلوگیری می‌کند. به عنوان مثال، نمی‌توان یک رکورد از جدول Customer را حذف

کرد، در صورتی که رکوردهای مرتبطی در جدول orders داشته باشد.

۵-۸-۴- جستجوی رکوردها : برای افزودن ویژگی جستجو به برنامه کاربردی‌تان، از متد

Find مربوط به مجموعه رکورد (Recordset) کنترل ADO Data استفاده کنید. متد Find مجموعه رکورد موجود را برای رکوردی که دارای شرط خاصی است، جستجو می‌کند. اگر شرط بارکوردی مطابقت داشته باشد، اشاره‌گر روی آن رکورد قرار می‌گیرد، در غیر این صورت به انتهای مجموعه رکورد (EOF) منتقل می‌شود. شکل کلی متد Find به صورت زیر است.

Recordset.Find Criteria, [SkipRows],[Search Direction],[Start]

متد Find دارای پارامتر ضروری شرط (Criteria) و سه پارامتر اختیاری است که عبارتند از :

Start و SearchDirection، SkipRows.

پارامتر شرط (Criteria) رشته‌ای است که شامل عبارتی است که نام ستون، عملگر مقایسه و

مقدار مورد استفاده در جستجو را شامل می‌شود. عملگر مقایسه در شرط ممکن است، بزرگتر (>)، کوچکتر (<)، مساوی (=) یا like (مطابقت الگو) باشد.


در مثال زیر، اولین مشتری که در ایالت واشنگتن زندگی می‌کند، برگردانده خواهد شد:

```
adoCustomers.Recordset.Find "state='WA'"
```

مقدار شرط ممکن است یک رشته، عدد ممیز شناور یا تاریخ باشد. مقادیر رشته‌ای با تک کوتیشن مشخص می‌شوند (به عنوان مثال، "state='WA'"). و مقادیر تاریخ با علامت # مشخص می‌شوند (به عنوان مثال، #7/2/93#)

برای جستجو براساس بخشی از یک رشته، می‌توان از کلید واژه Like موجود در SQL استفاده کرد. به خاطر داشته باشید که باید تک کوتیشن در شروع و پایان رشته قرار دهید. اگر عملگر مقایسه Like باشد می‌توانید به عنوان جایگزین یک یا چند نویسه از علامت % و جایگزین یک کاراکتر از علامت Dash استفاده کنید. کد زیر چگونگی به کارگیری این عملگر را نشان می‌دهد.

```
adostates.Recordset.Find "state LIKE 'M%'"
```

 **نکته:** متد Find می‌تواند کند باشد، مگر این که تعداد کل رکوردها در مجموعه رکورد را محدود کنید. روش دیگر برای پیدا کردن رکوردها، استفاده از عبارت SQL هنگام مقداردهی مشخصه RecordSource است. همچنین می‌توانید از مشخصه Filter در کنترل ADO برای محدود کردن تعداد رکوردهای مجموعه رکورد استفاده کنید.

سه پارامتر اختیاری برای متد Find شیء Recordset وجود دارد:

• **SkipRows:** این یک نوع داده Long است که مقدار پیش فرض آن صفر است. از این پارامتر برای تعیین فاصله از سطر جاری یا نشانه آغازین برای شروع جستجو استفاده می‌شود.

• **SearchDirection:** این پارامتر تعیین می‌کند جستجو باید از سطر جاری تا آخرین رکورد یا سطر جاری تا اولین رکورد انجام شود. جستجو در شروع یا پایان مجموعه رکورد خاتمه می‌یابد که بستگی به مقدار SearchDirection دارد. این پارامتر می‌تواند یکی از مقادیر عددی زیر باشد:

– adsearch Forward(0) – جستجو به سمت جلو از رکورد جاری در صورتی که جستجو

با موفقیت انجام نشود، EOF برمی‌گرداند.

– adsearch Backward(1) – جستجو به سمت عقب از رکورد جاری در صورتی که جستجو با موفقیت انجام نشود، BOF برمی گرداند.

• **Start:** این پارامتر، یک نشانه Variant است که به عنوان محل شروع جستجو مورد استفاده قرار می گیرد. مقدار این پارامتر می تواند یکی از مقادیر عددی زیر باشد :

– adBookmarkCurrent (0) – رکورد جاری

– adBookmark First (1) – اولین رکورد

– adBookmark Last (2) – آخرین رکورد

هنگامی که رکوردها را با استفاده از متد Find جستجو می کنید، می توانید مشخصه های EOF و BOF از شیء Recordset را به کار بگیرید تا تعیین کنید که آیا رکورد خاصی پیدا شده است یا نه؟ اگر جستجو ناموفق باشد، باید به رکوردی که جستجو شروع شده است، برگردید. کد زیر چگونگی استفاده از این مشخصه ها را نشان می دهد :

```
'If the record isn't found
If .EOF or .BOF Then
    'Return to the starting record
    .Bookmark=varBookmark
    MsgBox "Record not found."
```

End If

مشخصه های BOF، EOF و Bookmark از مشخصه های Recordset می باشند.
لذا این قطعه کد باید داخل بلوک

Recordset. نام کنترل ADO Data With

·
·
·

End With

قرار گیرد.

مشخصه Bookmark برای علامت گذاری رکورد معینی در شیء Recordset استفاده می شود.
برای این منظور با استفاده از متد Move روی رکورد رفته از دستور زیر استفاده می کنیم؛

Dim VarBookmark As Variant

VarBookmark = ADodc . Recordset . Bookmark

و هرگاه بخواهیم به رکورد علامت گذاری شده برگردیم از دستور زیر استفاده می کنیم؛

ADodc . Recordset . Bookmark = VarBook Mark

مثال زیر، متد Find را با استفاده از هر چهار پارامتر پیاده سازی می کند :

Dim VarBookmark As Variant

With adoFood . Recordset

 'mark the current record

 varBookmark = . Bookmark

 'specify the search criteria, start and direction

 .Find "CategoryName='Condiments'",0, adsearchForward, _

adBookmarkCurrent

 'If the record isn't found

 If . EoF or . BOF then

 'Return to the starting record

 .Bookmark = varBookmark

 MsgBox "Record not found."

 End If

End with

تمرین: جستجوی رکوردها در یک Recordset : در این تمرین، پروژه ای 

که در دو تمرین قبلی این فصل ایجاد کردید را ادامه خواهیم داد. عملیاتی را به برنامه کاربردی اضافه خواهیم کرد تا اسامی شرکت ها را جستجو کند.

۱- پروژه prjADO.vbp که در تمرین قبل ذخیره کردید را باز کنید.

۲- بین دو کنترل Text2 و Adodc1 یک دکمه فرمان اضافه کنید.

۳- در پنجره Properties، مشخصه Name کنترل Command1 را به

CmdSearch تغییر دهید.

۴- مشخصه Caption کنترل CmdSearch را به search & تغییر دهید.

۵- در رویداد Click دکمه فرمان Cmdsearch، کد زیر را تایپ کنید.

```
Adodc1.Recordset.MoveFirst
```

```
Adodc1.Recordset.Find "CompanyName='FrankenVersand'"
```

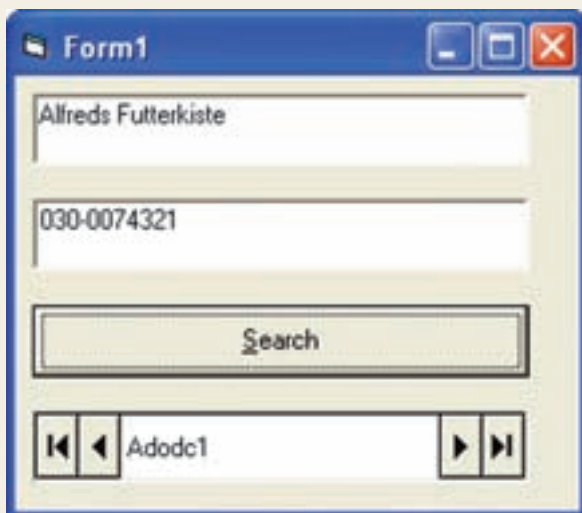
```
If Adodc1.Recordset.EOF Then MsgBox "Record not found."
```

۶- از منوی Run گزینه Start را انتخاب کنید.

۷- روی دکمه Search کلیک کنید.

توجه کنید که رکورد حاوی نام شرکت Frankenversand رکورد جاری خواهد


شد. برنامه کاربردی شبیه شکل ۱۸-۴ خواهد بود.



شکل ۱۸-۴- استفاده از متد Find برای جستجو

۸- روی End کلیک کنید.

۹- پروژه را ذخیره کنید.

 **تمرین:** روی خطی که متد Find قرار دارد Break Point گذاشته و مشخصه‌های کنترل ADodc1 را با کمک پنجره Watch مشاهده کنید. در فرم‌های طراحی شده مثال‌های قبل امکان نمایش تنها یک رکورد وجود داشت. حال می‌خواهیم از کنترل‌هایی استفاده کنیم که به ما امکان نمایش چندین رکورد را به‌طور همزمان می‌دهد.

جدول ۲-۴ - متدهای شیء Recordset

متد	شرح
AddNew	اضافه کردن رکورد جدید به Recordset
CancelUpdate	انصراف از تغییرات انجام شده که در حافظه ذخیره شده است.
Close	بستن Recordset
Delete	حذف کردن رکورد از Recordset
Move	جابجایی در Recordset
MoveFirst	جابجایی به اولین رکورد Recordset
MoveLast	جابجایی به آخرین رکورد Recordset
MoveNext	جابجایی به رکورد بعدی در Recordset
MovePrevious	جابجایی به رکورد قبلی در Recordset
Update	به روز کردن تغییرات انجام شده در Recordset
find	به منظور پیدا کردن رکوردهایی با ویژگی‌های معین در Recordset (جستجو در مجموعه رکوردها)

جدول ۳-۴ - مشخصه‌های شیء Recordset

مشخصه	شرح
BOF	نشان می‌دهد که رکورد جاری قبل از اولین رکورد جدول می‌باشد (اشاره گر رکورد از جدول خارج شده است و در بالای جدول قرار دارد.)
Bookmark	این مشخصه برای علامت گذاری یک رکورد معین استفاده می‌شود.
EOF	نشان می‌دهد که رکورد جاری بعد از آخرین رکورد جدول می‌باشد (اشاره گر رکورد از جدول خارج شده است و در پایین جدول قرار دارد.)
RecordCount	از نوع long است و تعداد رکوردهای موجود در Recordset را نشان می‌دهد.
Source	منبع داده در Recordset را نشان می‌دهد که می‌تواند عبارت SQL یا نام جدول باشد.

برای علامت گذاری رکورد، به کمک متدهای move روی رکورد رفته و از دستور زیر استفاده می کنیم؛

Dim varBookmark as variant

varBookmark =adodc.Recordset.bookmark

و هرگاه بخواهیم به رکورد علامت گذاری شده برگردیم از دستور زیر استفاده می کنیم؛

adodc.Recordset.bookmark =varBookmark



در این مثال فرمی مانند شکل ۴-۱۹ ایجاد کنید .



شکل ۴-۱۹- فرم نمایش رکوردهای جدول Customers

ابتدا نام کنترل adoCustomers را قرار داده و آن را به بانک NWind.MDB متصل کرده، نوع فرمان را در بخش Command Type مساوی adCmdText قرار داده و در بخش command Text(SQL) دستور زیر را وارد کنید.

Select * from customers

این دستور سبب می شود که تمام فیلدهای جدول customers از طریق کنترل adoCustomers قابل دسترسی باشند.

نام کنترل های TextBox را با توجه به فیلدی که نمایش می دهند و نام دکمه های فرمان را با توجه به عملیاتی که انجام می دهند، تغییر دهید. (txtCustomerId, txtCompanyName, txtPhone, cmdAdd, cmdUpdate, cmdDelete, cmdRefresh, cmdSort, cmdSearch, txtCity)

اضافه کردن رکورد به جدول :

برای اضافه کردن رکورد به جدول، قطعه کد زیر را در رویداد کلیک دکمه فرمان ADD اضافه کنید.

```
Private Sub CmdAdd_Click()  
    AdoCustomers.Recordset.AddNew  
End Sub
```

پس از اجرای فرمان فوق در انتهای جدول یک سطر خالی ایجاد می‌شود تا بتوان رکورد جدیدی از اطلاعات را به انتهای جدول اضافه کرد.

ثبت اطلاعات در جدول :

بعد از اجرای فرمان فوق برای ثبت اطلاعات وارد شده و یا برای ثبت تغییرات در برنامه باید فرمان زیر را اجرا کنیم. قطعه کد زیر را در بخش کد دکمه فرمان Update وارد نمایید.

```
Private Sub cmdUpdate_Click()  
    AdoCustomers.Recordset.Update  
End Sub
```

حذف اطلاعات در جدول :

برای حذف اطلاعات از جدول باید از فرمان Delete استفاده کنیم. قطعه کد زیر را در بخش کد دکمه فرمان Delete وارد نمایید.

```
Private Sub cmdDelete_Click()  
    AdoCustomers.Recordset.Delete  
End Sub
```

لغو عملیات‌های روی جدول :

اگر بخواهیم عملیات انجام شده روی جدول را لغو کنیم تا از ثبت تغییرات جلوگیری شود، باید Refresh از فرمان استفاده کنیم. قطعه کد زیر را در بخش کد دکمه فرمان Refresh وارد کنید. یکی دیگر از کاربردهای این دستور برای فراخوانی مجدد اطلاعات از بانک اطلاعاتی می‌باشد.

```
Private Sub cmd_Refresh_Click()  
    AdoCustomers.Refresh  
End Sub
```

استفاده از دستورات SQL :

برای استفاده از دستورات SQL در VB باید کدهای مورد نظر را در بخش RecordSource کنترل ADO وارد و سپس فرمان Refresh را اجرا کرد. می‌خواهیم در رویداد کلیک دکمه Sort اطلاعات جدول براساس فیلد CompanyName مرتب شود. کد زیر را در رویداد کلیک این دکمه بنویسید.

```
Private Sub cmdSort_Click()  
    AdoCustomers.RecordSource = "select * from customers order by  
    CompanyName"  
    AdoCustomers.Refresh  
End Sub
```

می‌خواهیم رکوردهایی را جستجو کنیم که فیلد City آن‌ها مساوی محتوای txtCity باشد لذا در رویداد کلیک دکمه Search کد زیر را بنویسید.

```
Private Sub cmdSearch_Click()  
    AdoCustomers.RecordSource = "SELECT * FROM customers  
    WHERE City="' & txtCity.Text & "' "  
    AdoCustomers.Refresh  
End Sub
```

اگر بخواهید جستجو براساس نام کمپانی و نام شهر انجام شود در رویداد کلیک دکمه Search و کد زیر را بنویسید.

```
Private Sub cmdSearch_Click()  
    AdoCustomers.RecordSource = "SELECT * FROM customers WHERE  
    CompanyName ='" & txtCompanyName.Text & "' & "AND City  
    ='" & txtCity.Text & "' "  
    AdoCustomers.Refresh  
End Sub
```

توجه کنید که مقادیر رشته‌ای و با تک کوتیشن مشخص می‌شوند.

۴-۹- کاربرد کنترل DataGrid

اولین کنترلی که به کار خواهیم برد، کنترل DataGrid است که امکان مشاهده و ویرایش چندین سطر از داده‌ها را به طور هم‌زمان فراهم می‌کند. هم‌چنین DataGrid برای ورود سریع مقادیر زیادی از داده‌ها مفید است. برای استفاده از این کنترل، باید آن را به جعبه ابزار ویژوال بیسیک اضافه کنید. مراحل زیر، چگونگی اضافه کردن این کنترل را نشان می‌دهند:

۱- از منوی Project گزینه Components را انتخاب کنید.

۲- Microsoft DataGrid Control 6.0 را از لیست انتخاب کرده و سپس روی OK کلیک

کنید. کنترل DataGrid در جعبه ابزار نشان داده خواهد شد.

اکنون می‌توانید DataGrid را به فرم اضافه کنید. روی آیکن DataGrid در جعبه ابزار دابل

کلیک کنید. کنترل را بزرگ کنید تا کل فرم را دربر بگیرد.

در صورتی که برنامه را اجرا کنید، DataGrid به طور کامل خالی خواهد بود زیرا هنوز آن را

به کنترل ADO Data پیوند نکرده‌اید.

مثال ۳-۴

پیوند Data Grid به کنترل ADO Data :

۱- روی کنترل DataGrid در روی فرم کلیک کنید. مشخصه Name را با dgrdCustomers

مقداردهی کنید، زیرا اطلاعات مشتری را نمایش خواهد داد.

۲- adoCustomers را از لیست بازشوی مشخصه DataSource انتخاب کنید (این همان

کنترل ADO Data است که قبلاً ایجاد شده است).

بعد از تعیین DataSource برای DataGrid، کنترل DataGrid به طور خودکار

تعداد سطرها و ستون‌ها را پیکربندی می‌کند. برای مشاهده فرم در عمل، از منوی Run

گزینه Start را انتخاب کنید. پنجره، داده‌های جدول Customers را نشان خواهد داد

(شکل ۴-۲۰).

به دلیل این که تنظیمات پیش فرض همیشه بهترین نیستند، DataGrid به طور کامل قابل پیکربندی

بوده و امکان نمایش ستون‌های خاص، قالب‌بندی و رنگ هر ستون را فراهم می‌کند. هم‌چنین می‌توان

DataGrid را به چندین بخش تقسیم کرد که هر کدام به طور مستقل از دیگری قابل مشاهده و مرور

باشند. انجام این کار در Excel و سایر صفحات گسترده متداول است.

CustomerID	CompanyName	ContactName
ALFKJ	Alfreds Futterkiste	Maria Anders
ANATR	Ana Trujillo Emparedados y heladerias	Ana Trujillo
ANTON	Antonio Moreno Taqueria	Antonio Moreno
AROUT	Around the Horn	Thomas Hardy
BERGS	Berglunds snabbköp	Christina Berglund
BLAUS	Blauer See Delikatessen	Hanna Moos
BLONP	Blondel père et fils	Frédérique Citeaux
BOLID	Bólido Comidas preparadas	Martín Sommer
BONAP	Bon app'	Laurence Leblond
BOTTM	Bottom-Dollar Markets	Elizabeth Lincoln
BSBEV	B's Beverages	Victoria Ashworth

شکل ۲۰-۴- DataGrid می تواند به طور خودکار خودش را بپیکربندی کند تا داده های منبع تعیین شده را نمایش دهد.

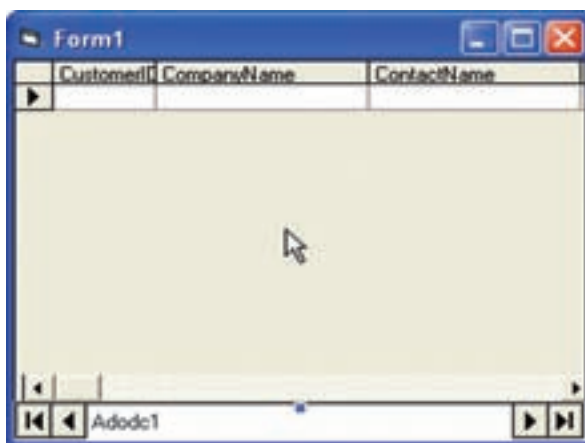
۱-۹-۴- انتخاب ستون ها :

به دلیل اینکه جدول های بانک اطلاعاتی با گذشت زمان، بزرگ می شوند، همیشه نمی خواهید که تمام ستون های جدول را به کاربران نشان دهید. در این قسمت، چگونگی تعیین ستون ها برای نمایش در کنترل DataGrid را نشان خواهیم داد. قبل از شروع، باید تصمیم بگیرید که کدام ستون ها در DataGrid نمایش داده شوند. برای این مثال، DataGrid ستون های Company Name، Contact Name، Phone Number و Country را نشان خواهد داد.

مثال ۴-۴

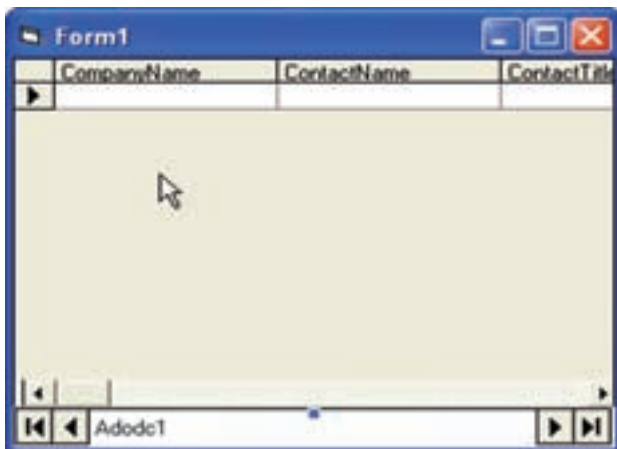
تعیین ستون ها در DataGrid:

۱- روی DataGrid کلیک راست کرده و از منوی میان بر، Retrieve Fields را انتخاب کنید.



هنگامی که ویزوال بیسیک درباره جایگزینی طرح کلی سؤال می کند، Yes را انتخاب کنید. تمام فیلدها از Recordsource انتخاب شده (در این حالت، جدول Customers) در کنترل DataGrid بارگذاری خواهند شد (شکل ۲۱-۴).

شکل ۲۱-۴- کنترل DataGrid تمام فیلدها را از جدول بارگذاری می کند بنابراین می توان یکی از آن ها را انتخاب کرده و مشاهده کرد.



۲- روی DataGrid کلیک

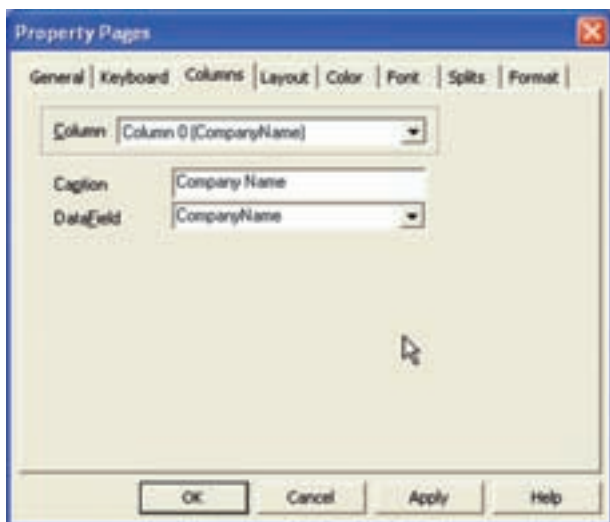
راست کرده و گزینه Edit را از منوی میان بر انتخاب کنید. اکنون می توانید روی ستون های خاصی کلیک کرده و آن ها را ویرایش کنید.

• برای حذف ستون ها از لیست، روی فیلد CustomerID کلیک راست کرده و Delete را انتخاب کنید (شکل ۲۲-۴).

شکل ۲۲-۴- هنگامی که DataGrid در حالت Edit است، می توان ستون ها را اضافه و حذف کرد.

۳- ستون های Contact Title، Address، City، Region، PostalCode و Fax را حذف کنید.

• در این کنترل نمی توان ستون ها را با استفاده از کشیدن و رها کردن دوباره مرتب کرد. به عنوان مثال، اگر می خواهید ستون های ContactName و CompanyName را جابه جا کنید، ابتدا باید ستون ContactName را Cut کرده و سپس روی CompanyName کلیک راست کرده و



Paste را انتخاب کنید. DataGrid تمام تنظیمات مربوط به یک ستون را نگه خواهد داشت.

۴- هنگام نمایش داده ها در DataGrid، ضروری نیست که عنوان ستون ها همان نام فیلدها باشد. برای تغییر عنوان ستون ها، روی ستون کلیک راست کرده و Properties را انتخاب کنید. روی صفحه Columns نام جدیدی را در کادر متن Caption وارد کنید (شکل ۲۳-۴).

شکل ۲۳-۴- مشخصه Caption صفحه Columns کنترل می کند که چه چیزی در عنوان هر ستون نمایش داده خواهد شد.

۵- مرحله ۶ را برای تغییر عنوان هر ستون موردنظر، تکرار کنید.

۲-۹-۴- **پر کردن فرم:** می‌توان مشخصه Visible کنترل ADO Data را با False

مقدار دهی کرد تا کنترل ADOData برای کاربران قابل مشاهده نباشد. هم‌چنین می‌توان DataGrid را به اندازه‌ای بزرگ کرد که فرم را پر کند. ابتدا مشخصه Align را با vbAlignTop مقداردهی کنید تا DataGrid در بالای فرم قرار گرفته و از چپ به راست فرم را پر کند. سپس کد زیر را اضافه کنید تا از بالا به پایین فرم را پر کند:

```
Private Sub Form_Resize()
```

```
    dgrdCustomers.Height=Me.scaleHeight
```

```
End sub
```

Company Name	ContactName	ContactTitle
Alfreds Futterkiste	Maria Anders	Sales Representative
Ana Trujillo Emparedados y heladerias	Ana Trujillo	Owner
Antonio Moreno Taqueria	Antonio Moreno	Owner
Around the Horn	Thomas Hardy	Sales Representative
Berglunds snabbkop	Christina Berglund	Order Administrator
Blauer See Delikatessen	Hanna Moos	Sales Representative
Blondel père et fils	Frédérique Citeaux	Marketing Assistant
Bólido Comidas preparadas	Martín Sommer	Owner
Bon app'	Laurence Leblanc	Owner
Bottom-Dollar Markets	Elizabeth Lincoln	Accountant
B's Beverages	Victoria Ashworth	Sales Representative
Cactus Comidas para llevar	Patricio Simpson	Sales Agent
Centro comercial Moctezuma	Francisco Chang	Marketing Assistant
Chop-suey Chinese	Yang Wang	Owner
Comércio Mineiro	Pedro Afonso	Sales Assistant
Consolidated Holdings	Elizabeth Brown	Sales Representative
Drachenblut Delikatessen	Sven Ottlieb	Order Administrator
Du monde entier	Janine Labrune	Owner
Eastern Connection	Ann Devon	Sales Agent
Ernst Handel	Roland Mendel	Sales Manager
Familia Arquibaldo	Aria Cruz	Marketing Assistant
FISSA Fabrica Inter. Salchichas	Diego Roel	Accountant
Foiles gourmandes	Martine Rancé	Assistant Sales Representative
Folk och färgvarer - HB	Maria Larsson	Owner


شکل ۴-۲۴ DataGrid کامل شده است.


به عنوان یک کار پایانی، عنوانی مثل Customer Viewer را در مشخصه Caption فرم قرار دهید تا فرم حرفه‌ای تر به نظر برسد.

۳-۹-۴- پیکربندی سایر مشخصه‌های DataGrid: می‌توان چندین مشخصه دیگر را روی DataGrid پیکربندی کرد که آن‌ها نیز به سادگی می‌توانند مورد استفاده قرار گیرند. جدول ۴-۵ هدف هر مشخصه را شرح می‌دهد.

جدول ۴-۵. مشخصه‌های DataGrid

مشخصه	شرح
Align	تعیین می‌کند که DataGrid نسبت به بالا، چپ، راست یا پایین تنظیم شود.
AllowAddNew	اگر با True مقداردهی شود، رکوردهای جدید را می‌توان در پایین DataGrid اضافه کرد.
AllowArrows	اگر با True مقداردهی شود، با استفاده از کلیدهای جهت‌دار می‌توان پیمایش را انجام داد.
AllowDelete	اگر با True مقداردهی شود، با فشار دادن کلید Delete می‌توان سطرها را از DataGrid حذف کرد.
AllowUpdate	اگر با True مقداردهی شود، می‌توان سطرها را ویرایش کرد و تغییرات به‌طور خودکار ذخیره می‌شوند.
CausesValidation	اگر با True مقداردهی شود، رویداد Validate هنگامی که فوکوس به کنترل دیگری منتقل می‌شود، فعال خواهد شد.
ColumnHeaders	عنوان ستون را فعال یا غیرفعال می‌کند.
HeadLines	تعداد خطوط عمودی برای عنوان‌های ستون را تعیین می‌کند.
RowDivider Style	گزینه‌های مختلفی را برای قالب‌بندی جدا کننده‌های سطرها ارائه می‌کند.
TabAcrossSplits	کلید Tab را برای انتقال بین بخش‌های DataGrid فعال می‌کند.
TabAction	پاسخ به فشار دادن کلید Tab را تعیین می‌کند.
WrapcellPointer	تعیین می‌کند که آیا مکان‌نما باید هنگام رسیدن به انتهای یک سطر به سطر بعدی و اولین ستون منتقل شود؟

 **نکته:** در صورتی که رکورد با جدول های دیگر مرتبط است، حذف رکورد ممکن نخواهد بود و سبب بروز پیام خطا می شود. حذف کردن فقط هنگامی ممکن است که تمام رکوردهای مرتبط حذف شده باشند. (Refrential Integrity)

 **نکته:** هنگامی که یک برنامه کاربردی را طراحی و درباره نمایش داده های بانک اطلاعاتی آن فکر می کنید، باید بدانید که اگر داده ها را در یک جدول ذخیره کنید می توانید آن ها را در DataGrid ویرایش کنید. در صورتی که نیاز به قراردادن داده ها در بیش از یک جدول دارید، باید نوع دیگری از فرم را به کار ببرید. به عنوان مثال، یک سفارش هم اطلاعاتی درباره سفارش و هم کالاهای سفارش شده خواهد داشت. سفارش را نمی توانید در یک grid ویرایش کنید. زیرا داده ها حداقل در دو جدول متفاوت ذخیره شده اند. با این وجود، DataGrid می تواند لیستی از کالاها را نشان دهد. در صورتی که فقط می خواهید داده های سایر جدول ها را مشاهده کنید، DataGrid هنوز هم گزینه مناسبی است.

۴-۱۰- کاربرد کنترل DataList

کنترل DataList خیلی ساده تر از کنترل DataGrid است. از این کنترل برای نمایش لیست یک ستونی از داده های جدول استفاده می شود. به دلیل این که این کنترل مقید به داده هاست، هیچ محدودیت حافظه برای نمایش تعداد عناصر ندارد. ولی اضافه کردن عناصر خیلی زیاد سبب می شود که پیدا کردن عنصر خاصی برای کاربران مشکل باشد. شبیه DataGrid کنترل DataList نیز از کنترل ADO Data استفاده می کند.

 مثال ۴-۵

ایجاد و استفاده از کنترل DataList:

- ۱- روی فرم جدید، یک کنترل ADO Data اضافه کنید که به جدول Customers از بانک اطلاعاتی Nwind.mdb دسترسی داشته باشد.
- ۲- کنترل DataList را از طریق کادر محاوره ای Components به جعبه ابزار اضافه کنید (در

کادر محاوره‌ای، گزینهٔ Microsoft DataList Control 6.0 را انتخاب کنید).

۳- در جعبه ابزار روی کنترل DataList دابل کلیک کنید تا به فرم اضافه شود.

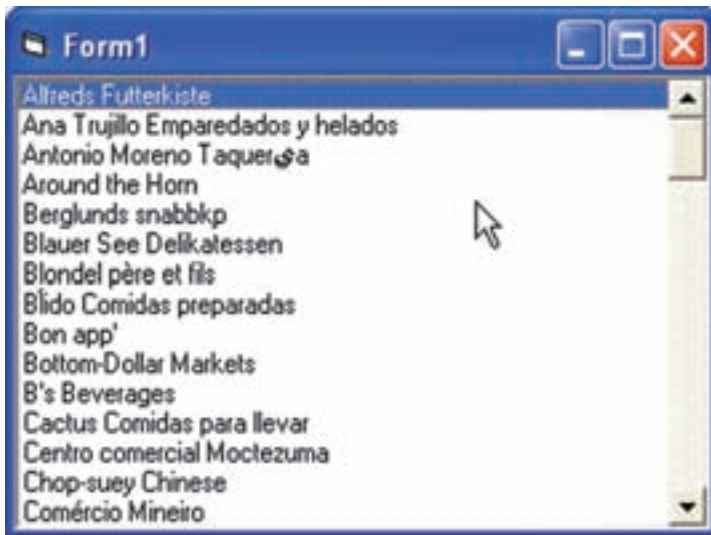
۴- نام کنترل DataList را dlstCustomers قرار دهید.

۵- مشخصهٔ RowSource را با adoCustomers مقداردهی کنید. این مشخصه منبع داده‌هایی

که لیستی از داده‌ها را برای نمایش ارائه خواهد کرد، مشخص می‌کند. برای تعیین فیلد موردنظر،

مشخصهٔ ListField را با CompanyName مقداردهی کنید.

۶- فرم را ذخیره کرده و پروژه را اجرا کنید (شکل ۲۵-۴).



شکل ۲۵-۴- کنترل DataList ستونی از داده‌های منبع داده را نشان می‌دهد.

هم‌چنین می‌توان روش پرکردن فرم که در قسمت قبلی آموختید را برای این فرم نیز به کار ببرید.

برای اینکه DataList به‌طور خودکار کل فرم را پر کند، کد زیر را به فرم اضافه کنید

```
Private sub Form_Resize( )
```

```
    dlstCustomers.Height = Me.scaleHeight
```

ارتفاع کنترل را مساوی ارتفاع فرم قرار می‌دهیم'

```
    dlstCustomers.Width = Me.scaleWidth
```

پهنای کنترل را مساوی پهنای فرم قرار می‌دهیم'

```
End sub
```

۱-۱۰-۴- پیکربندی مشخصه‌های کنترل DataList: همان‌طور که بیان شد، کنترل

DataList دارای مشخصه‌های کمتری نسبت به DataGridView است ولی چند ویژگی دارد که کنترل DataGridView ندارد (جدول ۴-۶).

جدول ۴-۶- برخی از مشخصه‌های کنترل DataList

مشخصه	شرح
CausesValidation	در صورتی که با True مقداردهی شود، رویداد Validate پس از انتقال فوکوس به کنترل دیگری، فعال می‌شود.
IntegralHeight	تعیین می‌کند که آیا کنترل DataList بخش‌هایی از عناصر را در صورتی که به اندازه کافی بزرگ نباشد، نشان دهد یا نه؟
MatchEntry	همزمان با تایپ کاربر، DataList براساس نویسه وارد شده به عنصر موردنظر رجوع می‌کند.

۱-۱۱-۴- کاربرد کنترل DataCombo

کنترل DataCombo دقیقاً شبیه کنترل DataList است و فقط ظاهر آن‌ها متفاوت است. این کنترل به جای این که فضای عمودی زیادی را اشغال کند، در صورت نیاز بخش لیست را باز می‌کند. این کار هنگامی مفید است که نیاز به فرم کوچک داشته باشیم. کنترل DataCombo شبیه کنترل ComboBox استاندارد است و برای تعداد کم (کمتر از ۱۰۰) مورد استفاده قرار خواهد گرفت.



ایجاد و استفاده از کنترل DataCombo:

- ۱- روی فرم جدید، کنترل ADO Data جدیدی را اضافه کنید که به جدول Customers از بانک اطلاعاتی Nwind.mdb دسترسی داشته باشد. به دلیل این که کنترل‌های DataList و DataCombo با هم هستند، پس کنترل DataCombo روی جعبه ابزار وجود خواهد داشت.
- ۲- روی کنترل DataCombo دابل کلیک کنید. نام آن را به دلیل این که برای نگهداری اسامی مشتریان مورد استفاده قرار می‌گیرد، dcboCustomers قرار دهید.
- ۳- مشخصه RowsSource کنترل DataCombo را با adoCustomers و مشخصه ListField را با CompanyName مقداردهی کنید (شبیه کنترل DataList عمل کنید).

هنگامی که برنامه را اجرا کنید، لیست مشتریان که در کنترل DataCombo لیست شده‌اند را مشاهده خواهید کرد (شکل ۴-۲۶).



شکل ۴-۲۶- کنترل DataCombo محتوای فیلد CompanyName را از جدول customers بارگذاری می‌کند.

مشخصه‌های قابل بیکربندی برای کنترل DataCombo همان مشخصه‌های DataList هستند. تنها تفاوت مشخصه Style است که شبیه مشخصه Style کنترل ComboBox عمل می‌کند. DataCombo می‌تواند فقط خواندنی یا قابل ویرایش بوده و به کاربران امکان اضافه کردن رکوردی به لیست موجود را فراهم کند. به دلیل این که DataCombo اغلب برای انتخاب مقادیر از لیست، مورد استفاده قرار می‌گیرد، معمولاً مقدار مشخصه Style آن را 2-DropDown List انتخاب می‌کنیم.

خودآزمایی

- ۱- سه رابط دسترسی به داده‌ها در ویژوال بیسیک، کدامند؟
 - ۲- دلیل استفاده از ADO را توضیح دهید.
 - ۳- از SQL به چه منظوری استفاده می‌شود؟
 - ۴- در SQL از چه عملگری برای جستجوی مقادیر براساس یک الگو استفاده می‌شود؟
 - ۵- دو مشخصه‌ای که برای اتصال کنترل ADO Data به بانک اطلاعاتی باید مقداردهی شوند، کدامند؟
 - ۶- دو روش برای به‌هنگام کردن رکوردها را شرح دهید.
 - ۷- پروژه‌ای طراحی کنید که از طریق آن بتوانید مشخصات CDهای موجود در هنرستان را در آن ذخیره کنید.
- توضیح: ابتدا بانک اطلاعاتی مربوطه را به کمک مربی خودتان طراحی و پیاده‌سازی کنید و سپس به کمک کنترل ADO Data رابط کاربری مناسبی برای انجام عملیات درج، حذف، ویرایش و جستجوی رکوردها در ویژوال بیسیک ایجاد کنید.