



به نام خدا

دانشکده فنی شهید شمسی پور

تهران

عنوان پروژه پایان دوره کاردانی :

نرم افزار کارآموز نسخه 1

این اثر کاملاً رایگان است. هدف تنها یادگیری است.

با تشکر از سایت پروژه دات کام

mahdikoochooloo@gmail.com

اگر کمک تو زمینه دیگر خواستی میل بده

1388

در قرن 21 دنیای برنامه نویسی متحول شده و به سمت شیء گرایی و شبکه پیش می رود. از این رو به نظر من بهترین تجربه برای یک برنامه نویس ، نوشتن یک پروژه به کمک شبکه است. این تفکر من را به وا داشت که یک برنامه برنامه تحت شبکه بنویسم.

کل این برنامه شامل 2 قسمت سرور و کلاینت که همان کارآموز می باشد می شود. از آن با کمی تغییر می توان به عنوان یک برنامه کافی نت و همچنین برنامه کنترل سیستم های شاگردان و هنرجویان و ... استفاده کرد.

ویژگی اصلی آن وجود امکان شبکه ای آن و اتصال بیش از یک کلاینت به آن می باشد. محیط بسیار کاربر پسند و گرم باعث شده که به آن به دید یک نرم افزار دوست و بسته خوب نگاه کرد.

تفکر اصلی آن از وقتی شروع شد که به عنوان کارآموز در شرکت نفت فعالیت می کردم با این مشکل برخوردیم که همه با صدا و فریاد و دست نوشته کار می کردند. مثلاً 5 تا کالا را می خواهم پس باید فاکتور آن را بنویسم و به خودم فاکتور را تحویل داده و برگردانم. همچنین اگر سیستمی خراب شد باید به صورت دستی به آن برچسب زده و آن را به کارآموز به صورت دستی تحویل داده و به وی بگوییم که فلان کارها را انجام بده و از همه بدتر یک دفتر چه هست که در آن تمامی فعالیت ها را یادداشت می کردیم. مثلاً فلان سیستم با فلان شماره و ... آورده شده ، من آن را تعمیر کردم به این مشکل و با این راه حل ها و قطعات.

از این رو برنامه ای نوشتم که تمامی این کارها را تحت تاثیر قرار می داد به عنوان مثال دیگر نیاز نبود سیستم را دستی از مشتری تحویل گرفته و رویش کار انجام دهیم بلکه بسیار راحت آن را دریافت کرده و شماره را به کامپیوتر می دادیم و با کامپیوتر به کارآموز تحویل می دهیم و وی کار را انجام داده و جوابش را به ما می دهد

یادگیری



فهرست

3.....	پیشگفتار
4.....	شرح کلی پروژه
6.....	چگونگی نصب و راه اندازی و نیاز های نرم افزاری و سخت افزاری
8.....	شرح بانک های به کار رفته در سیستم
12.....	مقدمه ی آموزشی
12.....	ساختمان داده ها و دستورات مانند زیربرنامه ها و توابع و ...
37.....	پیشنهاد و نتیجه گیری
40.....	منابع

اینترنت با هدف یادگیری

پیشگفتار

دوران ما را بدرستی عصر اطلاعات نامیده اند. در اندیشه ما، فناوری اطلاعات به مفهوم گردآوری، سازماندهی و پردازش داده های خام است بگونه ای که از آن معرفت جدید تولید شود.

ظهور شبکه های کامپیوتری در دهه هفتاد میلادی تمام اسباب و لوازم تولید و معرفل دانش را فراهم کرد و به آن شتابی انفجار گونه داد، چرا که خرد فردی به خرد جمعی تبدیل شد و محفل کوچک و خصوصی خردمندان به یک ضیافت بزرگ از نخبگان دهکده جهانی مبدل گشت. شاید به کار بردن واژه انقلاب اطلاعاتی برای این روند تولید دانش مناسب تر باشد.

شبکه کامپیوتری چیزی بیش از یک ابزار در خدمت تولید دانش نیست ولی همین ابزار کارهایی را که ذهن بشر به کندی انجام میدهد هزاران بار تسریع می کند، نتیجه مشخص است، تولید و نشر دانشی که دهها سال پژوهش، جمع آوری داده آزمایش و نتیجه گیری نیاز دارد در کوتاه ترین زمان ممکن محقق میشود. دسترسی سریع و آسان به اطلاعات، استفاده از قدرت پردازشی کامپیوترهای قوی و کارآمد ایجاد و به کار گیری تکنولوژی نوین ارتباطی برای تبادل اطلاعات و ایجاد نرم افزارهای هوشمند به شکوفایی تمام شاخه های علوم مختلف انجامیده است.

اعتقاد راسخ داریم که شبکه های کامپیوتری بیش از آنکه یک مقوله صرفا تکنولوژیک و یکی از شاخه های علوم کاربردی باشد یک پدیده است، پدیده ای که روش زندگی تعاملات اجتماعی و گرایشات انسانی را نیز دستخوش تحولات جدی خواهد کرد.

آنچه که بر دوش متخصصین فناوری اطلاعات سنگینی میکند شناخت دقیق این فناوری و بکارگیری صحیح و اصولی آن است، زیرا ملتی که در عصر جدید، خود مولد دانش نباشد مقلدی بیش نیست و باید سیطره و نفوذ صاحبان دانش را بر خود بپذیرد!

بی شک برای افرادی که در دنیای کامپیوتر زندگی می کنند مشکل است که دور از دنیای برنامه نویسی زندگی کنند.

برنامه نویس کسی است که دنیای واقعی را به شکل ملموس اما مجازی در دنیای کامپیوتری خود میسازد. اساس کار برنامه پیش روی شما؛ شبکه است. امروز داغ ترین بحث ها در کامپیوتر شبکه می باشد پس مطمئنا برنامه نویسی آن نیز دارای اهمیت می باشد.



شرح کلی پروژه

کار اصلی پروژه ما بدین شرح است :

مشتری یا کارمند شرکت سیستم معیوب خود را برای تعمیر به خدمات کامپیوتر می برد. سیستم مورد نظر را به مسئول تحویل داده و مسئول نیز آن را به همراه اطلاعات شخصی فردی که سیستم را تحویل داده در کامپیوتر خود ثبت می کند. سپس این کار ها را به کارآموزان، یا عناوین دیگر مانند کارکنان یا کارگران تحویل میدهد. به این علت نام این بسته را کارآموز گذاشته ام که مطابق با پروژه ای باشد که در مکان کارآموزی خود انجام می دادم.

به هر حال کارآموز کار را که از طریق شبکه برایش ارسال شده بود ، دریافت کرده و آن را انجام می دهد. سپس کارش را بر اساس اینکه انجام شده یا نه مشخص کرده و برای سرور یا مسئول می فرستد. عملیات ثبت کار مربوطه به صورت خودکار در کامپیوتر سرور انجام می شود.

به این نسخه امکانات دیگری نیز می توان اضافه کرد مانند مانیتور کردن کامپیوتر ها، میز کاری که از اهداف خود بنده نیز بود اما به علت کمبود وقت از انجام آن خودداری شد.

سرور امکانات دیگری هم دارد مانند ثبت کارآموز، ثبت نام کاربری ، ثبت در انبار و برداشت از آن ، ارسال پیغام برای تمامی کاربران متصل به وی ، صدور انواع نامه، تایپ آن و ذخیره آن به عنوان Template ، چاپ آن و همچنین انواع گزارشات ترکیبی که به مدد خداوند در ادامه به آنها می پردازم.

عنوان پروژه همان عنوان نرم افزار می باشد یعنی **کارآموز** که دارای دو نسخه ی سرور و کلاینت می باشد. این دو نسخه بر روی هر کامپیوتری که دارای حداقل سخت افزار و کارت گرافیک حداقل 16 بیتی باشد قابل اجرا می باشد.

نسخه سرور دارای قابلیت های زیادی است از قبیل ثبت و حذف و اضافه کردن های مختلف ، همچنین امکان چک کردن این که کدام کاربر به کامپیوتر ما متصل است می باشد. انواع گزارش گیری از جمله گزارش گیری های آنلاین، ترکیبی و خطی می باشد.

ساختار ساختمان داده های این برنامه معمولاً به صورت خطی می باشد. یعنی برای جستجو، در اکثر موارد از جستجوی خطی استفاده شده است ، زیرا تعداد رکورد ها زیاد نیست.

این دو نسخه برای شرکت هایی که پذیرای کارآموزند و کار آنها خدمات کامپیوتر می باشد مفید می باشد همچنین با کمی تغییر می توان از آن به عنوان یک نرم افزار کافی نت استفاده کرد . زیرا کاری که در این قبیل نرم افزار ها انجام می شود این است که سیستم کلاینت را مانیتور کرده و زمان مورد استفاده را به پول تبدیل کرده و در کامپیوتر طرف بنویسد .

همچنین با کمی تغییر می توان از آن به عنوان یک سیستم مانیتورینگ مانند سیستم هایی که در مدارس استفاده می شود ، استفاده کرد.

بانک اطلاعاتی مورد استفاده در این نرم افزار SQL server میباشد که این نسخه آن 2000 می باشد . زبان برنامه نویسی که در آن به کار رفته نیز Microsoft Visual Basic 6 می باشد.

بیشتر الگوریتم های به کار رفته در آن استفاده از توابع های کنترل WinSock می باشد. همچنین در جستجو ها از جستجوی خطی و همچنین کمی Index استفاده شده است.

این برنامه دارای توابعی برای درج و حذف و اضافه کردن و جستجو می باشد که بیشتر برای سه مورد اول از توابع تعبیه شده در خود ویژوال بیسیک استفاده شده است.

لازم به ذکر است که کار اصلی این برنامه کار با بانک های اطلاعاتی نیست بلکه وظیفه اصلی آن دریافت سیستم از مشتری، ثبت سیستم مورد نظر، ارسال کار برای کارآموز می باشد به همین علت در بانک آن از روابط ناچیزی استفاده شده است. همچنین بعضی از روابط نیز درون خود پروژه و در عمق برنامه نویسی گنجانده شده است.

در این نسخه سعی شده با توجه به رابط کاربری دوستانه عمل شود حال این نظر با خودتان است .

سطوح کاربری موجود در این نرم افزار شامل سه دسته اصلی می باشد بدین شرح :

(1) مدیر

(2) مسئول

(3) کارآموز

اگر با این کاربر وارد شوید برای شما منوی جدیدی به عنوان مدیر باز می شود که می توان در آن مدیر یا مسئولان جدیدی ایجاد کرده اما مسئول هرگز نمی تواند یک مدیر ایجاد کند، کارآموز نیز حق استفاده از سیستم مسئول و مدیر را ندارد.

کارآموز به کمک شبکه به سیستم Log In می کند و به کمک بانکی که در سیستم سرور است به سیستم خود وارد می شود. اما کارهای تحویلی، ارسالی و ... را در سیستم خود ذخیره می کند . راجع به محدوده دسترسی نیز فقط مدیر حق تغییر آن را دارد.

برای تقریباً تمامی امکانات سرور، یک سطح دسترسی قرار داده شده است که می توان این سطح را تغییر داد.

از دیگر امکانات نسخه سرور می توان نامه پایان دوره کارآموزی را نام برد، که به ما امکان می دهد به صورت تمام و کمال یک کارآموز را نمره دهی کرده و نامه آن را چاپ کرده ، به وی داد.

امکانات دیگری نیز موجود است که به طور کامل در آینده به آنها می پردازم.

چگونگی نصب و راه اندازی و نیاز های نرم افزاری و سخت افزاری

برای کار با این نرم افزار قبل از هر چیز نیاز به کارت شبکه به تعداد کاربران می باشد، در صورتی که تعداد آنها از یک عدد متجاوز است باید از یک سویچ برای ارتباط آنها با سرور استفاده کرد. به صورت کلی نیاز های سخت افزاری به صورت زیر است :

- یک عدد کامپیوتر حداقل PIII برای نسخه سرور
- حداقل یک عدد کامپیوتر حداقل PIII و حداکثر در این نسخه 9 عدد
- به ازای هر کامپیوتر یک کارت شبکه و کابل های مورد نیاز

نکته دیگر این که سیستم های کلاینت باید به سرور متصل شوند اما سرور نیازی به این کار نیست. برای نصب نسخه سرور به صورت زیر عمل کنید :

فایلی به نام Setup.exe را از سی دی موجود در پوشه Server اجرا کنید. و مراحل زیر را طی کنید. این مراحل دقیقاً برای نصب نسخه کلاینت نیز تجویز کی شود.

در صفحه ابتدایی کادر خوش آمد گویی را ملاحظه می شود، با زدن Ok به صفحه بعد روانه شوید.

در این صفحه، مسیر نصب را انتخاب کنید. مهم نیست که مسیر شما چیست، زیرا کارآموز این امکان را دارد که خود را با همه چیز Match کند..

با کلیک روی دکمه ای که رویش عکس یک سیستم کامپیوتری است به صفحه بعد بروید، در این قسمت باید نام پروفایلی که می خواهید در All programs ظاهر شود را بنویسید.. با زدن دکمه Continue به مرحله نصب وارد شوید. در این مرحله سیستم شروع به نصب کرده که باید تا تکمیل آن صبر کنید.. تمام! سیستم آماده پیکر بندی است به این صورت عمل کنید :

همانند همه ی نسخه های دیگری که با Visual basic 6 نوشته شده این نسخه نیز به همان نحو قابل راه اندازی است با این تفاوت کامپیوتر سرور باید دارای شماره IP به صورت 192.168.0.1 میباشد.

برای تعویض IP سرور از مسیر زیر استفاده کنید :

Start > Control Panel > Network connection> Local area network> Internet Protocol(TCP/IP) > Properties > Use the following IP Address

سپس در قسمت مربوطه وارد کنید : 192.168.0.1

نکته اینکه این آدرس وحی منزل نیست و می توان آن را تغییر داد به این صورت که در پوشه نرم افزار کلاینت یک فایل به نام IP&Port.dat موجود است که با راست کلیک بر روی آن و انتخاب گزینه Open with و سپس انتخاب گزینه Notepad می توان محتوای آن را خواند، در صورتی که IP

همین و بس

استفاده شخصی از این اثر ممنوع است- فقط نگاه کن ازش یک چیز یاد بگیر-



سیستم سرور را نمی توانید تغییر دهید، کافی است در قسمتی که نوشته است: **server Address:** آدرس سرور مورد نظر را وارد کنید. همچنین اگر دلتان نخواست از پرت در نظر گرفته شده استفاده کنید، می توانید در همین فایل آن **Port** را تغییر دهید به علاوه اینکه باید همان پرت را در نسخه سرور نیز قرار دهید مگر نه سیستم کار نمی کند.

از امکانات نرم افزاری آن نصب بودن نرم افزار **visual basic Runtime** بر روی سیستم که اگر خود **VB6** باشد بهتر است. همچنین باید نرم افزار **SQLServer** نیز در سیستم نصب باشد. در این صورت باید بانک های مورد نظر را در سیستم های مورد نظر **Restore** کرد. نحوه ی **Restore** کردن آن بدین شرح است:

ابتدا **Enterprise Manager** را باز کرده و مسیر زیر را دنبال کنید:

Microsoft SQL server > SQL server group > Local(Windows NT)> Databases

سپس روی آن کلیک راست کرده و از منوی ظاهر شده گزینه **All Task** و سپس گزینه **Restore Data base** را انتخاب کنید. مسیر را به کمک دکمه رادیویی **From Device** انتخاب کرده و در نهایت دکمه **OK** را کلیک کنید.

مطلب دیگر آنکه برای هر کاربر کارآموز باید بانکی به نام **DBforKaramuz** را **Restore** کرد و برای سرور نیز بانک **Karamuz** را به طریقی که گفته شد.

پس از انجام کارهای فوق سیستم به صورت **Plug And Play** پیکربندی می شود و با باز شدن اولین صفحه سرور، آماده ی دریافت اتصال از طرف کارآموزان دیگر می شود.



شرح بانک های به کار رفته در سیستم

این سیستم دارای چندین بانک بوده اما به علت اینکه رابطه در این سیستم زیاد مهم نیست از جداول آن بیشتر به عنوان جانگهدار استفاده می شود..
جداول بانک در نسخه سرور با بانکی به نام Karamuz از این قرار اند :

Column Name	Data Type	Length	Allow Nulls
ID_ROW	int	4	
inputdate	nvarchar	50	
Shomare_amval	nvarchar	50	
shomare_sabt	nvarchar	50	
Irad	nvarchar	100	
name	nvarchar	50	
mobile	nvarchar	20	✓
innerTell	nvarchar	30	✓

(1) مشتری که شامل فیلد های زیر است :

- ID_ROW : که به منزله شماره سیستمی است که مشتری تحویل میدهد.
- Input Date : تاریخی که سیستم توسط مشتری تحویل داده می شود.
- Shomare_amval و shomare_sabt که معمولاً پشت سیستم درج می شود.
- Irad : که ایراد سیستمی است که آورده شده است.
- Name : نام مشتری یا کارمندی که سیستم وی نیازمند تعمیر است.
- Mobile : شماره موبایل
- Innertell : که تلفن داخل شرکتی را شامل می شود.

(2) جدول کارآموز که به صورت زیر است :

Column Name	Data Type	Length	Allow Nulls
Name	nvarchar	50	✓
Lname	nvarchar	50	
Branch	nvarchar	50	
Specially	nvarchar	50	✓
Tell	nvarchar	30	✓
description	nvarchar	200	✓
ID	nvarchar	50	
startDate	nvarchar	50	
worklenght	smallint	2	

- Name : نام کارآموز را شامل می شود. از نوع Nvarchar گرفته شده است.

همین و بس

استفاده شخصی از این اثر ممنوع است- فقط نگاه کن ازش یک چیز یاد بگیر-

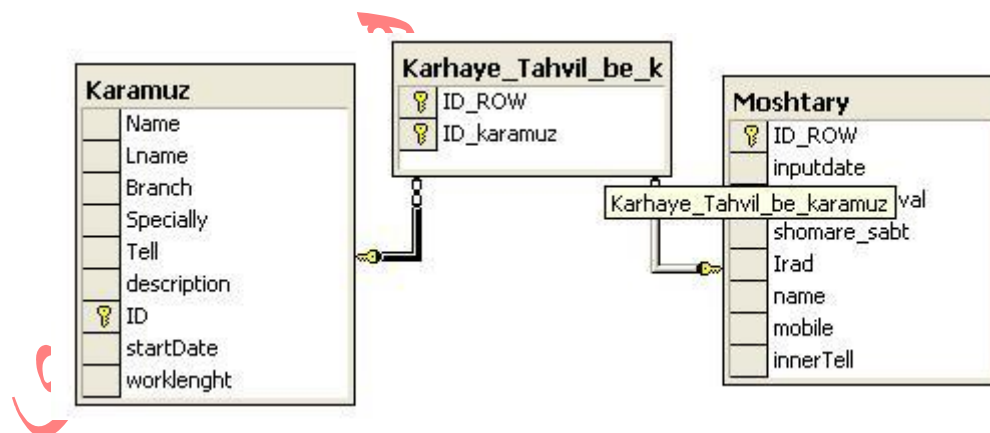
10

- Lname : نام خانوادگی کارآموز را شامل می شود. ، از نوع Nvarchar گرفته شده است.
- Branch : مقطع تحصیلی کارآموز را شامل می شود مثلاً فوق دیپلم ، از نوع Nvarchar گرفته شده است.
- Specially : که تخصص کارآموز را شامل می شود مثلاً هوش مصنوعی و ... ، از نوع Nvarchar گرفته شده است.
- Tell : به تلفن وی اشاره می کند. ، از نوع Nvarchar گرفته شده است.
- Description : اگر نیاز باشد توضیحاتی در مورد کارآموز را شامل می شود.
- ID : کد کارآموز را شامل می شود.
- Startdate : که تاریخ شروع کارآموزی را شامل می شود.
- WorkLenght : به مدت کارآموزی اشاره می کند.

(3) جدول میانی دو جدول بالا :

Karhayeh_Tahvil_be_karamuz			
Column Name	Data Type	Length	
ID_ROW	int	4	
ID_karamuz	nvarchar	50	

و همچنین دیاگرام آنها به صورت زیر است :



ارتباط این دو به صورت چند به چند است.

به علت زیاد بودن Table ها فقط به ذکر آنها به همراه تصویر اکتفا می کنم :

(4) ثبت جنس در انبار :

همین و بس

استفاده شخصی از این اثر ممنوع است- فقط نگاه کن ارزش یک چیز یاد بگیر-

Sabte_jens_dar_anbar				
	Column Name	Data Type	Length	Allow Nulls
PK	KalaCode	int	4	
	name	nvarchar	50	✓
	tedad	int	4	✓
	[date]	nvarchar	50	✓
	masulesabt	nvarchar	50	

به کمک این جدول اطلاعات موادی که خریداری می شود را ذخیره می کنند.

(5) درخواست از انبار

(6) کارهای انجام شده توسط کارآموز که در جدولی با عنوان Karhaye_anjam_shode ذخیره شده است.

(7) از جداول مهم و با ارزش در این بسته نرم افزاری جدول Login کردن کارآموز است که هم کارآموز؛ هم مسئول و هم رئیس به کمک آن به ان Login می کنند.

تصویر آن به صورت زیر است:

KaramuzLogin				
	Column Name	Data Type	Length	Allow Nulls
PK	Username	nvarchar	50	☐
	PassWord	nvarchar	50	✓
	Number	smallint	2	
	KARACODE	nvarchar	1	✓
	REGinsys	nvarchar	1	✓
	REGnewkara	nvarchar	1	✓
	REGinivent	nvarchar	1	✓
	REGdeliver	nvarchar	1	✓
	LETintroduction	nvarchar	1	✓
	LETrestaurant	nvarchar	1	✓
	LETprogresswork	nvarchar	1	✓
	SHOdelievtokara	nvarchar	1	✓
	SHOfinishwork	nvarchar	1	✓
	REPkara	nvarchar	1	✓
	REPalIkara	nvarchar	1	✓
	REPALlrecfrominv	nvarchar	1	✓
	REPrecfrominv	nvarchar	1	✓
	REPrecgininv	nvarchar	1	✓
	HLP	nvarchar	1	✓
	Manager	nvarchar	1	✓

فیلد اول و دوم که نیاز به توضیح ندارد.

فیلد سوم به سه قسم تقسیم می شود که توسط آن برنامه تشخیص می دهد کاربر چه نوع کاربری است :

1: کارآموز

2: مسئول

3: مدیر

اکثر فیلدهای دیگر نیز به منظور محدودیت دسترسی قرار گرفته شده اند. که صفر به منزله ی عدم دسترسی و یک به منزله ی دسترسی می باشد.

7) گزارش پایانی کارآموز که همان گزارش پیشرفت کاری و به عنوان جا نگه دار موقت اطلاعات می باشد.

8) کارهای تحویلی به کارآموز

9) زمانبندی یا Scheduling که برای زمانبندی به کار میرود.

10 تا 15) جانگهدارهایی به عنوان Temporary که شامل چندین جانگهدار است.

همچنین این بانک شامل چندین View است.

در این سیستم سعی شده تا به جای View از امکان خود VB به عنوان Record Source استفاده شود.

اینترنت با هدف یادگیری

مقدمه آموزشی

کنترل WinSock نسبت به تمام کنترل‌های اینترنت در سطح پایبندی قرار دارد. این کنترل امکان ایجاد سرویس‌های شبکه ای مبتنی بر پروتکل‌های TCP و UDP را مهیا می‌کند. عبارت دیگر توسط این کنترل می‌توان برنامه‌های کاربردی Client/Server (سرویس گیرنده / سرویس دهنده) ایجاد و با استفاده از پروتکل TCP و یا UDP بین آنها ارتباط برقرار نمود.

با تنظیم خصوصیات و فراخوانی متدهای این کنترل می‌توانید به راحتی به یک کامپیوتر راه دور متصل شوید و داده‌ها را در هر دو جهت جابجا نمائید. نمونه کاربرهایی که می‌توان با این کنترل ایجاد نمود: Client-server Chat, Mail client, Mail server, Server Proxy, Network Port Scanner, Game, پیاده سازی الگوریتم‌های موازی و ...

مبانی TCP :

پروتکل کنترل اینترنت (Transfer Control Protocol) اجازه می‌دهد یک اتصال (Connection) را از طریق سوکت (socket) به یک کامپیوتر راه دور (Remote Computer) ساخته و استفاده کنید. با استفاده از این اتصال، هر دو کامپیوتر می‌توانند داده‌ها را بین خودشان انتقال دهند. برقراری ارتباط از طریق TCP همانند صحبت کردن با تلفن است که باید حتماً اتصالی بین دو کامپیوتر صورت گیرد تا بتوانند با هم ارتباط برقرار کنند.

اگر یک برنامه Client می‌سازید بایستی بدانید که نام یا آدرس IP کامپیوتر Server چیست (Remote Host IP) و همچنین از طریق چه پورتی می‌توانید به آن متصل شوید (Remote Port). حال بایستی به آن پورت Connect کنید.

همچنین اگر یک برنامه Server می‌سازید بایستی پورتی را که روی آن به درخواستها گوش می‌دهید مشخص کنید (LocalPort) و سپس به پورت گوش دهید (Listen). زمانیکه یک کامپیوتر Client تقاضای یک اتصال را می‌دهد Server این درخواست را Accept می‌کند.

زمانیکه یک اتصال ساخته می‌شود، هر دو کامپیوتر می‌توانند داده را فرستاده و دریافت کنند.

مبانی UDP :

پروتکل دیتاگرام کاربر (User Datagram Protocol) پروتکلی بدون اتصال (Connectionless) است. برخلاف TCP، کامپیوترها نیاز به برپا کردن یک اتصال ندارند بنابراین یک برنامه می‌تواند یک client و یا یک server باشد. برقراری ارتباط در UDP شبیه ارسال نامه از طریق پست است.

برای انتقال داده توسط UDP ابتدا باید Local Port کامپیوتر Client تنظیم گردد . کامپیوتر Server تنها بایستی RemoteHost را برابر آدرس کامپیوتر Client قرار دهد و همچنین Remote Port را همان Local Port کامپیوتر Client قرار دهد . سپس دو کامپیوتر می توانند داده ها را بین خود جابجا کنند .

استفاده از کنترل WinSock :

1 - انتخاب پروتکل: در زمان استفاده از کنترل WinSock اولین کاری که باید انجام دهید انتخاب یکی از پروتکل های TCP یا UDP است . طبیعت برنامه ای که شما می سازید نوع پروتکلی را که باید استفاده کنید مشخص می کند . چند سوال زیر به شما کمک می کند که پروتکل مورد نیازتان را انتخاب کنید :
 - آیا برنامه شما در زمانی که داده فرستاده می شود یا دریافت می شود نیاز به اطلاعاتی از طرف Server یا Client دارد ؟ اگر چنین است بایستی یک اتصال TCP قبل از ارسال یا دریافت داده ایجاد شود .
 - آیا داده بسیار بزرگ است (مثل تصویر یا فایل های صوتی) ؟ زمانی که یک اتصال TCP ساخته می شود پروتکل TCP اتصال را باقی نگه می دارد و درستی ارسال داده تضمین شده است . این اتصال در هر حال به منابع محاسباتی بیشتری نیاز دارد و بنابراین پرهزینه تر است .
 - آیا داده متناوب ارسال می شود یا در یک نشست (Session) ارسال خواهد شد ؟ برای مثال اگر شما یک برنامه می سازید که کامپیترهای مشخصی را در یک زمان خاص از انجام شدن عملیاتی مطلع می کند پروتکل UDP مناسب تر است . پروتکل UDP همچنین برای ارسال مقادیر کوچک داده ای مناسب تر می باشد .

2 - تنظیم پروتکل : برای تنظیم پروتکلی که می خواهید در برنامه تان از آن استفاده کنید در زمان طراحی برنامه خاصیت Protocol کنترل WinSock را برابر sckTCPProtocol و یا sckUDPProtocol قرار دهید . همچنین می توانید پروتکل خود را توسط کد زیر تنظیم کنید :

```
WinSock.Protocol=sckTCPProtocol
```

3 - مشخص کردن نام کامپیوتان : برای اتصال به کامپیوتر راه دور بایستی آدرس IP و یا نام کامپیوتر را بدانید

نام کامپیوتر در Panel/Network/Identification Control موجود است . در صورتیکه می خواهید دو برنامه Client و Server خود را روی یک کامپیوتر تست کنید از آدرس IP 127.0.0.1 برای هر دو استفاده کنید اما اگر دو برنامه را روی دو کامپیوتر مجزا در شبکه قرار داده اید با اجرای دستور ipconfig در DOS Prompt می توانید آدرس IP کامپیوتر ها را بدست آورید .

4 - ایجاد اتصال TCP : در زمان ساخت برنامه ای که از پروتکل TCP استفاده می کند ابتدا باید تصمیم بگیرید که این برنامه Client است یا Server . برای ساخت یک برنامه Server بایستی روی یک پورت خاص Listen کنید . زمانی که Client تقاضای یک اتصال را می دهد ، برنامه Server می



تواند آنرا Accept کند و بنابراین اتصال کامل شده است . حال Client و Server می توانند با هم ارتباط داشته باشند .

مراحل زیر ساخت یک سرور چت ساده بر مبنای TCP را نشان می دهد :

- از منوی Project گزینه Components را انتخاب کنید و در لیست Component ها مورد Microsoft WinSock 6.0 را انتخاب کنید .
- یک کنترل WinSock در فرم خود قرار دهید و نام آنرا tcpserver بگذارید
- دو textbox با نامهای txtSendData و txtReceiveData و نیز یک دکمه در فرم قرار دهید .
- کد زیر را در رویداد Form Load بنویسید :

```
Tcpserver.LocalPort=1000
tcpserver.Listen
```

- زمانی که درخواستی از طرف Client می آید رویداد Connection Request اجرا می شود . در این رویداد ابتدا باید چک کنید که حالت کنترل بسته باشد . اگر چنین نیست اتصال را قبل از پذیرفتن اتصال جدید ببندید . سپس تقاضا را بر اساس پارامتر requestID می پذیریم :

```
Private Sub Winsock1_ConnectionRequest(ByVal requestID
As Long)
    If Tcpserver.State <> sckClosed Then
        Tcpserver.Close
        Tcpserver.Accept requestID
    End If
End Sub
```

- حال اتصال بین Client و Server برقرار شده است . کد زیر را برای event مربوط به کلیک دکمه Send بنویسید :

```
TxtSendData.Text Tcpserver.SendData
```

- اگر داده ای از طرف Client بیاید رویداد Data Arrival اجرا می شود . کد زیر را برای این رویداد بنویسید :

```
Private Sub Winsock1_DataArrival(ByVal bytesTotal As
Long)
    Dim strData As String
    Tcpserver.GetData strData
    txtReceiveData.Text = strData
End Sub
```

- کد زیر را برای رویداد Form_Unload بنویسید :

```
Tcpserver. Close
```

مراحل ساخت یک TCP Client بصورت زیر است :

- یک کنترل WinSock در فرم قرار دهید و نام آنرا tcpclient بگذارید .
- دو textbox با نامهای txtsend و txtreceive و نیز یک دکمه با نام send در فرم قرار دهید .
- یک دکمه با نام connect در فرم قرار دهید .
- کد زیر را برای متد Form Load بنویسید :

```
Tcpclient.RemoteHost = "your_server_name Or server_IP"
Tcpclient.RemotePort = your port number
```

- کد زیر را برای رویداد کلیک شدن دکمه connect بنویسید :

```
Tcpclient.Connect
```

- کد زیر را برای رویداد کلیک شدن دکمه send بنویسید :

```
txtsend. Text tctclient.SendData
```

- کد زیر را برای رویداد Data Arrival بنویسید :

```
Private Sub Winsock1_DataArrival(ByVal bytesTotal As
Long)
Dim strData As String
tcpclient.GetData strData
Txtreceive. Text = strData
End Sub
```

- کد زیر را برای رویداد Form_Unload بنویسید :

```
Tcpclient.Close
```

کدهای فوق یک سیستم Client-Server ساده را نشان می دهد . فایل exe هر دو برنامه را بسازید و آنها را اجرا کنید تا بتوانید سیستم خود را تست کنید .

11

5- پذیرفتن بیش از یک تقاضای اتصال : Server ای که در بالا ساخته شد تنها می تواند تقاضای یک اتصال را بپذیرد . با استفاده از ایجاد یک آرایه از کنترل WinSock می توان چندین تقاضای اتصال را پذیرفت . برای اینکار کافی است یک کپی (instance) از کنترل بسازیم (با تنظیم خاصیت Index) و متد Accept را برای instance جدید بکار ببریم . فرض کنید یک کنترل WinSock با نام sckServer در فرم داریم که خاصیت Index آنرا صفر قرار داده ایم . همچنین یک متغیر intMax از نوع Long تعریف می کنیم که تعداد اتصالات همزمان به Server را نگه می دارد . در event مربوط به Form_Load کد زیر را بنویسید :

```
intMax=0
sckServer(0).LocalPort=1000
sckServer(0).Listen
```

هر بار که تقاضای یک اتصال می رسد کد ابتدا تست می کند که مقدار Index چقدر است . اگر مقدار Index صفر باشد متغیر intMax یکی افزایش می یابد و از intMax برای ساخت یک instance جدید از کنترل استفاده می شود . حال از این instance برای پذیرفتن تقاضای اتصال استفاده می گردد . برای اینکار کد زیر را برای رویداد Connection Request بنویسید :

```
Private Sub Winsock1_ConnectionRequest(ByVal requestID
As Long)
If Index = 0 Then
intmax = intmax + 1
Load sckServer(intmax)
sckServer (intmax).LocalPort = 0
sckServer(Index).Accept requestID
End If
End Sub
```

6- ایجاد اتصال UDP : ساخت یک برنامه UDP ساده تر از برنامه های TCP است زیرا پروتکل UDP به اتصال نیاز ندارد . در برنامه TCP بالا یک کنترل WinSock بایستی حتماً Listen می کرد و یک کنترل دیگر یک اتصال را توسط متد Connect ایجاد نمود . در عوض پروتکل UDP نیازی به اتصال ندارد . برای ارسال داده بین دو کنترل WinSock سه مرحله بایستی انجام شود :

- پارامتر RemoteHost برابر نام کامپیوتر مقابل است .

- پارامتر RemotePort برابر پارامتر LocalPort کامپیوتر مقابل

- استفاده از متد Bind برای مشخص کردن LocalPort

چون هر دو کامپیوتر از نظر ارتباط مساوی هستند ، این نوع برنامه ها را Peer-to-Peer گویند . برای نمونه از کد زیر برای ساخت یک برنامه chat استفاده می کنیم :

همین و بس

استفاده شخصی از این اثر ممنوع است- فقط نگاه کن ارزش یک چیز یاد بگیر-

100

- یک کنترل WinSock در فرم قرار دهید و نام آنرا UDPpeerA بگذارید .
- خاصیت Protocol آنرا UDPProtocol قرار دهید .
- دو textbox با نامهای txtsend و txtreceive و نیز یک دکمه در فرم قرار دهید .
- کد زیر را برای متد Form Load بنویسید :

```
UDPpeerA.RemoteHost="NameOfPeerB"
UDPpeerA.RemotePort=1001
UDPpeerA.Bind 1002
```

- کد زیر را برای event مربوط به کلیک دکمه بنویسید :

```
txtsend.text UDPpeerA.SendData
```

- کد زیر را برای رویداد Data Arrival بنویسید :

```
String Dim strData as
UDPpeerA.GetData strData
txtreceive.Text=strData
```

برای ساخت UDPpeerB مشابه مراحل بالا عمل کنید فقط خاصیت RemoteHost آنرا نام کامپیوتر PeerA و خاصیت RemotePort آنرا 1002 و خاصیت Bind آنرا 1001 قرار دهید .

بررسی خواص کنترل WinSock :

ByteReceived : مقدار داده دریافت شده (موجود در بافر receive) را نشان می دهد . توسط متد GetData می توان این داده را دریافت نمود .

LocalHostName : نام ماشین محلی را نشان می دهد . این پارامتر فقط خواندنی است .

LocalIP : آدرس IP ماشین محلی را بصورت یک string برمی گرداند . این پارامتر فقط خواندنی است .

LocalPort : برای خواندن و یا تنظیم شماره پورت محلی بکار می رود .

Protocol : برای خواندن و یا تنظیم پروتوکل مورد استفاده توسط کنترل WinSock بکار می رود .

استفاده شخصی از این اثر ممنوع است- فقط نگاه کن ازش یک چیز یاد بگیر-

RemoteHost : برای خواندن و یا تنظیم نام یا آدرس IP ماشین راه دور بکار می رود .

RemoteHostIP : آدرس IP ماشین راه دور را برمی گرداند :

1- برای برنامه های Client بعد از زمانیکه یک اتصال توسط متد Connect پذیرفته شد ، این خاصیت حاوی آدرس IP ماشین راه دور است .

2 - برای برنامه Server ، بعد از آمدن یک Request Connection این خاصیت شامل آدرس IP ماشین راه دور است .

3 - در زمان استفاده از پروتکل UDP بعد از اینکه رویداد Data Arrival رخ داد این خاصیت حاوی آدرس IP ماشینی است که داده را فرستاده .

RemotePort : برای خواندن و یا تنظیم شماره پورت ماشین راه دوری که می خواهید به آن متصل شوید بکار می رود .

SocketHandle: مقداری را برمی گرداند که مرتبط با سوکتی است که کنترل WinSock را مدیریت می کند و برای ارتباط با لایه WinSock بکار می رود . این پارامتر فقط خواندنی است و تنها برای ارسال به API های WinSock طراحی شده است .

State : وضعیت کنترل WinSock را نشان می دهد . وضعیتهای ممکن برای State عبارتند از :

1 - sckClosed : اتصال بسته است .

2 - sckOpen : اتصال باز است .

3 - sckListening : حالت گوش دادن به پورت

4 - sckConnectionPending : معلق شدن اتصال

5 - sckResolvingHost : تصمیم گیری در مورد میزبان

6 - sckHostResolved : در مورد میزبان تصمیم گیری شد .

7 - sckConnecting : حالت برقراری ارتباط

8 - sckConnected : ارتباط برقرار شد .

استفاده شخصی از این اثر ممنوع است- فقط نگاه کن ارزش یک چیز یاد بگیر-

9 - sckClosing : حالت قطع اتصال

10 - sckError : حالت خطا

بررسی متدهای کنترل WinSock :

متد Accept : تنها برای برنامه های TCP Server بکار می رود . این متد برای پذیرفتن یک اتصال در زمان مدیریت رویداد ConnectionRequest استفاده می شود .

متد Bind : این پارامتر LocalPort و LocalIP یک اتصال را مشخص می کند .

متد Close : برای بستن یک اتصال TCP و یا بستن یک listening socket بکار می رود .

متد GetData : بلوک جاری داده دریافت شده را گرفته و آنرا در متغیری از نوع Variant ذخیره می کند . شکل کلی این متد بصورت زیر است :

Winsock1.GetData (Data,[Type],[MaxLen])

که data داده دریافتی است . اگر داده کافی موجود نباشد data برابر empty خواهد بود . type نوع داده دریافتی است که می تواند مقادیر زیر باشد :

vbByte - vbInteger - vbLong - vbSingle - vbDouble - vbDate -

vbError - vbString - vbArray+vbByte - vbBoolean

maxlen حداکثر سائز را در زمان دریافت یک byte Array و یا یک string مشخص می کند . متد Getdata در رویداد Data Arrival استفاده می شود که این رویداد یک پارامتر با نام Total Bytes دارد . اگر maxlen ای که شما تعیین کرده اید کمتر از Total Bytes باشد پیغام هشدار شماره 10040 دریافت می کنید بدین معنی که بایتهای باقیمانده گم خواهند شد . **متد Listen :** یک سوکت می سازد و آنرا در حالت Listen قرار می دهد . این متد تنها در اتصالات TCP بکار میرود .

متد PeekData : مشابه GetData است با این تفاوت که داده را از صف ورودی حذف نمی کند . این متد تنها برای اتصالات TCP بکار می رود .

متد SendData : برای ارسال داده به کامپیوتر راه دور بکار می رود .

استفاده شخصی از این اثر ممنوع است- فقط نگاه کن ارزش یک چیز یاد بگیر-

بررسی event های کنترل WinSock :

رویداد Close : زمانی رخ می دهد که کامپیوتر راه دور اتصال را ببندد .

رویداد Connect : بعد از اینکه یک اتصال به Server ایجاد شد روی می دهد . شکل کلی آن بصورت زیر است :

```
Private Sub WinSock_Connect(ErrorOccurred As Boolean)
```

که پارامتر ErrorOccurred دو مقدار دارد : اگر True باشد یعنی اتصال Fail شده است و اگر False باشد یعنی اتصال با موفقیت انجام شده است .

با رویداد Connect می توانید error هایی که در زمان فرایند باز کردن اتصال برگردانده شده را چک کنید .

رویداد Connection Request : زمانی رخ می دهد که یک کامپیوتر راه دور تقاضای یک اتصال را بدهد . این رویداد فقط برای برنامه های TCP Server بکار می رود .

رویداد Data Arrival : زمانی رخ می دهد که داده جدیدی بیاید .

رویداد Error : زمانی رخ می دهد که یک خطا در فرایند ارتباط رخ دهد (مثلاً Failed to Connect و یا Failed to Send) . شکل کلی آن بصورت زیر است :

```
Private Sub Winsock1_Error(ByVal Number As Integer,
Description As String, ByVal Scode As Long, ByVal
Source As String, ByVal HelpFile As String, ByVal
(HelpContext As Long, CancelDisplay As Boolean
```

number: شماره کد خطا است .

description: توضیحی در مورد خطا است .

source: توصیف منبع خطا

cancel display : مشخص می کند آیا پیغام خطای پیش فرض نشان داده شود یا نه

رویداد SendComplete : زمانی رخ می دهد که یک عمل Send تکمیل شده باشد .

همین و بس

استفاده شخصی از این اثر ممنوع است- فقط نگاه کن ارزش یک چیز یاد بگیر-

22

رویداد **SendProgress** : زمانی رخ می دهد که کنترل شروع به ارسال داده نماید . شکل کلی آن بصورت زیر است :

```
Private Sub Winsock1_SendProgress(ByVal bytesSent As
(Long, ByVal bytes Remaining As Long
```

که bytesSent تعداد بایتهای ارسال شده و bytesRemaining تعداد بایتهای باقیمانده است .

با این مقدمه و آشنایی با طرز کار کنترل Winsock به سراغ محیط نرم افزار می رویم و با آن آشنا می شویم.

اینترنت را در اینترنت با هدف یادگیری

ساختمان داده ها و دستورات مانند زیر برنامه ها و توابع و ...

ساختمان اکثر برنامه ها در بحث جستجو خطی است.

کار را از ابتدا شروع می کنیم از کامپیوتر سرور :

ابتدا باید به کامپیوتر LogIn کنیم پس باید به جدول مخصوص آن متصل شویم به این صورت :

باید از کنترل Adodc استفاده کرد که در Component ها موجود است :

```
Adodc1.connectionstring = Provider=SQLOLEDB.1;Integrated
Security=SSPI;Persist Security Info=False;Initial
Catalog=Karamuz;Data Source = "."
Adodc1.recordsource = karamuzlogin
```

البته می توان این کار را به صورت ویزاردی هم انجام داد که من تماما همگی را ویزاردی انجام داده ام.
این قطعه کد نمونه ای از یک جستجوی خطی برای پیدا کردن یک User نشان می دهد:

```
Adodc1.Recordset.MoveFirst
For i = 1 To Adodc1.Recordset.RecordCount
If trim(text1.text) =
adodc1.recordset.fields(0) and trim(text2.text) =
adodc1.recordset.fields(1) and adodc1.recordset.fields(2) =
2 then
Msgbox "سلام مسئول عزیز، به کارآموز خوش آمدید", vbcritical, ""
Else
End if
Next
```

با این قطعه کد Port ی که به کمک آن اجازه می دهیم که کلاینت ها با سرور ارتباط برقرار کنند را از فایل IP&PORT می خوانیم :

```
Open App.Path + "\IP&Port.dat" For Input As #1
For i = 1 To 7
Line Input #1, Var
If i = 5 Then
port = Var
End If
Next
```

در این سیستم از دو عدد کنترل Winsock که در بالا توضیح داده شد استفاده می شود.

```
Winsock1(0).LocalPort = port
Winsock1(0).Listen
```

دستور اول پرت کنترل Winsock1 را برابر با متغیر Port قرار می دهد که در بالا ذکر شده است .

24

و دستور دوم به این پرت گوش می دهد. به پراگماتیسم که جلوی Winsock نوشته شده است (0) به این منظور است که هر تعداد کارآموزی که به سیستم متصل شوند از یکی از این عدد ها دریافت کرده و با آن کار می کنند، به همین دلیل باید در ابتدا آرایه ای از نام ها درست کرد و نام هر کاربر و IP ئی را در آن قرار می دهد.

به این روش Index گویند.

به کد های زیر دقت کنید :

```
SB1.Panels(1).Text = Time
SB1.Panels(2).Text = FarDate1.today
```

به کمک کنترل Status Bar می توان هر وضعیتی مانند تاریخ، ساعت، نوع کامپیوتر و ... را در آن نمایش داد. مثلا دو کد بالا را نگاه کنید : به کمک متد Panel که یک متد Index ی است می توان انواع وضعیت ها را ثبت کرد تا کاربر با دیدن آن متوجه شود که برنامه در چه وضعیتی است.

به کمک تابع LoadKeyboardLayout می توان نوع زبان سیستم را تغییر داد که یک API است و باید آن را فراخوانی کرد. به منظور تبدیل کیبورد به فارسی از قطعه کد زیر استفاده کنید :

```
LoadKeyboardLayout "00000429", 1
```

که کد 429 برای فارسی و کد 409 برای انگلیسی می باشد.

حال کمی به کنترل Winsock می پردازم :

به قطعه کد روبرو دقت کنید :

این تکه کد ها در رویداد ConnectionRequest نوشته شده است. یعنی منتظر باش تا کسی یک درخواست دهد و این رویداد فراخوانی شود.

```
Private Sub Winsock1_ConnectionRequest(Index As Integer, ByVal requestID As Long)
Static i As Integer
If Index = 0 Then
i = i + 1
Load winsock1(i)
winsock1(i).LocalPort = 0
winsock1(i).Accept requestID
End If
End sub
```

ابتدا متغیر i را تعریف کرده از نوع static که با هر بار دستور دادن هر کاربر که می خواهیم به شما متصل شوم این متغیر یکی اضافه شده و به کمک خط 5 یک winsock جدید از آن می سازد. سپس در خط 7 می گوید که من منتظر دریافت از این کاربری هستم که ID آن را به کمک متغیر RequestID برایم فرستاده است.

25

نوشتن خط 6 الزامی است و با خط 3 می گوییم که منظور ما از Winsock ی که کار می کند همان Winsock(0) است نه کنترل دیگری.

به منظور دریافت اطلاعات از کامپیوتر کلاینت باید از رویداد Data Arrival به معنی ورود داده استفاده کرد

```
Private Sub Winsock1_DataArrival(Index As Integer, ByVal bytesTotal As Long)
```

پارامتر اول یا Index بیانگر این است که کدام کاربر برایمان اطلاعات ارسال کرده است و پارامتر دیگر مقدار داده ای که برایمان ارسال شده را نشان می دهد.

تذکر بسیار مهم :

بهتر است بگویید تجربه بسیار مهم :

اگر بخواهید یک سیستم خیلی بزرگتر از این به عنوان مثال یک اتوماسیون اداری بنویسید حتما باید از Select case استفاده کنید : فکر خوبی است نه ؟

درون رویداد DataArrival کد زیر را نوشته ایم :

```
Winsock1(Index).GetData str
```

به این معنی که داده ها از کامپیوتری که Index آن را متمایز می کند دریافت شده و در متغیری به نام str ذخیره می شود. حال اگر سیستم ما بزرگ باشد باید هر داده ای ارسال میشود برای هر بخش یک کد تعیین کنیم تا کارمان آسان شود مثلا :

```
For I = 1 to len(str)
    strRecieve = mid(str,I,1)
    If A = " " then
        MyStrRecieve = mid(str,1,i)
        Exit for
    End if
Next
Select Case MyStrRecieve
    Case "Request"
        .
        .
        Case "bakhsh1 Alizadeh"
        .
        Case "Delivery"
        .
        Case "Payam"
        .
        .
End select
```

در این صورت می توانید این کنترل ها را به طور کامل در دست بگیرید و حداکثر استفاده را از آن بکنید .

همین و بس

استفاده شخصی از این اثر ممنوع است- فقط نگاه کن ارزش یک چیز یاد بگیر-

29

در این برنامه من از کنترل Tray استفاده کرده ام که آن را از Internet دانلود کرده ام. از سایت www.barnamenevis.org دانلود کرده ام. به کمک آن می توان انواع بالن را در قسمت Tray سیستم خود نمایش دهید.

کد های روبرو برای راه انداختن یک بالن به کار می رود :

```
Tray1.Add
Tray1.balloonIcon = Error
Tray1.BalloonText = " کار آموز عزیز ، لطفا کار خود را انجام دهید "
Tray1.BalloonTitle = "اخطار"
Tray1.PopupBalloon
```

در صورتی که در وسط ارتباط و در حال کار به هر دلیلی ارتباط قطع شد باید از رویداد زیر استفاده کرد :

```
Private Sub Winsock1_Close(Index As integer) 1
IntegerfrmKrmConnection.Image2(Index - 1).Visible =
False
frmKrmConnection.Image1(Index - 1).Visible = True 2
Winsock1(Index).Close 3
Winsock1(Index).LocalPort = 0 4
Call Winsock1(Index).Listen 5
End Sub 6
```

اصل کد ها را خطوط 3 و 4 و 5 تشکیل می دهند در خط 3 ابتدا ارتباط را می بندیم، در خط 4 دوباره خط را آزاد کرده تا به کمک خط 5 به خط گوش کنیم.

در صورتی که به هر دلیلی خطا رخ داد باید از رویداد Winsock1_Error استفاده کرد

```
Private Sub Winsock1_Error(. . . .)
Dim result As Integer
result = MsgBox(Source & ":" & Description, vbOKOnly,
"tcp/ip error"
End Sub
```

وقتی هر کارآموزی به سیستم وارد می شود باید مانیتور آن را در فرمی که همه ی کارآموزان در آن مانیتور شده اند روشن کرد :

```
If Index = 0 Then .a
i = i + 1 2
Load FrmMainServer.Winsock1(i) 3
frmKrmConnection.Image1(i - 1).Visible = False 4
frmKrmConnection.Image2(i - 1).Visible = True 5
FrmMainServer.Winsock1(i).LocalPort = 0 6
FrmMainServer.Winsock1(i).Accept requestID 7
```

8. End If

27

مشاهده می کنید که در دو خط 4 و 5 این عمل انجام می شود و مانیتور وی روشن می شود.

این مانیتور چیزی نیست جز یک Picture Box و دو تا Label

برای اینکه Ip کارآموز را در لیبل مشاهده کنیم باید از دستور زیر استفاده کنیم :

```
frmKrmConnection.label(i - 1).caption =  
Winsock1(Index).RemoteHostIP
```

به منظور ذخیره کردن نام کلاینت در جعبه ی مانیتور باید به این صورت عملکرد که از طرف کلاینت نام text را که در آن کارآموز خودش را معرفی می کند و دکمه ی اتصال را می زند همان نام را برایمان ارسال میکند و ما به کمک WinSock آن را دریافت کرده و چاپ می کنیم

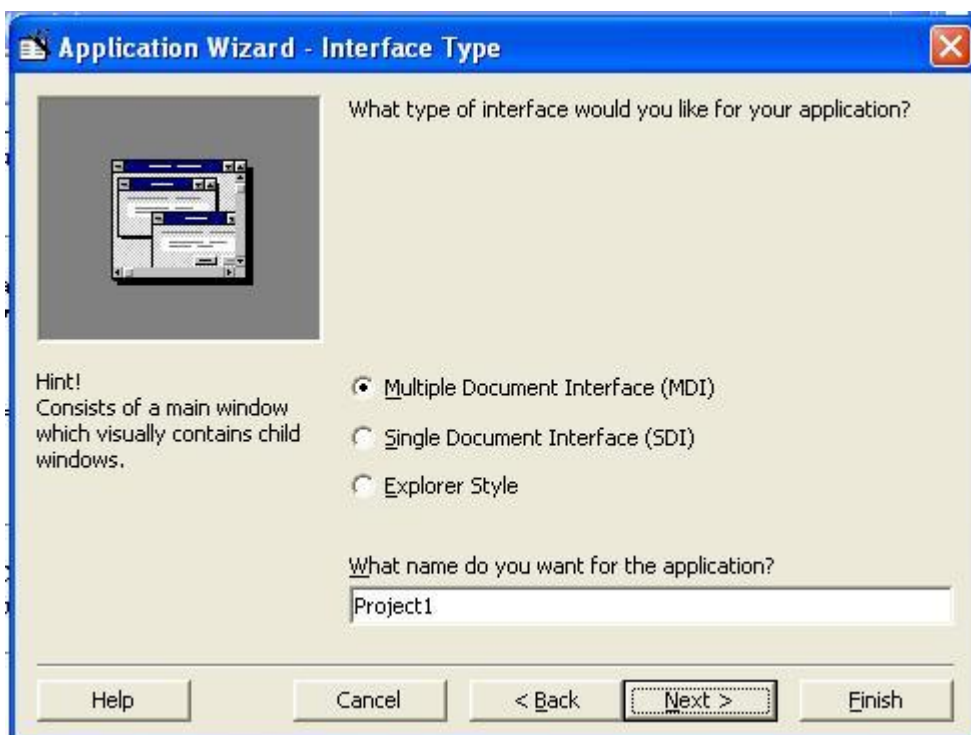
چت Online نیز در این سیستم تعبیه شده است که در بالا یک نمونه ساده آن توضیح داده شده است. به منظور ارسال یک پیام برای کارآموز خاص در ابتدا باید ایندکس مورد نظر را داشت ، پس باید ایندکس را با یک جستجو در آرایه ای که از ایندکس ها ایجاد کرده ایم و هر لحظه در حال تغییر است انجام دهیم. می توان کاری دیگر هم کرد که من کردم، یعنی بگوییم در جعبه ی مانیتورینگ چه کسی Online است و چک کنیم که کدام اسم با Label مورد نظر هم خوانی دارد پس از آن می توان از کد زیر استفاده کرد :

```
FrmMainServer.ws2(val(Your Index)).Send Data Text1.Text
```

به راحتی داده ها به سیستم مورد نظر ما ارسال می شود.

در این بسته برای تایپ نامه ها و چاپ آنها از یک فرم MDI(multiple Document Interface) استفاده شده است که آماده است به این منظور باید مراحل زیر را طی کرد :

Add-ins > Add-ins manager > VB6 application wizard > loaded/unloaded را کلیک می کنیم و سپس دوباره به منوی Add-ins رفته و Application-wizard را کلیک می کنیم. حال بقیه آن را به راحتی به صورت ویزاردی طی می کنیم.



به راحتی با کد های زیر می توان یک صفحه جدید در این صفحه Load کرد که من در اینجا یک template ی که از قبل آماده کردم را Load کردم تا از آن به عنوان نامه ی رستوران استفاده شود و هر جا که مایل بودند اضافه یا حذف کرد.

Load NewDoc

```
ActiveForm.rtfText.LoadFile App.Path & "\letter\" &
FrmMainServer.Text1.Text
```

خود این صفحه نیز پروژه ای است برای خودش که مایکروسافت زحمت کشیده و کار ما رو راحت کرده.

در قسمت گزارش پایانی کارآموز که فرمی است هم شامل نظرسنجی و هم نمایش نام و نام خانوادگی از کدی مانند زیر استفاده شده است :

```
Adodc1.Recordset.MoveFirst
For i = 1 To Adodc1.Recordset.RecordCount
If Trim(Adodc1.Recordset.Fields(1)) = Trim(Text1.Text)
Then
Text3.Text = Adodc1.Recordset.Fields(2)
Text5.Text = Adodc1.Recordset.Fields(7)
Text1.Text = Adodc1.Recordset.Fields(0) & " " &
Adodc1.Recordset.Fields(1)
Text2.SetFocus
Exit Sub
End If
Adodc1.Recordset.MoveNext
Next
```

همین و بس

استفاده شخصی از این اثر ممنوع است- فقط نگاه کن ارزش یک چیز یاد بگیر-

29

با این قطعه کد در صورتی که در قسمت نام خانوادگی و نام، نام خانوادگی را ذکر کنیم، خودش نام، و نام خانوادگی، و سایر مشخصات را از بانک خوانده و در کادرهای مربوطه نمایش می دهد.

به منظور ایجاد نتیجه فرم نظر سنجی یا نظر خواهی از کدهای زیر استفاده شده است، در این صورت به راحتی به کارآموز نمره داده می شود.

با دقت کامل نمایید

ضعیف (یک امتیاز)	متوسط (دو امتیاز)	خوب (سه امتیاز)	خیلی خوب (چهار امتیاز)	خصوصیات کارآموز:
☐	☐	☐	☐	استعداد و قدرت یادگیری:
☐	☐	☐	☐	طرز انجام کار و میزان علاقه:
☐	☐	☐	☐	وقت شناسی:
☐	☐	☐	☐	حس همکاری:
☐	☐	☐	☐	خود آموزی:
☐	☐	☐	☐	دیگر آموزی:
☐	☐	☐	☐	رفتار و حفظ شئون:

نمایش  از ۲۸ امتیاز

جمع امتیازات:

= 1 To
28

For i

```
If val(Option1(i).Caption) Mod 5 = 0 And Option1(i).Value =
True And Mid(Option1(i).Caption, 1, 1) = "V" Then Status =
Status + 4
```

```
ElseIf val(Option1(i).Caption) Mod 7 = 0 And
Option1(i).Value = True And Mid(Option1(i).Caption, 1, 1) =
"G" Then Status = Status + 3
```

```
ElseIf val(Option1(i).Caption) Mod 11 = 0 And
Option1(i).Value = True And Mid(Option1(i).Caption, 1, 1) =
"M" Then Status = Status + 2
```

```
ElseIf val(Option1(i).Caption) Mod 13 = 0 And
Option1(i).Value = True And Mid(Option1(i).Caption, 1, 1) =
"B" Then Status = Status + 1
```

```
End If
Next
```

```
Text9.Text = Status
```

همین و بس

استفاده شخصی از این اثر ممنوع است- فقط نگاه کن ارزش یک چیز یاد بگیر-



در این فرم به منظور نمایش رکوردی خاص باید اطلاعات را فیلتر کرد اما می توان فقط در همان لحظه اطلاعات را از یک بانک نیز خواند و همان موقع نمایش داد و همان موقع تمامی محتویات بانک را برای دفعات بعد پاک کرد اما قطعاً مورد اول بهینه تر و کاراتر است.

در قسمت نمایش و جستجو انواع جستجوی ترکیبی و آنلاین تعبیه شده است، برای نمونه چند مورد را ذکر می کنیم :

```
Private Sub Text1_Change ()
Adodc1.RecordSource = "SELECT Karamuz.Name + ' ' +
Karamuz.Lname, Moshtary.ID_ROW,
Moshtary.shomare_sabt, Moshtary.Irad, Moshtary.inputdate FROM
Karhayeh_Tahvil_be_karamuz INNER JOIN Karamuz ON
Karhayeh_Tahvil_be_karamuz.ID_karamuz =Karamuz.ID INNER JOIN
Moshtary ON Karhayeh_Tahvil_be_karamuz.ID_ROW = Moshtary.ID_ROW
'" & where Moshtary.ID_ROW LIKE '" & Text1.Text
Adodc1.Refresh
End Sub
```

چون از سه جدول باید اطلاعات را خارج کرد تا نام و نام خانوادگی، و شماره سیستم را خارج کرد باید از دستور Select به کمک Inner Join استفاده کرد .

همچنین به کمک دستور Like به راحتی می توان جستجوی آنلاین درست کرد به شرطی که دستورات را در رویداد Change مربوط به Text نوشت.

به منظور ایجاد جستجو های ترکیبی در یک مورد به جای یک Where در دستورات SQL از سه تا استفاده کرده ام و به متن بالا که برای Text1_change نوشته بودم اضافه کردم مثلاً به منظور فهرست گیری از اطلاعات سیستم ها از سه پارامتر شماره سیستم، شماره ثبت و کارآموز انجام دهنده استفاده شده است و از شرط زیر استفاده شده است :

```
WHERE (Moshtary.ID_ROW = '" & Text4.Text & "') AND
(Moshtary.shomare_sabt = N'" & Text5.Text & "') AND
& (Karamuz.Lname = N'" & Text6.Text & "'")
```

به دوتا N در دستورات نگاه کنید، کلید اصلی کار همین است، در مواقعی که دستورات به صورت رشته ای است باید از این روش استفاده کرد.

به یک نمونه دیگر توجه کنید :

```
Private Sub Command1_Click()
Adodc1.RecordSource = "SELECT ID, ID_ROW, Irad ,description ,
InputDate, OutputDate, ID_Karamuz FROM
dbo.Karhayeh_anjamshode where ID_Karamuz =' '" & Text6.Text &
'" and shomaresabt = '" & Text5.Text & "'" AND ID_ROW = '" &
Text4.Text
Adodc1.Refresh
End Sub
```

در این سیستم از نرم افزار Crystal Report استفاده شده است. نرم افزاری بسیار کاربردی و راحت به منظور راه انداختن این نرم افزار در وی بی باید ابتدا آن را در سیستم نصب کرد، سپس باید در کادر Reference آن را فراخوانی کرد ، به این منظور باید مسیر زیر را طی کرد :



Object menu > references > Crystal report 9 ActiveX Designer Design And Runtime Library

این چک باکس را کلیک کرده و سعی کنید هرچه با نام کریستال ریپورت نشان شده را انتخاب کرد و مسیر زیر را دنبال کنید :

Object Menu > Component > Crystal report viewer Control 9

سپس کنترلی که به جعبه ابزار اضافه شده را به صفحه اضافه می کنیم.
حال باید کد نویسی را آغاز کنیم :

```
Dim crApp As New CRAXDDRT.Application
Dim crRept As New CRAXDDRT.Report
Dim crDBTab As CRAXDDRT.DatabaseTable
```

کد ابتدایی برای استفاده از Application است و دومی برای کار با Report
حال برای گرفتن یک گزارش آسان و ساده باید ابتدا آن را توسط خود Crystal report ایجاد کرد و ذخیره کرده و در Vb از آن استفاده کرد.
حال برای ارتباط با آن باید از کد زیر استفاده کرد :

```
Set crRept = crApp.OpenReport(App.Path &
"\Reports\Sample_Report.Rpt")
```

در این صورت ما با یک مسیر که در آن یک فایل Report موجود است ارتباط برقرار کرده و آماده ی نمایش آن می شویم.
با استفاده از کد های زیر می توان صفحه را نمایش داد :

```
CrystalReport.ReportSource = crRept
CrystalReport.ViewReport
crRept.DiscardSavedData
```

آخرین خط کد مشخص می کند که پایگاه Refresh شود.
با این اطلاعات و اوصاف می توان گزارش های پیشرفته ی دیگری را نیز ایجاد کرد مثلا گزارش هایی که نیاز به فیلتر و محدود کردن دارند.
برای این کار باید از متد RecordSelectionFormula استفاده کرد ، به عنوان مثال کد زیر نام های خانوادگی که برابر با Text باشند را نمایش می دهد

32 crRept.RecordSelectionFormula = "{karamuz.Lname} = '" &
Text1.Text & "'"

از کدهای بسیار پر کاربرد اما کمی مشکل می توان به پیدا کردن یک بازه از تاریخ نام برد که من در Data report نتوانستم معادلش را پیدا کنم اما در Crystal Report معادلش به صورت زیر است.

crRept.RecordSelectionFormula = "{karamuz.StartDate} IN ('" & Trim (F1.Text) & "' TO '" & Trim(F2.Text) & "'"

به کمک این کد می توان گفت که از بازه ی F1.text تا بازه F2.text را نمایش بده.
نکته دیگری می ماند و آن این است که ممکن است بخواهیم اطلاعاتی را در گزارش نمایش دهیم ، مثلا تاریخ شمسی داریم و باید آن را در گزارش نمایش دهیم ، در این صورت از کد زیر استفاده می کنیم :

Crep.FormulaFields.GetItemByName("shamsi").Text = "'" &
F1.today & "'"

در قسمت ایجاد نام کاربری جدید باید ابتدا نامی را خواند، چک کرد که آیا این نام از قبل وارد شده یا خیر؛ سپس باید آن را به بانک به عنوان یکی از سه حالت مدیر، کارآموز، مسئول در بانک ذخیره کرد و از همه جالب تر اضافه کردن آن به کنترل ListView می باشد.

Adodc1.Recordset.AddNew .1
Adodc1.Recordset.Fields(0) = Text1.Text .2
Adodc1.Recordset.Fields(1) = text2.Text .3
If Combo1.ListIndex = 0 Then .4
UDT = 1 .5
ElseIf Combo1.ListIndex = 1 Then .6
UDT = 2 .7
ElseIf Combo1.ListIndex = 3 Then .8
UDT = 3 .9
End If .10
Adodc1.Recordset.Fields(2) = UDT .11

کد های نمایش داده شده در بالا مراحل اضافه کردن به بانک را نشان می دهد . سه خط اول که تابلو است، اما خطوط 4 تا 10 به ما نشان می دهد که هر کدام از سه حالت مدیر، مسئول و کارآموز را انتخاب کنیم یک شماره به ما اختصاص داده می شود. سپس شماره مورد نظر در بانک قرار می گیرد.

حال نوبت به تنظیمات **ListView** می باشد :

33

```
ListView1.View = lvwIcon
```

به کمک کد بالا می توان شکل و حالت نمایش کنترل لیست را به صورت آیکن بزرگ در آورد که در ویندوز نیز با آن سر کار داریم.

به دستور زیر دقت کنید :

```
For i = 1 To Adodc1.Recordset.RecordCount
If Adodc1.Recordset.Fields(2) = 1 Then
User = "karamuz"
user2 = "کارآموز"
ElseIf Adodc1.Recordset.Fields(2) = 2 Then
User = "masul"
user2 = "مسئول"
ElseIf Adodc1.Recordset.Fields(2) = 3 Then
User = "modir"
user2 = "مدیر"
End If
Set X = ListView1.ListItems.Add(, User & i,
Adodc1.Recordset.Fields(0) & vbCrLf & user2, 1, 1)
End if
next
```

به کمک دستور بالا یک گره یا **Node** جدید در **ListView** درج می کنیم.

User & i به ما می گوید که گره ی جدید ما دارای نام محتوای متغیر **User** + محتوای متغیر **i** می باشد.

همچنین کد **Adodc1.Recordset.Fields(0) & vbCrLf & user2** مشخص می کند که نام گره ی جدیدی که ساخته می شود برابر نام کاربری + یک خط پایین + نوع کاربری باشد.

از میان مجموعه کد ها یک نمونه انتخاب شده است :

```
If KeyAscii = 13 And ListView1.ListItems.Count <> 0 Then
Adodc1.Recordset.MoveFirst
Adodc1.Recordset.Move Index - 1
Adodc1.Recordset.Fields (1) = ""
Label19.Visible = False
Label13.Visible = True
Text8.Visible = False
Exit Sub
End If
```

به کمک این تکه کد می توان پسورد یا کلمه عبور کاربر مورد نظر را حذف کرد.

همین و بس

استفاده شخصی از این اثر ممنوع است- فقط نگاه کن ازش یک چیز یاد بگیر-

34

از قسمت های مهم دیگر در برنامه قسمت محدوده دسترسی می باشد :
 شکل زیر را ملاحظه کنید :

می بینید که هر قسمتی را می توان اجازه دسترسی داد و هر قسمتی را خیر.
 کد مربوط به این قسمت به صورت زیر می باشد :

```

If Combo1.Text = "" Then
MsgBox "لطفا یک کاربر از مسئول یا مدیر انتخاب کنید", vbcritical, ""
Exit Sub
Else
'//////////to search a user//////////
Adodc1.Recordset.MoveFirst
For i = 1 to Adodc1.Recordset.RecordCount
If Adodc1.Recordset.Fields (0) = Trim (Combo1.Text) Then
Adodc1.Recordset.Fields (4) = Check1.Value
Adodc1.Recordset.Fields (5) = Check3.Value
Adodc1.Recordset.Fields (6) = Check4.Value
Adodc1.Recordset.Fields (7) = Check5.Value
'
Adodc1.Recordset.Fields (8) = Check7.Value
Adodc1.Recordset.Fields (9) = Check8.Value
Adodc1.Recordset.Fields (10) = Check9.Value

```

```

        Adodc1.Recordset.Fields (11) = Check14.Value
        Adodc1.Recordset.Fields (12) = Check15.Value
    '
        //////////////////////////////////////
        Adodc1.Recordset.Fields (13) = Check11.Value
        Adodc1.Recordset.Fields (14) = Check17.Value
        Adodc1.Recordset.Fields (15) = Check12.Value
        Adodc1.Recordset.Fields (16) = Check18.Value
        Adodc1.Recordset.Fields (17) = Check19.Value
    '
        //////////////////////////////////////
        Adodc1.Recordset.Fields (18) = Check21.Value
        Adodc1.Recordset.Fields (19) = Check13.Value
        Adodc1.Recordset.Update
        MsgBox " عملیات با موفقیت پایان یافت ", vbinformation, ""
        Unload Me
    Exit For
End If
Adodc1.Recordset.MoveNext
Next
End If

```

در مورد کادر **About** هم می توان گفت که این کادر از کادر های آماده در **Visual Basic** می باشد که برای اضافه کردن آن در فرم باید روی قسمت **Project Explore** در مکان خالی آن راست کلیک کرده و از منوی ظاهر شده به ترتیب **ADD** و **Form** را انتخاب کنید سپس در کادر ظاهر شده گزینه **About Box** را انتخاب کرده و بر اساس آنچه تمایل دارید تغییر دهید.

❌ **حال کمی به کامپیوتر کلاینت می پردازیم:**

برای Load کردن اینکه کلاینت به سرور Login می کند از یک Component به نام VistaProgressBar استفاده کرده ام که در اینترنت پیدا می شود. برای زیبا جلوه دادن پروژه ، گفتم که همراه با یک Label هر چند صدم ثانیه یک بار به مقدارش اضافه کن و در نهایت در شماره 200 ارتباط را برقرار کن.

کد مورد نظر زیر علاوه بر توضیح بالا، Ip و Port را هم از یک فیل می خواند و به کمک آن به سرور متصل می شود.

```
If V1.Value = 200 Then
Ws3.Close
Winsock1.Close
Open App.Path + "\IP&Port.dat" For Input As #1
For i = 1 To 7
Line Input #1, Var
If i = 5 Then
IP = Var
ElseIf i = 7 Then
port = Var
End If
```

Next

Close #1

به منظور اینکه بدانیم آیا Ip وارد شده در فایل توسط کاربر ، معتبر یا Valid یا از نظر املائی صحیح است یا خیر از الگوریتم زیر استفاده شده است :

```

For k = 1 To Len(IP) .1
a = Mid(IP, k, 1) .2
If a = "." Then .1
dot = dot + 1 .2
End If .b
Next .3
If dot <> 3 Then .4
MsgBox "آدرس وارد شده معتبر نیست", vbcritical, "" .5
Exit Sub .i
End If .6

```

خطوط اول تا سوم چک می کند که در آی پی وارد شده چند تا نقطه وجود دارد و در خطوط بعدی مشخص می کند که اگر سه تا نبود پس این آدرس صحیح نیست .
در طرف کلاینت برخلاف سرور برای اتصال باید از کدی مانند کد زیر استفاده کرد:

Winsock1.Connect IP, port

در رویداد دریافت داده این کد نوشته شده است :

Case "Request"
frmChat.Show

بیان می کند که اگر داده ای که دریافت شد از کامپیوتر سرور، برابر با کد Request بود ، فرم چت روم را باز کن.

در صورتی که در ارتباطات خطایی رخ داد، باید در رویداد Error آن را یادداشت کرد بدین صورت :

```

Private Sub Ws3_Error(. . .)
Ws3.Close
Ws3.connect IP,Port
End Sub

```

که می گوید ارتباط را قطع کن و دوباره متصل شو.

برای دریافت یک کار از سرور چنین کدی نوشته شده است :

```

Private Sub Winsock1_DataArrival(ByVal bytesTotal As Long)
Static Counter As Integer
Dim str As String
Winsock1.GetData str
If Mid (str, 1, 1) = "*" Then
frmTree.Adodc1.Recordset.AddNew
frmTree.Adodc1.Recordset.Fields (0) =
frmTree.Adodc1.Recordset.RecordCount
frmTree.Adodc1.Recordset.Fields (1) = Mid (str, 3, 1)
frmTree.Adodc1.Recordset.Fields (2) = Mid (str, 4, Len (str))

```

همین و بس

استفاده شخصی از این اثر ممنوع است- فقط نگاه کن ازش یک چیز یاد بگیر-

37

```

frmTree.Adodc1.Recordset.Fields (3) = frmTree.FarDate1.today
frmTree.Adodc1.Recordset.Update
frmMainKaramuz.STBar.Panels (4).Text = "کار جدید" &
frmTree.Adodc1.Recordset.RecordCount
If Mid (str, 2, 1) = 1 Then
frmMainKaramuz.Tray1.BalloonIcon = Error
frmMainKaramuz.Tray1.BalloonTitle = "کار مهم جدید"
frmMainKaramuz.Tray1.BalloonText =
"کار جدید با شماره " & Mid(str, 3, 1) & "به شما سپرده شده است. برای کسب اطلاعات بیشتر <<سریعا >> به جعبه
کاری خود مراجعه کنید"
frmMainKaramuz.Tray1.PopupBalloon
Else
frmMainKaramuz.Tray1.BalloonTitle = "کار جدید"
frmMainKaramuz.Tray1.BalloonText =
"کار جدید با شماره " & Mid(str, 3, 1) & "به شما سپرده شده است. برای کسب اطلاعات بیشتر به جعبه کاری خود مراجعه
کنید"
frmMainKaramuz.Tray1.PopupBalloon
End If
Dim NewJob As Node
Set NewJob = frmTree.TreeView2.Nodes.Add("NewJob", tvwChild, "NewChild"
& frmTree.Adodc1.Recordset.RecordCount, "System Number : " & Mid(str, 3,
1) & " Date " & frmTree.FarDate1.today, 3)
NewJob.ExpandedImage = 3
NewJob.Expanded = True
"Winsock1.SendData "Delivery
Exit Sub
End If
End Sub

```

از کنترل های خوش دست می توان به کنترل ListView اشاره کرد که برای راه انداختن آن می توان به طریق زیر عمل کرد:

```

Dim X As ListItem
Dim Y As ColumnHeader
ListView1.View = lvwIcon
Set X = ListView1.ListItems.Add(, , "System Num:" &
Adodc1.Recordset.Fields(1) & vbCrLf &
Adodc1.Recordset.Fields(3), 1, 1)

```

همانطور که مشاهده می کنید ، برای ساخت گره جدید همانند کنترل TreeView عمل می کنیم، به دلخواه می توان نامی به عنوان کلید برای آن ذکر کرد که من اینجا از آن خود داری کرده ام. من در این کنترل بیشتر از فلاگ ها استفاده کرده ام، زیرا سه Node در treeview داشتم و باید داده ها را بین اینها جابجا می کردم به همین دلیل از سه فلاگ به نام های Sysnew sysnotdone ، sysdone می باشد که هر زمان که روی نودی کلیک کنیم ، اینها تغییر میکنند، به عنوان آشنایی با نحوه کار به این قطعه کد توجه کنید :

```

Private Sub TreeView2_NodeClick(ByVal Node As MSComCtlLib.Node)
On Error Resume Next

```

همین و بس

استفاده شخصی از این اثر ممنوع است- فقط نگاه کن ارزش یک چیز یاد بگیر-

```

ListView1.ListItems.Clear
/////////setting for list view of node number new////////
If Node.Text = "کارهای جدید برای امروز" then
SysNEW = True
SysDONE = False
SysNOTDONE = False
Dim X As ListItem
Dim Y As ColumnHeader
ListView1.View = lvwIcon
Adodc1.Recordset.MoveFirst
For j = 1 To Adodc1.Recordset.RecordCount
Set X = ListView1.ListItems.Add(, , "SysNum:" &
Adodc1.Recordset.Fields(1) & vbCrLf & Adodc1.Recordset.Fields(3), 1, 1)
Adodc1.Recordset.MoveNext
Next
/////////setting for list view of node number anjam nashode////////
ElseIf Node.Text = "کارهای انجام نشده" then
sysNEW = False
sysDONE = False
sysNOTDONE = True
ListView1.View = lvwIcon
Option2.Enabled = False
Adodc3.Recordset.MoveFirst
For j = 1 To Adodc3.Recordset.RecordCount
Set X = ListView1.ListItems.Add(, , "SysNum:" & Adodc3.Recordset.Fields(1)
& vbCrLf & Adodc3.Recordset.Fields(3), 1, 1)
Adodc3.Recordset.MoveNext
Next
'/////////setting for list view of node number anjam shode////////
ElseIf Node.Text = "کارهای انجام شده" Then
SysNEW = False
SysDONE = True
SysNOTDONE = False
ListView1.View = lvwIcon
Adodc2.Recordset.MoveFirst
For j = 1 To Adodc2.Recordset.RecordCount
Set X = ListView1.ListItems.Add(, , "SysNum:" & Adodc2.Recordset.Fields(1)
& vbCrLf & Adodc2.Recordset.Fields(3), 1, 1)
Adodc2.Recordset.MoveNext
Next
End If
End Sub

```

مشاهده می کنید که با کلیک بر روی هر نود، علاوه بر اینکه فلاگ ها برای استفاده های بعدی، در مناطق بعدی، تغییر میکند، با استفاده از یک حلقه، تمامی محتویات آن بانک هم در Listview نمایش داده می شود

پیشنهاد و نتیجه گیری

36

مطمئناً برای هر دانشجو، در این دانشکده تفکری از کار گروهی وجود دارد. زیرا معمولاً دانشجویان دو به دو در میزها برای کار می‌نشینند، پس بهتر آنکه کارها را به صورت گروهی در اختیار دانشجو قرار داده و این اختیار به وی داده شود که کارش را گروهی انجام دهد. در این صورت است که راهی برای پیشرفت دانشجو باز کرده اید. همچنین بسیار دیده شده که اهمیت کمی به Document ها می‌شود و در برخی موارد فقط نگاه جزیی کرده می‌شود. این کارها مسئولیت ایجاد می‌کند و مطمئناً نزد خداوند عز و جل مقصر محسوب می‌شوند.

در کل نتیجه گیری که می‌توان از انجام پروژه داشت این است که فرد به صورت تجربی و تحقیقی به مطالعه و جمع آوری اطلاعات پرداخته و به دنبال کسب نیازها و علاوه بر آن بدست آوردن دانش می‌رود که این عمل باعث یادگیری هر چه بیشتر وی شده و باعث پیشرفت در زمینه تحصیل و فن تجربه فرد می‌شود.

در قرن حاضر جنبه های تئوری کمی مختص کلاس و ... است و برای یک تکنیسین چیزی مهمتر از تجربه نیست.

امروز برنامه نویسی به شکل دستی انجام می‌شود. یعنی توسط نیروی انسانی، با خطای انسانی، هزینه های جانبی و ...

اکنون سرمایه گذاری های کلان برای ساخت نرم افزار هایی انجام شده که این نرم افزار ها، به صورت اتوماتیک برنامه نویسی می‌کنند.

است. Code Generator نام آنها یک زنگ خطر برای بیکار شدن نیروهای انسانی برنامه نویس. چند نکته در مورد نقش دانشجو در انجام پروژه های بزرگ

× نوآوری در موضوع پروژه. چرا که Insert, Select, Update, Delete را همه بلد اند.

× پروژه ترس ندارد! پروژه ای انتخاب کنید که چیزی برای آموختن در آن باشد.

× پروژه های بزرگ و گروهی.

× طوری پروژه انتخاب کنید که بعد از انجام پروژه بتوانید آن را به فروش برسانید.

× در پروژه های نرم افزاری، به سمت طراحی و تحلیل شیء گرا حرکت کنید.

× به کاربرد در پروژه های خود اهمیت دهید.

× نقش فعال در تشکل های علمی دانشجویی داشته باشید. بسیاری از ایده های جدید، آموختن تکنولوژی های جدید و ... در جمع های علمی مطرح می‌شود.

پروژه ی شما، رزومه ی شماست و رزومه، تاثیر مستقیم بر موقعیت شغلی مناسب دارد.

راهنمای سریع کاربر

این صفحه شامل 3 قسمت است که قسمت بالایی را منو (Menu) و قسمتی که Label برنامه در آن قرار می گیرد و قسمت سوم که در پایین ترین قسمت وجود دارد و به اصطلاح به آن Status Bar گویند و وضعیت برنامه را مشخص میکند.

منوها از ابتدا عبارت اند از :

ثبت که شامل زیر منو های :

1) ثبت سیستم معیوب : در این قسمت سیستمی که مشتری آورده را ثبت کرده و ایرادات آن یادداشت می شود.

2) کارآموز جدید : در این قسمت کارآموزان جدید ثبت شده و به آنها نام کاربری تعلق می گیرد.

3) انبار که خود شامل دو زیر منوی ثبت در انبار و درخواست از انبار می باشد. در این قسمت تمامی موجودی انبار را که به صورت جزئی و فقط در حد رفع نیاز شامل می شود.

4) تحویل کاربه کارآموز: در این قسمت سیستم های معیوب به کارآموزان تحویل داده می شود تا تعمیر گردد.

5) تعویض کاربر: می توان از حالت مسئول به مدیر و بالعکس منتقل شد.

6) خروج.

شبکه که شامل زیر منوهای :

1) اتصالات کارآموزان که به صورت کادر انتخابی می باشد

2) بازگشت به گفتگوی آنلاین : که در صورتی که کارآموزی درخواستی مبنی بر اتصال داده باشد با این گزینه می توان به محل گفتگو بازگشت.

3) پیغام فوری: اگر مسئول نیاز ببیند که پیغامی را به کارآموزی ارسال کند از این گزینه استفاده می کند.

صدور نامه که شامل زیر منوهای :

1) نامه معرفی کارآموز: برای معرفی کارآموز به منظور ثبت نام وی در موسسه ی مورد نظر

2) نامه غذا : به کمک این نامه کارآموز می تواند از امکانات رستوران استفاده کند.

3) گزارش پیشرفت کاری پایانی کارآموز : گزارش نهایی که در آن به ارزیابی ، نمره دهی ، گزارش عملکرد و ... کارآموز می پردازد و در انتها فرمی چاپ می کند که در آن تمامی مشخصات به همراه برگه تاییدیه پایانی کارآموز گنجانده شده است.

نمایش و جستجو که شامل زیر منوهای :

41

5) کارهای تحویلی به کارآموز : کارهایی که در مدت زمان به کارآموز های مختلف برای تعمیر سپرده می شود را شامل می شود.

6) کارهای به اتمام رسیده : قضیه به این صورت است که ابتدا مسئول کاری را برای کارآموز ارسال کرده و کارآموز به آن می پردازد، پس از اینکه کارآموز کار را به پایان رساند، یا توانسته یا خیر؛ در این صورت اطلاعی را برای مسئول ارسال می کند که من توانستم کار فلان را تمام کنم، این گزینه ، آن قبیل کارها را نمایش می دهد.

مدیر که شامل زیر منوی : لازم به ذکر است که در صورتی این گزینه فعال می شود که شما با کاربر مدیر وارد شده باشید.

1) تعریف نام کاربری می باشد که می توان نام کاربری تعریف کرد
2) اصلاح کاربر : به منظور تغییر نام، تعویض کلمه رمز، حذف آن ، حذف نام کاربری و ... از این گزینه استفاده می شود.

3) محدوده دسترسی : در آن می توان انواع دسترسی ها به انواع مناطق برنامه را به کاربر مورد نظر داد.

گزارش که شامل :

1) اطلاعات کارآموزان: شامل اطلاعات کارآموزان است که در قالب فیلتر شده نمایش داده می شود، این گزارشات با نرم افزار کریستال ریپورت گرفته شده است.

2) اطلاعات تمامی کارآموزان: شامل اطلاعات کارآموزان است که هیچ فیلتری در آن به کار نرفته است و صریحا تمامی کارآموزان موسسه را ذکر کرده است.

3) تمام دریافتی ها از انبار : همانند گزینه بالا می باشد با این تفاوت که تمامی اجناسی که از انبار درخواست شده را نمایش می دهد

4) دریافتی ها از انبار : تمامی اجناسی که از انبار درخواست شده را نمایش می دهد اما در قالب فیلتر شده و به صورت ترکیبی و غیر ترکیبی

5) ثبتی ها در انبار: تمامی اجناسی که در انبار ثبت شده را نمایش می دهد اما در قالب فیلتر شده و به صورت ترکیبی و غیر ترکیبی

کمک که شامل زیر منوی :

1) وب سایت : برای افزایش منو گزاشته شده است

2) پشتیبانی

3) درباره

منابع :

استفاده شخصی از این اثر ممنوع است- فقط نگاه کن ازش یک چیز یاد بگیر-

- 1- مرجع کامل ویژوال بیسیک 6 انتشارات نص؛
- 2- آموزش ویژوال بیسیک 6 در 21 روز
- 3- برنامه نویسی پایگاه داده در VB6 در 21 روز
- 4- www.dev.ir
- 5- <http://sheidaian.persianblog.com>
- 6- www.barnamenevis.org
- 7- www.visualBasic.blogfa.com
- 8- بهره گیری از مقاله آموزشی تیم نرم افزاری In_chand_nafar

انتشار در اینترنت با هدف یادگیری