

بسم الله الرحمن الرحيم

مستند پروژه: آموزگار

دس شی کرا

زیر نظر استاد محترم:

جناب آقای مهندس زنگونی

تهیه کنندگان:

حسن دهجو

هادی خسروی

محمد قلیزاده

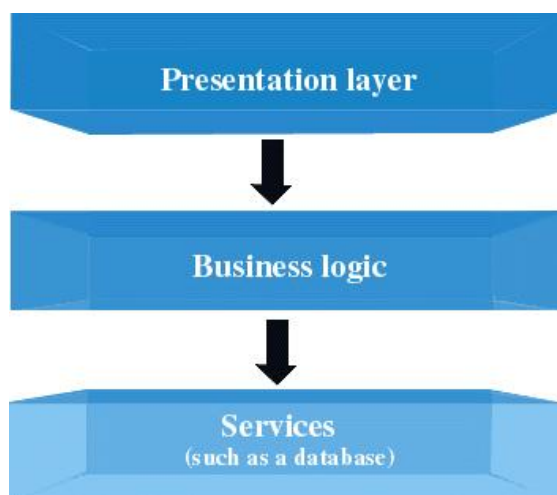
فهرست مطالب

معرفی برنامه	۳
معرفی جداول	۴
معرفی کلاسها	۵
کلاس DataAccess	۶
کلاس Class_Date	۶
کلاس Teacher , Sudent	۷
معرفی متد ها	۷
متد Delete	۷
متد SelectAll	۷
متد GetOnce	۸
متد Update	۸
کلاس Lesson	۸
کلاس Time	۸
کلاس User	۸
روال ویرایش اطلاعات	۹

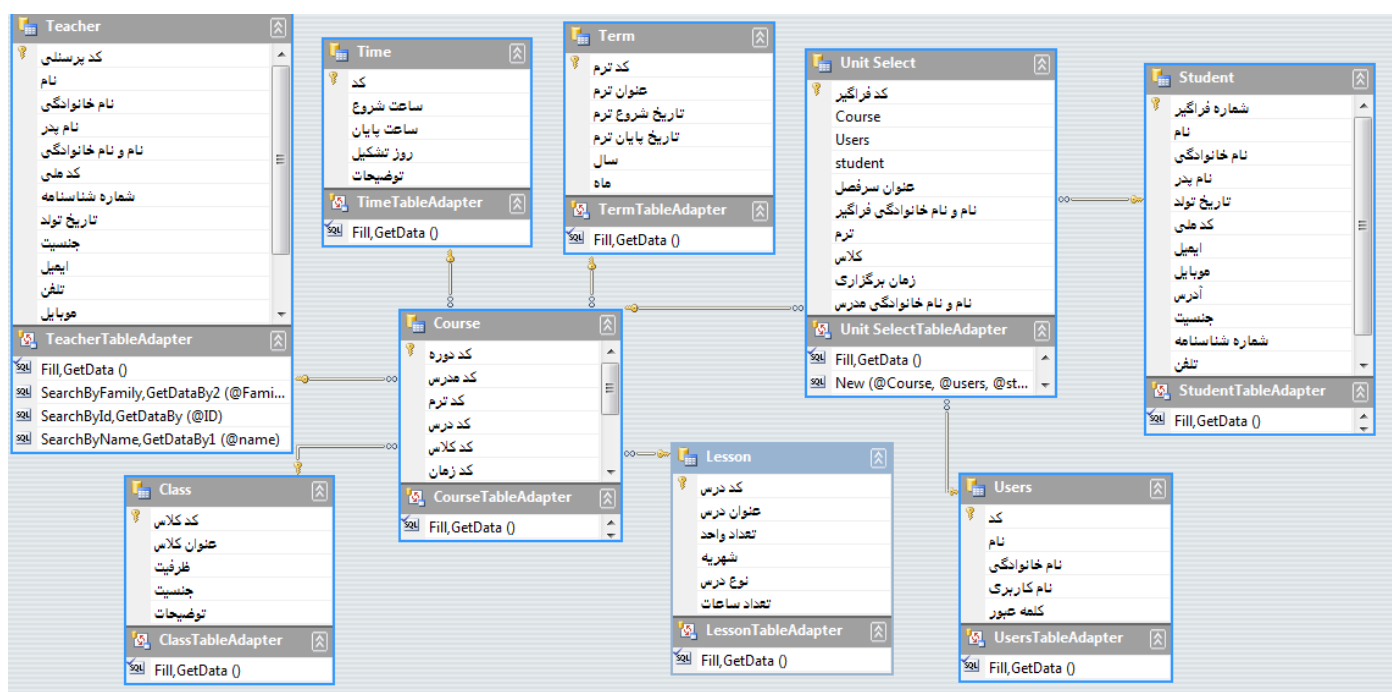
معرفی برنامه

برنامه آموزشگاه که ما طراحی کردیم توانایی ثبت اطلاعات فراگیران و مربیان و ثبت اطلاعات دروس و ترم ها و امکان مدیریت دوره ها و توانایی گزارش گیری در قسمت های مختلف را فراهم می کند .

این پروژه سعی شده است تا به صورت سه لایه شبیه سازی شود .



چون پروژه آموزشگاه یک نرم افزار است که با بانک اطلاعاتی زیاد سر و کار دارد ابتدا به تشریح جداول بانک اطلاعاتی و نحوه ی ارتباط آنها با هم می پردازیم سپس قسمت های دیگر پروژه را معرفی می کنیم .



همانطور که در شکل مشخص شده است این دیتا بیس شامل ۹ جدول است .

معرفی جداول :

: Student

در این جدول اطلاعات فراگیران قرار می گیرد که شامل اطلاعات فردی ، اطلاعات تماس و ... می باشد.

: Teacher

اطلاعات مربیان و مدرسان دوره های مختلف در این جدول ثبت می شود.

: Term

اطلاعات ترم شامل عنوان ترم ، تاریخ شروع و تاریخ پایان در این جدول ثبت می شود.

: Lesson

اطلاعات دروس شامل عنوان ، تعداد واحد ، میزان شهریه ، تعداد ساعات و ...

: Class

اطلاعات کلاسها و ظرفیت آن و ...

: Time

اطلاعات زمان برگزاری کلاس مثلاً کلاس شی گرا در روز شنبه ، ساعت ۱۰ تا ۱۲ در این جدول قرار می گیرد.

: Users

اطلاعات کاربران و استفاده کنندگان از سیستم در این جدول قرار می گیرد مانند نام کاربری و کلمه عبور و ...

: Course

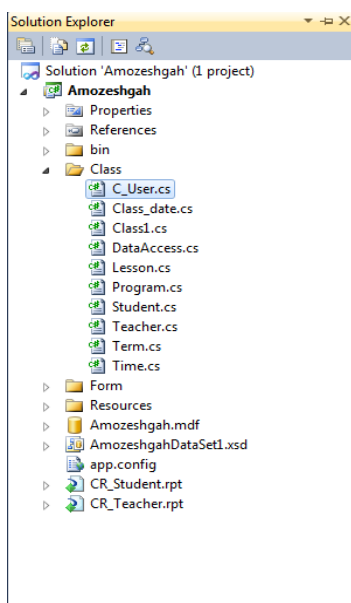
اطلاعات یک دوره آموزشی که شامل اطلاعات ترم ، درس ، کلاس ، زمان و مربی است در این جدول قرار می گیرد به عنوان مثال در یک دوره آموزشی چه درسی تدریس می شود که اطلاعات آن از جدول دروس و توسط چه کسی تدریس می شود که اطلاعات مربی از جدول Teacher و در چه ترمی ارائه می شود ، اطلاعات ترم از جدول term و در چه کلاسی و چه زمانی برگزار می شود .

Unit Select

جدول انتخاب واحد که مشخص کننده اطلاعات ثبت نام شدگان در یک دوره خاص می باشد و علاوه بر اطلاعات دوره کد فراگیری که در هر دوره ای ثبت نام کرده است در این جدول ثبت می شود.

The screenshot shows a software window titled 'فرم اصلی' (Main Form) with a menu bar containing 'برنامه مدیریت آموزشگاه' (Institution Management Program) and a dropdown menu 'فراگیران' (Students). The menu bar also includes 'مدیریت سیستم' (System Management), 'نمره و کارنامه' (Grade and Transcript), 'انتخاب واحد' (Unit Selection), 'دوره ها' (Courses), 'گزارشگیری' (Reporting), 'کلاس و زمان برگزاری' (Class and Scheduling), 'دروس و ترم' (Courses and Semester), and 'مربیان' (Staff). The main area has three tabs: 'بت نام فراگیر' (Student Registration), 'لیست فراگیران' (Student List), and 'ویرایش اطلاعات فراگیران' (Edit Student Information). The 'بت نام فراگیر' tab is active, showing a large empty gray area for data entry. The status bar at the bottom displays 'طراحی: حسن دهجو، هادی خسروی، محمد قلبيزاده' (Designed by: Hassan Dehghani, Hadi Khosravi, Mohammad Ghalibzadeh) and 'Time: ساعت' (Time: Hour) and 'Date: تاریخ' (Date: Date).

معرفی کلاسها:



این کلاس ارتباط برنامه را با بانک اطلاعاتی برقرار می کند یعنی هر زمانی که خواستیم اطلاعاتی در دیتابیس درج کنیم یا اطلاعات یک جدول را استخراج کنیم از این کلاس برای ارتباط بین کلاس های برنامه با دیتابیس استفاده شده است حال به معرفی خصیصه ها و متد های آن می پردازیم :

```
class DataAccess
{
    //public string ServerName;
    public string DBName = "Amozeshgah.mdf";
    // public string username;
    public string password;
    SqlConnection con;
    SqlCommand cmd;
    SqlDataAdapter da;

    public DataAccess()
    {
        con = new SqlConnection();
        cmd = new SqlCommand();
        da = new SqlDataAdapter();
        cmd.Connection = con;
        da.SelectCommand = cmd;
    }

    public void connect()
    {
        string cs = "Data source=\\SQLEXPRESS;Attachdbfilename=|DataDirectory|\\{0};Integrated security=true;user Instance=true";
        cs = string.Format(cs, this.DBName);
        con.ConnectionString = cs;
        con.Open();
    }

    public void Disconnect()
    {
        con.Close();
    }
}
```

این کلاس شامل ۶ خصیصه است که اطلاعات ورود به دیتابیس و سه شی برای ارتباط با دیتابیس می باشد .

متد سازنده این کلاس خصیصه های کانکشن و SqlCommand و SqlDataAdapter را مقدار دهی می کند

متد Connect برای باز کردن یک کانکشن با دیتابیس استفاده شده است و متد Disconnect برای بسته شدن کانکشن دیتابیس می باشد.

این کلاس شامل دو متد دیگر هم می باشد که برای اجرا کردن دستورات sql از متد docammand و برای استخراج اطلاعات یک جدول خاص از متد Select استفاده می شود.

کلاس Class_date :

از این کلاس برای تبدیل تاریخ میلادی به شمسی استفاده می شود و تنها شامل یک متد Shamsi می باشد که تاریخ میلادی را دریافت و به شمسی تبدیل می کند .

به دلیل تشابه زیاد این دو کلاس این دو کلاس را با هم معرفی می کنیم

```
class Student
{
    public int ID;
    public string name;
    public string family;
    public string father;
    public string birth;
    public string code;
    public string email;
    public string mobile;
    public string address;
    public string genus;
    public string shomareh;
    public string tel;
    public string date;

    public DataAccess da = new DataAccess();
}
```

۷

همانطور که در شکل مشاهده می شود خصیصه های این کلاس برای نگهداری اطلاعات فردی و اطلاعات تماس می باشد و همانطور که گفته شد از کلاس DataAccess هم برای عملیات روی دیتابیس استفاده شده است .

معرفی متدها

اکثر کلاسهایی که نوشتیم دارای متد های مشترک هستند که در همین به شرح آن می پردازیم :

متد Delete

این متد در همه کلاس هایی که نوشته شده است اطلاعات آن شی را از دیتابیس حذف می کند به عنوان مثال این متد در کلاس Lesson برای حذف اطلاعات درس مشخص شده و یا این متد در کلاس Teacher برای حذف اطلاعات یک مربی استفاده می شود که دستور Delete اس کیو ال را توسط کلاس DataAccess فراخوانی می کند و باعث حذف اطلاعات رکورد مشخص شده می شود.

متد SelectAll

این متد هم در کلاسهای ... , Lesson , Time , Student , Teacher استفاده شده است هدف این متد گرفتن اطلاعات از جدول مشخص شده و قرار دادن آن در DataGrid می باشد . این متد دستور select را روی جدول مشخصی اجرا می کند مانند دستور زیر که اطلاعات همه فراگیران را استخراج می کند .

```
select * from student
```

متد SelectAll برای ارتباط با دیتابیس متد docommand از کلاس DataAccess را فراخوانی می کند .

متد GetOnce

این متد اطلاعاتی رکورد مشخصی (فقط یک رکورد) را از جدول مشخص شده فراخوانی می کند و برای این کار متد Select از کلاس DataAccess را فراخوانی می کند معمولا این متد زمانی استفاده می شود که ما بخواهیم اطلاعات یک شی مانند فراگیر را در یک فرم مشخص ببینیم و بتوانیم اطلاعات آن را ویرایش کنیم استفاده می شود در ادامه که روال ویرایش اطلاعات را بیان کنیم نحوه ی استفاده از این متد را متوجه خواهید شد.

متد Update

متد update برای ویرایش اطلاعات جدول در دیتابیس استفاده می شود که در ادامه به صورت کامل روال ویرایش اطلاعات به صورت کامل شرح می دهیم که کاربرد این متد مشخص می شود.

کلاس های Time , Lesson , Term , Class1 , C_user

کلاس Time اطلاعات زمان برگزاری کلاس هاست و شامل خصیصه هایی است که در جدول Time مشخص شده است .

کلاس Lesson دارای خصیصه هایی است که فیلد های جدول Lesson را در دیتابیس تشکیل می دهد و دارای متدهای Delete , Update , GetAll , GetOnce می باشد که در صفحه قبل این متدها معرفی شدند

کلاس Term دارای خصیصه های جدول Term است و شامل متدهای Delete , Update , GetAll , GetOnce , می باشد

روال ویرایش اطلاعات در دیتابیس

با توجه به اینکه روال های اصلی در برنامه هایی که با دیتابیس سروکار دارند معمولاً درج و ویرایش و حذف از جمله مهم ترین اتفاقاتی است که در اینجور برنامه ها رخ می دهد بنابراین برنامه آموزشگاه هم که ما روی آن کار می کنیم اکثر کد های برنامه با دیتابیس سروکار دارند که از آن جمله می توان به درج اطلاعات فراگیران ، مربیان ، درس ها و ... اشاره کرد یکی از مهم ترین دستورات در sql دستور update می باشد که نیاز به اجرای دو دستور یعنی دستور select و اجرای دستور update می باشد . حال به تشریح این روال می پردازیم

ابتدا دستور update را معرفی می کنیم

Update **TableName** Set **Column1** = **Value1** , **Column2** = **Value2** , ... where [key] = value

همان طور که در دستور update مشاهده می شود نیاز به نام جدول که با رنگ قرمز مشاهده می شود و فیلد های جدول که با رنگ سبز مشاهده می شود و مقادیر این فیلد ها که با رنگ آبی مشخص شده نیاز داریم . از مهمترین قسمت های دستور update قسمت where می باشد که ما تعیین می کنیم کدام رکورد جدول باید مقادیر فیلد های آن عوض شود .

نکته : اگر در دستور update قسمت where نوشته نشود همه رکورد های جدول ویرایش می شود به این صورت که مثلاً ما می خواهیم در جدول فراگیران نام فراگیری که اشتبهاً حسین درج شده و فامیلی او اکبری می باشد را می خواهیم به محمد حسین علوی تغییر دهیم حال اگر در دستور update را بدون where مانند شکل زیر بنویسیم چه اتفاقی می افتد؟

Update student name = " mohammad hosien " , family = " alavi "

خوب جوابش مشخص است چون ما در قسمت where رکورد خاصی را مشخص نکردیم این دستور بر تمام رکورد های جدول تاثیر می گذارد و نام و فامیلی تمام رکورد ها یکسان می شود و اطلاعات سایر فراگیران از بین می رود .

خوب تا حالا هر چه گفته شد در مورد دستور update بود از اینجا به بعد به طور مثال در برنامه مون اطلاعات یک درس را به صورت مرحله به مرحله اجرا می کنیم

مرحله اول

Get selected Id (key) from DataGridView Or ...

اولین مرحله در ویرایش اطلاعات یک درس یا هر چیز دیگر بدست آوردن کلید آن می باشد حال ممکن است این کلید در دیتاگرید یا هر کنترل دیگر برای ما نشان داده شود که در برنامه ما در دیتا گرید نشان داده می شود و کلید هر رکورد در اولین ستون دیتا گرید می باشد ما برای اینکه این کلید را بدست آوریم از دستورات زیر استفاده می کنیم

```
int ID;  
ID = Int32.Parse(dataGridViewX1.CurrentRow.Cells[0].Value.ToString());
```

در دستور بالا گفتیم که مقدار اولین ستون از سطر جاری در یتا گرید را به ما بده و آن را به عدد تبدیل کردیم و داخل متغیر ID قرار داریم

خوب این از مرحله اول می ریم سراغ مرحله دوم

مرحله اول

Get Other Information From Table By Select

در این مرحله باید به دیتابیس وصل شویم و سایر اطلاعات این رکورد که کلید آن را داریم به کاربر نشان دهیم پس دستورات زیر را می نویسیم

```
Lesson lss = new Lesson();  
lss.ID = ID;  
lss.GetOne();
```

خوب در دستور اول یک شی از کلاس درس درست کردیم و مقدار ای دی آن را برابر با کلیدی که از دیتا گرید گرفته بودیم قرار دادیم و در خط سوم هم متد GetOne این شی را اجرا کردیم

خوب متد GetOne چه کار انجام می دهد؟

```
public void GetOne()  
{  
    DataTable dt = new DataTable();  
    da.connect();  
    dt = da.select("select * from Lesson where L_id=" + this.ID);  
    this.Title = dt.Rows[0]["L_ttile"].ToString();  
    this.Vahed = dt.Rows[0]["L_vahed"].ToString();  
    this.Sharyeh = dt.Rows[0]["L_sharyeh"].ToString();  
    this.kind = dt.Rows[0]["L_kind"].ToString();  
    this.Duration = dt.Rows[0]["L_duration"].ToString();  
    da.Disconnect();  
}
```

خوب این متد در جدول Lesson جستجو می کند و رکوردی که با کلید مشخص شده برابر است را در داخل DataTable قرار می دهد حالا ما می توانیم فیلدها (سایر اطلاعات) درس مورد نظر را برداریم و در شی مون ست کنیم به عنوان مثال می گویم title شی کلاس برابر با dt سطر صفرم و ستونی که عنوان آن هست L_title قرار بده و بقیه هم به همین شکل ...

مرحله سوم

Created An instance of edit form

خوب تا اینجا یک بار با دستور select به دیتابیس مراجعه کردیم و سایر اطلاعات درسی که می خواستیم ویرایش کنیم را بدست آوردیم

```
Update_Lesson frm = new Update_Lesson();
```

در این مرحله یک نمونه از فرمی که می خواهیم اطلاعات بیرون کشیده شده از دیتابیس را در آن نشان دهیم ایجاد می کنیم

مرحله چهارم

Read from Dt and fill in Edit Form textboxes

در این مرحله textbox های روی فرمی که شی frm را از روی آن ساختیم مقدار دهی می کنیم که این کار با شی که قبلا از کلاس lesson درست کردیم انجام می دهیم و مقدار هر attribute شی را به textbox مورد نظر در فرم ویرایش انتقال می دهیم نه از dt

```
frm.txtID.Text = lss.ID.ToString();  
frm.txtTitle.Text = lss.Title;  
frm.Kind.Text = lss.kind;  
frm.txtVahed.Text = lss.Vahed;  
frm.txtDuration.Text = lss.Duration;  
frm.txtSharyeh.Text= lss.Sharyeh;
```

مرحله پنجم

Show Edit Form

در این مرحله فرمی که textbox های آن را مقدار دهی کردیم به کاربر نشان می دهیم به شکل زیر

```
frm.ShowDialog();
```

ویرایش اطلاعات دروس

کد درس : 1020

عنوان درس : طراحی الگوریتم

تعداد واحد : 3

شهریه : 350000

نوع درس : انتخاب نوع درس

تعداد ساعات : 32

انصراف ✖

بروزرسانی ✔

همون طور که در شکل نشان داده می شود اطلاعات درس مورد نظر با کلید ۱۰۲۰ که مقدار این textbox قابل تغییر نیست نشان داده می شود و کاربر اطلاعات مورد نظرش را ویرایش می کند که بعد از ویرایش بر روی دکمه بروز رسانی کلیک می کند .

مرحله ششم و هفتم

- Create An Instance Form Lesson
- And Update Record From Table

خوب در این مرحله دوباره یک شی از کلاس lesson ایجاد می کنیم و مقادیری که کاربر ویرایش کرده را داخل آنها قرار می دهیم بدین صورت

```
Lesson lss = new Lesson();

lss.ID = Int32.Parse(txtID.Text);
lss.Sharyeh = txtSharyeh.Text;
lss.Title = txtTitle.Text;
lss.Vahed = txtVahed.Text;
lss.Duration = txtDuration.Text    ;
lss.kind = Kind.Text;
lss.Update();
this.Close();
```

و متد update شی کلاس را اجرا می کنیم و فرم را می بندیم و دوباره برمی گردیم به فرم مبدا که اطلاعات همه دروس در داخل اون نشون داده می شود برای اینکه اطلاعاتی که کاربر ویرایش که در داخل دیتاگرید نشون داده بشه نیاز به refresh اون داریم که این کار در مرحله آخر انجام میشه

مرحله هشتم

➤ Load Data By Call Form Load Method

```
Edit_Lesson_Load(null, null);
```

خوب تنها یک دستور در این مرحله اجرا می کنیم و اون هم فرم رو دوباره load می کنیم تا دوباره اطلاعات همه دروس رو از دیتابیس بخونه و درسی که ما هم ویرایش کردیم بینم اطلاعاتش تغییر کرده
