**شیوه نامه­ی تهیه­ی داکیومنت پروژه­ی پایانی رشته کامپیوتر**

1. اندازه­ی متن فارسی 14 و با قلم B Nazanin
2. اندازه­ی متن انگلیسی 12 و با قلم Times New Roman
3. اندازه تیتر (عنوان) همانند نمونه داکیومنت پیوست باشد.
4. حاشیه متن از طرف راست 3 سانتی متر و از طرف چپ 2 سانتی متر و از طرف بالا و پائین 2.5 سانتی متر باشد.
5. فاصله­ی بین خط­ها 1.25 سانتی متر
6. خط اول هر پاراگراف 0.5 سانت تورفتگی داشته باشد.
7. متن توضیحی جدول در بالای جدول آورده شود. با اندازه­ی 10
8. متن توضیحی شکل در پائین شکل آورده شود. با اندازه­ی 10
9. در شماره گذاری فهرست مطالب، شکل­ها، جدول­ها و ... ابتدا شماره فصل و بعد شماره بخش یا شماره شکل یا شماره جدول آورده شود.
10. در نوشتن متن در صورت نیاز به علائم ویرگول، نقطه گذاری و .... ابتدا علائم نوشته و سپس فاصله گذاشته شود.
11. ....

**\* دانشجوی گرامی در صورت داشتن سوالی لطفا با داخلی 112 دانشگاه (مهندس نورزاد) تماس بگیرید.**

* **دانشجوی گرامی حتما از نمونه داکیومنت پروژه استفاده شود.**

با تشكر

مديرگروه صنعت

نورزاد



مرکز آموزش

علمی کاربردی

گنبدکاووس2

کاردانی رشته­ی نرم‌افزار

موضوع پروژه:

مو....

استاد راهنما:

نام استاد

نام دانشجویان :

نام و نام خانوادگی

نیمسال دوم 92-1391

تشكر و قدرداني:

تشکر و قدردانی........

چکیده

در این قسمت چکیده نوشته شود.

واژه‌هاي كليدی:

فهرست مطالب

عنوان صفحه

[فصل1 تعریف مسئله **Error! Bookmark not defined.**](#_Toc291269076)

[1-1. تعریف کتابخانه **Error! Bookmark not defined.**](#_Toc291269077)

[1-2. اهداف تحقیق 3](#_Toc291269078)

[1-3. قابلیت های نرم افزار **Error! Bookmark not defined.**](#_Toc291269079)

[فصل2 ابزار های بکار رفته 5](#_Toc291269084)

[2-1. Microsoft SQL Server 6](#_Toc291269085)

[2-1-1. تاریخچه 6](#_Toc291269087)

[2-2-2. SQL Server 2008 8](#_Toc291269088)

[2-2. زبان برنامه نویسی C# 9](#_Toc291269091)

[2-3. Microsoft Visual Studio 2010 **Error! Bookmark not defined.**](#_Toc291269092)

[فصل3 پایگاه داده 11](#_Toc291269099)

[3-1. مقدمه 12](#_Toc291269100)

[3-2. معرفی جداول پایگاه داده 13](#_Toc291269101)

[3-2-1. جدول tbl\_categori (طبقه بندی) 13](#_Toc291269102)

[3-2-2. جدول tbl\_book (کتاب) 13](#_Toc291269103)

[3-2-3. جدول tbl\_customer (عضو) 15](#_Toc291269104)

[3-2-4. جدول tbl\_employe (کارمند) 16](#_Toc291269105)

[3-2-5. جدول tbl\_user (کاربر) 17](#_Toc291269105)

[3-3. ارتباط جداول برنامه کتابخانه در پایگاه داده 17](#_Toc291269106)

[فصل4 توضیح کد 19](#_Toc291269107)

[4-1. ارتباط با پایگاه داده 20](#_Toc291269108)

[4-2. فضای نام 20](#_Toc291269109)

[4-3. تعریف کلاس در برنامه 21](#_Toc291269109)

[4-4. تعریف شی ء اتصال 22](#_Toc291269109)

[4-5. درج اطلاعات در پایگاه داده 24](#_Toc291269109)

[4-6. ویرایش اطلاعات 26](#_Toc291269109)

[4-7. حذف یک سطر از جدول 28](#_Toc291269109)

[4-8. حذف یک امانت 29](#_Toc291269109)

[4-9. نمایش تاریخ شمسی سیستم 30](#_Toc291269109)

[4-10. جستجو 31](#_Toc291269109)

[4-11. درج عکس 33](#_Toc291269109)

[4-12. بررسی خالی نبودن textbox 34](#_Toc291269109)

[4-13. بررسی درج عدد صحیح در textbox 34](#_Toc291269109)

[فصل5 راهنمای برنامه 35](#_Toc291269116)

[5-1. شروع کار برنامه 36](#_Toc291269117)

[5-1. شروع ثبت کتاب 38](#_Toc291269117)

[5-2-1. ثبت رده بندی 38](#_Toc291269120)

[5-2-2. ویرایش رده 40](#_Toc291269121)

[5-2-3. ثبت کتاب 40](#_Toc291269121)

[5-2-4. ویرایش کتاب 42](#_Toc291269121)

[5-3. ثبت اعضای کتابخانه 43](#_Toc291269122)

[5-4. ویرایش اعضای کتابخانه 44](#_Toc291269122)

[5-5. چاپ کارت عضویت 44](#_Toc291269122)

[5-6. ثبت کارمندان 46](#_Toc291269122)

[5-7. ویرایش کارمندان 47](#_Toc291269122)

[5-8. ثبت کاربران 48](#_Toc291269122)

[5-8. ثبت کاربران 48](#_Toc291269122)

[5-9. ویرایش کاربران 49](#_Toc291269122)

[5-10. گزارش گیری 50](#_Toc291269122)

[5-11. ثبت امانت 51](#_Toc291269122)

[5-12. ویرایش امانت 52](#_Toc291269122)

[5-13. حذف امانت 53](#_Toc291269122)

[نتیجه گیری 54](#_Toc291269116)

[پیشنهاد 54](#_Toc291269116)

[منابع 55](#_Toc291269116)

فهرست اشکال

عنوان صفحه

[فصل1 تعریف مسئله **Error! Bookmark not defined.**](#_Toc291269076)

[فصل2 ابزار های بکار رفته 5](#_Toc291269084)

[فصل3 پایگاه داده 11](#_Toc291269099)

[فصل4 توضیح کد 19](#_Toc291269107)

[فصل1 تعریف مسئله 1](#_Toc340129552)

[1-1. تعریف کتابخانه : 2](#_Toc340129553)

[1-2. اهداف تحقیق 3](#_Toc340129554)

[1-3. قابلیت های نرم افزار : 4](#_Toc340129555)

[فصل2 ابزار های بکار رفته 5](#_Toc340129556)

[2-1. Microsoft SQL Server 6](#_Toc340129557)

[2-1-1. تاریخچه 6](#_Toc340129558)

[2-1-2. SQL Server 2008 8](#_Toc340129559)

[2-2. زبان برنامه نویسی C# 9](#_Toc340129560)

[2-3. Microsoft Visual Studio 2010 10](#_Toc340129561)

[فصل3 پایگاه داده 11](#_Toc340129562)

[3-1. مقدمه 12](#_Toc340129563)

[3-2. معرفی جداول پایگاه داده 13](#_Toc340129564)

[3-2-1. جدول tbl\_categori (طبقه بندی) 13](#_Toc340129565)

[3-2-2. جدول tbl\_book (کتاب) 13](#_Toc340129566)

[3-2-3. جدول tbl\_customer (عضو) 15](#_Toc340129567)

[3-2-4. جدول tbl\_employe (کارمند) 16](#_Toc340129568)

[3-2-5. جدول tbl\_ user (کاربر) 17](#_Toc340129569)

[3-3. ارتباط جداول برنامه کتابخانه در پایگاه داده 17](#_Toc340129570)

[فصل4 توضیح کد 19](#_Toc340129571)

[4-1. ارتباط با پایگاه داده 20](#_Toc340129572)

[4-2. فضای نام 20](#_Toc340129573)

[4-3. تعریف کلاس در برنامه 21](#_Toc340129574)

[4-4. تعریف شی ء اتصال 22](#_Toc340129575)

[4-5. درج اطلاعات در پایگاه داده 24](#_Toc340129576)

[4-6. ویرایش اطلاعات 26](#_Toc340129577)

[4-7. حذف یک سطر از جدول 28](#_Toc340129578)

[4-8. حذف یک امانت 29](#_Toc340129579)

[4-9. نمایش تاریخ شمسی سیستم 30](#_Toc340129580)

[4-10. جستجو 31](#_Toc340129581)

[4-11. درج عکس 33](#_Toc340129582)

[4-12. بررسی خالی نبودن textbox 34](#_Toc340129583)

[4-13. بررسی درج عدد صحیح در textbox 34](#_Toc340129584)

[فصل5 راهنمای برنامه 35](#_Toc340129585)

[5-1. شروع کار برنامه 36](#_Toc340129586)

[5-2. شروع ثبت کتاب 38](#_Toc340129587)

[5-2-1. ثبت رده بندی 38](#_Toc340129588)

[5-2-2. ویرایش رده 40](#_Toc340129589)

[5-2-3. ثبت کتاب 40](#_Toc340129590)

[5-2-4. ویرایش کتاب 42](#_Toc340129591)

[5-3. ثبت اعضای کتابخانه 43](#_Toc340129592)

[5-4. ویرایش اعضای کتابخانه 44](#_Toc340129593)

[5-5. چاپ کارت عضویت 45](#_Toc340129594)

[5-6. ثبت کارمندان 46](#_Toc340129595)

[5-7. ویرایش کارمندان 47](#_Toc340129596)

[5-8. ثبت کاربران 48](#_Toc340129597)

[5-9. ویرایش کاربران 49](#_Toc340129598)

[5-10. گزارش گیری 50](#_Toc340129599)

[5-11. ثبت امانت 51](#_Toc340129600)

[5-12. ویرایش امانت 52](#_Toc340129601)

[5-13. حذف امانت 53](#_Toc340129602)

[نتیجه گیری : 54](#_Toc340129603)

[پیشنهاد : 54](#_Toc340129604)

[فصل1 تعریف مسئله 1](#_Toc340129605)

[فصل2 ابزار های بکار رفته 5](#_Toc340129606)

[فصل3 پایگاه داده 11](#_Toc340129607)

[فصل4 توضیح کد 19](#_Toc340129608)

[فصل5 راهنمای برنامه 35](#_Toc340129609)

1. تعریف مسئله

1-1. تعریف کتابخانه :

کتابخانه نهادی است که با گردآوری ، سازماندهی ونگهداری دانش مدون بشر امکان رشد اندیشه ، شکوفایی استعداد و باروری ذهن خلاق انسان پویا و جستجوگر را فراهم می کند. با توجه به گسترش روز افزون علوم و فنون در زمانها و مکانهای مختلف تنها کتابخانه است که دسترسی آسان به انبوه دانش آدمی را میسر می سازد و تبادل افکار را به بار می آورد .

از نظر فیزیکی می توان تعاریف زیر را برای کتابخانه در نظر گرفت:

- کتابخانه مکانی است برای حفظ و نگهداری آثار مکتوب و غیر مکتوب که طی تاریخ تمدن بشر به وجود آمده است.

- کتابخانه مخزنی است برای حفظ مواد و منابع هنری، علمی، ادبی، اجتماعی و غیره که بر روی رسانه های مختلف ثبت و ضبط شده اند.

- کتابخانه مکانی است که انواع منابع و مواد در آن به صورت سازماندهی شده حفظ و نگهداری می شود.

- کتابخانه مکان فیزیکی است که انواع رسانه های مکتوب، دیداری و شنیداری و الکترونیکی در آن نگهداری می شود.

اما از نظر کارکردی و اجتماعی تعاریف زیر را می توان ارائه داد:

- کتابخانه نهادی است اجتماعی که سازوکار اصلی ذخیره سازی، حفاظت و اشاعه دانش و اطلاعات را فراهم می سازد.

- کتابخانه نهادی است زاده ی جامعه که در طول تاریخ تمدن در خدمت حفظ و نشر دانش بوده است.

- کتابخانه نهادی اجتماعی است که روح تمدن بشری را حفظ و نگهداری کرده است.

- کتابخانه نهادی است که کارکردهای اصلی حرفه ی کتابداری یعنی جمع آوری، سازماندهی، اشاعه و مدیریت منابع اطلاعاتی در آن انجام می شود.

- کتابخانه پلی است میان دانش مضبوط بشری بین نسل های گذشته، حال و آینده.

با تلفیق این تعاریف می توان کتابخانه را این اینگونه تعریف کرد: کتابخانه نهادی اجتماعی است که با ذخیره سازی، حفاظت و اشاعه ی پیشنه های مکتوب، دیداری و شنیداری و الکترونیکی با استفاده از خدمات کارکنان آموزش دیده در خدمت تعلیم و تربیت و توسعه سیاسی، اقتصادی، اجتماعی، علمی و فرهنگی است.

1-2. اهداف تحقیق

همانطور که در تعریف کتابخانه گفته شد ، کتابخانه مکانی برای جمع آوری اطلاعات کتاب ها و منابع می باشد. و لزوماً برای جمع آوری این منابع و اطلاعات نیاز به ثبت و درج آنها در مکانی برای سهولت در دسترسی به آنها باشد.

1. ابزار های بکار رفته

2-1. Microsoft SQL Server



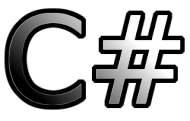
یک بانک اطلاعاتی از نوع دیتابیس های رابطه ای یا Relational Database می باشد که توسط کمپانی Microsoft ارایه شده و وظیفه اصلی آن ذخیره و بازیابی اطلاعات بر اساس درخواست نرم افزارهای دیگر میباشد. این نرم افزار دارای نسخه های بسیار متنوع و متناسب با مخاطبان متفاوت و محیط های کار مختلف ( از نرم افزارهای کوچک که اطلاعات در ماشین لوکال خود ذخیره می کنند تا میلیون ها کاربر و ماشین که احتیاج به دستیابی به اطلاعات از طریق شبکه به طور همزمان دارند.)

همانطور که نام این نرم افزار مشخص می باشد از زبان های Query مانند T-SQL و ANSI SQL استفاده می کند.

2-1-1. تاریخچه

کد های پایه ای این نرم افزار ابتدا توسط شرکت Sybase SQL Server طراحی شده بود که بعدها شرکت مایکروسافت امتیاز کامل آن نرم افزار را خریداری نمود.هدف مایکروسافت ازخریداری این نرم افزار، ارتقا دادن آن به سطح عظیم و در نهایت رقابت با دیتابیس های ارایه شده توسط شرکت های Oracle و IBM بود. ......

2-2. زبان برنامه نویسی C#



سی‌شارپ (C#) زبانی شیءگرا و سطح بالا (high level) از خانواده ی زبان‌های چارچوب دات‌نت شرکت مایکروسافت است.زبان #C ، یک زبان برنامه نویسی چند الگویی است و منظم شده مدل‌های تابعی، امری، عمومی، شی گرا و جز گرا می‌باشد. این زبان توسط مایکروسافت و جزئی از دات نت به وجود آمد و بعدا استانداردهای ECMAو ISO را نیز در بر گرفت. #C یکی از ۴۴ زبان برنامه نویسی ای است که توسط Common Language Runtime از .NET Framework پشتیبانی می‌شوند و در همه جا به وسیله Microsoft Visual Studio شناخته می‌شود. این زبان برپایه سادگی، مدرن بودن، همه منظوره و شی گرا بودن ساخته شد.Anders Hejlsberg، طراح زبان برنامه نویسی دلفی، سرپرستی تیم طراحان زبان #C را بر عهده داشت.این زبان دارای دستوری شی گرا مشابه ++C است و به شدت از زبان‌های جاوا و دلفی تاثیر پذیرفته‌است.در ابتدا نام این زبان COOL بود که مخفف C like Object Oriented Language بود، هر چند در جولای ۲۰۰۰، زمانی که مایکروسافت پروژه را عمومی اعلام کرد، اسم آن به #C تغییر پیدا کرد.

1. پایگاه داده

3-1. مقدمه

پایگاه داده یا همان Database ، مجموعه ای سازمان داده شده از اطلاعات است که به شکل جداول ذخیره می شوند. برای سازمان دهی این اطلاعات روشهای متفاوتی وجود دارد که هدف تمامی آنها فراهم کردن روشهایی مناسب برای سهولت در برقراری ارتباط با پایگاه های داده و استفاده از اطلاعات موجوددر آنها است. سیستم مدیریت پایگاه داده (DBMS) ، مکانیزمی را جهت ذخیره سازی و بازیابی اطلاعات در پایگاه داده فراهم می نماید. در حقیقت DBMS باعث می شود تا برنامه نویس بدون نگرانی در باره چگونگی ذخیره سازی داده ها در پایگاه داده و ساختار آنها ، به اطلاعات دسترسی پیدا کند و بتواند داده های جدید را در آن ذخیره نماید.

1. توضیح کد
   1. ارتباط با پایگاه داده

برای اینکه بخواهیم از اطلاعات داخل پایگاه داده و جدول ها استفاده کنیم ، اولین کار اتصال به پایگاه داده می باشد connection string . یک رشته متنی است که تنظیمات اتصال را مشخص میکند و از یه سری کلمات کلیدی تشکیل شده است مانند نام سرور ، نام دیتابیس ، مسیر دیتابیس و …  .

نمونه کد :

<configuration>

<appSettings>

<add key="cnnStr" value="server=COMPUTER1;database=Library;integrated security=true" />

</appSettings>

</configuration>

**Key** : نام کلید

**Server** : نام سرور

**Database :** نام بانک اطلاعاتی

**Integrated Security** : اگر مقدار آن را برابر false قرار دهیم باید حتما در connection string یوزرنیم و پسورد را برای استفاده از دیتابیس در connection string ذکر کنیم . اگر ture یا sspi باشد از اکانت ویندوز برای اعتبار سنجی استفاده میشود.

* 1. فضای نام

فضاهای نام روشی برای مدیریت کد نویسی هستند. برای مثال آن ها ایجاد شده اند تا تداخلی بین نام های توابع در برنامه شما رخ ندهد. این مساله در پروژه های بزرگ خود را نشان می دهد و ممکن است دو آیتم در یک پروژه نام های یکسانی را پیدا کنند. بدین وسیله این شانس تصادم و تداخل کاهش پیدا می کند.

**Using** : با استفاده از کلمه کلیدی Using می توان فضای نام درخواستی خودتان را به پروژه اضافه کنید.

برای اینکه بتونیم با بانکهای اطلاعاتی SQL SERVER کار کنیم ما باید از فضای نام System.Data.SqlClient استفاده کنیم.

در این فضای نام کلاسهای مختلف و قدرتمندی برای انجام خواسته های ما وجود دارن.من از بین تعداد بسیار زیاد این کلاسها چند تایی که خیلی زیاد استفاده می شن و بسیار مهم هستند رو برای شما معرفی خواهم کرد.این کلاسها عبارتند از :

·         SqlConnection

·         SqlCommand

·         SqlDataAdapter

·         Dataset

نمونه فضاهای نام مورد استفاده در برنامه:

using System.Configuration;

using Library\_Version1;

using System.Data.SqlClient;

using System.Globalization;

* 1. تعریف کلاس در برنامه

برای اینکه بخواهیم در برنامه بعضی از متغییر ها در تمام فرم ها اضافه بشوند از کلاس استفاده می کنیم. برای اینکار از منوی Project گزینه ی Add class را انتخاب کنید تا کلاسی جدید به نام clsDate.cs را اضافه نماییم.

نمونه کد :

class clsMain // نام کلاس

{

static public string cnnStr = string.Empty

// متغیر سراسری برای قرار دادن رشته اتصال به پایگاه داده کتابخانه

}

* 1. تعریف شی ء اتصال

**SqlConnection** : با استفاده از این کلاس ما می تونیم با بانک اطلاعاتی ارتباط برقرار کنیم. برای ایجاد یک شی از این کلاس به صورت زیر عمل می کنیم :

public partial class frmMain : Form

{

SqlConnection cnn = new SqlConnection(clsMain.cnnStr );

}

همانطور که مشاهده می نمائید برای اینکه این کلاس در تمام برنامه اجرا شود آنرا در یک تابع عمومی قرار داده ایم. در SqlConnection در داخل پرانتز از کلاس clsMain رشته cnnStr که نوشته بودیم را فراخوانی کرده ایم.

**SqlCommand** : برای اجرای دستورات بر روی بانک اطلاعاتی از این کلاس استفاده می نمائیم. شی ء دستور را به صورت زیر می نویسیم :

SqlCommand cmd = new SqlCommand();

**: SqlDataAdapterاین کلاس در برنامه های بانک اطلاعاتی ، همانند پلی بین جداول اطلاعاتی و نیز داده های موجود در حافظه که به وسیله ی** DataSet **نگهداری می شوند ، عمل می کنند . و برای استفاده از آن در برنامه باید یک شیئ از نوع آن ساخته شود .**

SqlDataAdapter dadp = new SqlDataAdapter();

**Fill : با استفاده از این متد در کلاس** SqlDataAdapter **, میتوانیم دستور موجود در** ُSelectCommand **را در بانک اطلاعاتی اجرا کرده , و سپس داده های برگشتی از اجرای این دستور را درون یک** DataSet **در حافظه قرار دهیم.**

**DataSet : برای نگهداری اطلاعات بدست آمده از بانک اطلاعاتی در حافظه به کار می رود. این کلاس شامل مجموعه ای از جداول , رابطه ها , قید و شرط ها و دیگر مواردی است که از بانک اطلاعاتی خوانده شده است.**

**نمونه کد :**

DataSet dsBook = new DataSet();

**نمونه کد پایانی :**

public partial class frmMain : Form

{

SqlConnection cnn = new SqlConnection(clsMain.cnnStr);

SqlCommand cmd = new SqlCommand();

SqlDataAdapter dadp = new SqlDataAdapter();

DataSet dsBook = new DataSet();

}

private void frmMain\_Load(object sender, EventArgs e)

{

cmd.CommandText = @" SELECT b\_id, b\_number, b\_name, b\_subject,

b\_catNumber, b\_ca\_id, ca\_id

FROM tbl\_book INNER JOIN tbl\_categori ON

b\_ca\_id = ca\_id

WHERE 1=1 ";

cmd.Connection = cnn;

تنظیم شیء دستور برای آداپتر dadp.SelectCommand = cmd; //

اگر اتصال باز نیست if (cnn.State != ConnectionState.Open) //

{

بازکردن شیء اتصال// cnn.Open();

}

ds.Clear();

خواندن اطلاعات از پایگاه داده و قرار دادن آنها در دیتاست// dadp.Fill(ds);

}

* 1. درج اطلاعات در پایگاه داده

**برای دسترسی به لیست پارامترهایی که در یک شئ از کلاس** SqlCommand **وجود دارد می توانیم از خاصیت** Parameters **در این کلاس استفاده کنیم . این خاصیت حاوی لیستی از** Placeholder **ها به همراه متغیرهای وابسته به آنها است بنابراین در کد قبل از اجرای دستور ، باید به وسیله ی این لیست مشخص کنیم که هر** Placeholder **با مقدار چه متغیری باید تعویض شود.**

cmd.Parameters.AddWithValue( "@b\_number" , txtNumber.Text );

cmd.Parameters.AddWithValue("@b\_name", txtName.Text);

cmd.Parameters.AddWithValue("@b\_subject", txtSubject.Text);

cmd.Parameters.AddWithValue("@b\_writer", txtWriter.Text);

**:CommandText خاصیت بعدی که باید از کلاس** SqlCommand **تنظیم شود خاصیت** CommandText **است .این خاصیت متنی را دریافت می کند که می تواند حاوی یک دستور** Sql **و یا فراخوانی یک پروسیجر ذخیره شده در بانک اطلاعاتی باشد که باید روی داده ها اجرا شود .**

**برای درج اطلاعات در بانک از دستور** INSERT **استفاده می کنیم .**

cmd.CommandText = @" INSERT INTO tbl\_book

(b\_number, b\_name, b\_subject, b\_writer)

**Placeholder ها :متغیرهایی هستند که در یک دستور** Sql **قرار می گیرندو می توانند در زمان اجرای برنامه جای خود را با عبارتی خاص عوض کنند این متغیرها با علامت @ در یک دستور مشخص می شوند . و هنگامی که از آنها در یک دستور** Sql **استفاده کنیم قبل از اجرای دستور باید تمامی آنها را با مقادیر مناسب تعویض کنیم . که این کار به صورت اتوماتیک توسط برنامه در زمان اجرای دستور انجام می شود.اما باید پارامترهایی را ایجاد کرده و آن را در لیست** parameters **در شئ ایجاد شده از کلاس** SqlCommand **قرار دهیم تا برنامه بداند هنگام اجرای دستور هر** placeholder **را باید با مقدار چه متغیری در برنامه عوض کند . هیچ ضرورتی ندارد که نام یک** placeholder **همنام فیلدی باشد که قرار است مقدار** placeholder **در آن قرار بگیرد.**

VALUES

(@b\_number, @b\_name, @b\_subject, @b\_writer)

**: ExecuteNonQuery این متد دستورات را بر روی بانک اطلاعات اجرا می کند .  
برای استفاده از این متد باید ابتدا اتصال خود را به بانک اطلاعاتی برقرار کنید سپس با فراخوانی این متد دستور موجود در شئ** Command **را اجرا کنید .**

if (cnn.State != ConnectionState.Open) // اگر اتصال باز نیست

{

cnn.Open(); // بازکردن شیء اتصال

}

cmd.ExecuteNonQuery();

cnn.Close();

MessageBox.Show("درج با موفقیت انجام شد");

* 1. ویرایش اطلاعات

**برای ویرایش سطری از جدول ابتدا باید تمامی اطلاعات مربوط به سطر مورد نظر از روی پایگاه داده لود شود و در** textbox **های مربوط در فرم ویرایش قرار گیرد. برای خواندن اطلاعات از** DataReader **استفاده می کنیم.**

**دستور خواندن اطلاعات مربوط از پایگاه داده در هنگام لود شدن فرم:**

private void FrmBookEdite\_Load\_1(object sender, EventArgs e)

{

cmd.CommandText = @" SELECT b\_number, b\_name, b\_subject, b\_writer

FROM tbl\_book

WHERE b\_id = " + book\_id;

}

**Sql Data Reader : از طریق این کلاس می توانیم اطلاعات را از بانک دریافت کنیم و به صورت زیر تعریف می شود :**

SqlDataReader dr;

**: Execute Reader برای دریافت اطلاعات از بانک از متد** ExecuteReader **استفاده می کنیم ، یک شی از نوع** SqlDataReader **تعریف کرده و مقدار** ExecuteReader **را برابر آن قرار می دهیم.**

dr = cmd.ExecuteReader();

**سپس ازطریق متد** Read **شی** DataReader **اطلاعات را دریک حلقه واکشی می کنیم. یعنی اگر سطری برای خواندن وجود داشت آنهارا به ترتیب در داخل** textbox **قرار می دهد.**

SqlDataReader dr;

if (cnn.State != ConnectionState.Open) // اگر اتصال باز نیست

{

cnn.Open(); // بازکردن شیء اتصال

}

dr = cmd.ExecuteReader();

if (dr.Read())

{

b\_number = txtNumber.Text = dr.GetValue(0).ToString();

txtName.Text = dr.GetValue(1).ToString();

txtSubject.Text = dr.GetValue(2).ToString();

txtWriter.Text = dr.GetValue(3).ToString();

txtYear.Text = dr.GetValue(4).ToString();

txtCover.Text = dr.GetValue(5).ToString();

}

dr.Close();

**برای ویرایش اطلاعات خوانده شده ، از دستور** UPDATE **استفاده می کنیم با استفاده از این دستور مقادیر قبلی را با مقادیر جدید ست می کنیم.**

cmd.Parameters.AddWithValue( "@b\_number" , txtNumber.Text );

cmd.Parameters.AddWithValue("@b\_name", txtName.Text);

cmd.Parameters.AddWithValue("@b\_subject", txtSubject.Text);

cmd.Parameters.AddWithValue("@b\_writer", txtWriter.Text);

cmd.CommandText = @" UPDATE tbl\_book

SET b\_number =@b\_number, b\_name =@b\_name, b\_writer =@b\_writer

WHERE b\_id = " + book\_id ;

* 1. حذف یک سطر از جدول

برای حذف یک سطر از جدول می بایست از دستور DELETE استفاده نمود. برای اینکار ابتدا باید شی دستور را نوشت و به پایگاه داده ارتباط برقرار نمود. و در شی دستور برای حذف آن سطر شرطی قرار خواهیم داد که اگر b\_id (آی دی کتاب) برابر شد با سطر انتخاب شده آنگاه آن سطر را حذف کن.

cmd.Connection = cnn;

dadp.SelectCommand = cmd;

dgvbook.Select();

var yesNo = MessageBox.Show("آیا مطمئن هستید؟ ", " حذف ", MessageBoxButtons.YesNo).ToString();

if (yesNo == "No")

return;

cmd.CommandText = @"DELETE FROM tbl\_book

WHERE b\_id=" + dgvbook.CurrentRow.Cells[0].Value.ToString();

if (cnn.State != ConnectionState.Open)

{

cnn.Open();

}

cmd.Connection = cnn;

cmd.ExecuteNonQuery();

cnn.Close();

MessageBox.Show("کتاب با موفقیت حذف گردید ");

loadInformation();

* 1. حذف یک امانت

برای حذف یک امانت نمی توانیم مانند حذف اطلاعات در بخش 4-7 استفاده نمود. زیرا امانت ها در جدول کتاب یعنی در سطر هرکتاب ثبت خواهد شد و اگر از دستور DELETE استفاده نمائیم تمام سطر آن کتاب حذف خواهد شد. برای اینکار هنگام درج امانت ما از روش ویرایش که در بخش 4-6 آنرا توضیح دادیم استفاده می نمائیم اما مقدار Placeholder ها را برابر DBNull.Value قرار می دهیم. با اینکار بجای حذف سطر ، پایگاه داده ستون های امانت مورد نظر را برابر Null قرار می دهد.

cmd.Parameters.AddWithValue("@b\_c\_id", DBNull.Value);

cmd.Parameters.AddWithValue("@b\_senDay", DBNull.Value);

cmd.Parameters.AddWithValue("@b\_senMonth", DBNull.Value);

cmd.Parameters.AddWithValue("@b\_senYear", DBNull.Value);

cmd.Parameters.AddWithValue("@b\_recDay", DBNull.Value);

cmd.Parameters.AddWithValue("@b\_recMonth", DBNull.Value);

cmd.Parameters.AddWithValue("@b\_recYear", DBNull.Value);

cmd.CommandText = @" UPDATE tbl\_book

SET b\_c\_id =@b\_c\_id, b\_senDay =@b\_senDay,

b\_senMonth =@b\_senMonth, b\_senYear =@b\_senYear,

b\_recDay =@b\_recDay, b\_recMonth =@b\_recMonth,

b\_recYear =@b\_recYear

WHERE b\_id = " + BookId;

* 1. نمایش تاریخ شمسی سیستم

برای نمایش تاریخ شمسی سیستم در فرم ها باید ابتدا در فضای نام ، دستور متناسب با نمایش تاریخ شمسی را بنویسیم . برای اینکار از دستور زیر استفاده می نمائیم :

using System.Globalization;

سپس در تابع لود فرم کد های زیر را قرار می دهیم :

PersianCalendar PC = new PersianCalendar();

lblData.Text = PC.GetYear(DateTime.Now).ToString() + "/" + PC.GetMonth(DateTime.Now).ToString("00")

+ "/" + PC.GetDayOfMonth(DateTime.Now).ToString("00");

lblData نام لیبلی است که می خواهیم تاریخ را در آن به نمایش بگذاریم.

و اگر بخواهیم هر بخش تاریخ را در یک textbox به نمایش بگذاریم از کد زیر استفاده می کنیم مانند فرم درج امانت جدید و یا درج عضو جدید.

PersianCalendar PC = new PersianCalendar();

txtIssYear.Text = PC.GetYear(DateTime.Now).ToString();

txtIssMonth.Text = PC.GetMonth(DateTime.Now).ToString("00");

txtIssDay.Text = PC.GetDayOfMonth(DateTime.Now).ToString("00");

باید دقت داشته باشیم که کد های بالا را در تابع لود فرم قرار دهیم تا هنگام باز شدن فرم بلافاصله تاریخ روز نمایش داده شود.

* 1. جستجو

برای جستجو در یک جدول ، ابتدا باید دستور خواند اطلاعات از پایگاه داده را در یک تابع عمومی قرار دهیم و سپس شرط جستجو را در شی دستور اضافه نمائیم. برای اینکار قبل از تابع لود فرم یک متغیر از نوع string با نام دلخواه (مثال constQuery) می سازیم و دستور را در داخل آن قرار می دهیم مانند کد زیر :

string constQuery= @" SELECT b\_id, b\_number, b\_name, b\_subject,

b\_catNumber, b\_ca\_id, ca\_id

FROM tbl\_book INNER JOIN tbl\_categori ON b\_ca\_id = ca\_id

WHERE 1=1 ";

به دلیل اینکه ما نمی خواهیم در هنگام لود فرم جستجویی انجام دهیم شرط (WHERE) را برابر با 1 قرار داده ایم.

سپس بعد از تابع لود فرم ، تابعی می سازیم به نام loadInformation (نام دلخواه) و این تابع را در تابع لود فرم فراخوانی می کنیم. سپس در خود تابه loadInformation شی دستور را قرار می دهیم و شی دستور را برابر با متغیر constQuery که ساخته ایم قرار می دهیم :

private void loadInformation()

{

cmd.CommandText = constQuery;

}

ما برای تایپ جستجو نیاز به یک textbox داریم تا با تایپ کردن در آن بتوان در جدول مورد نظر عملیات جستجو را انجام داد که باید در تابع loadInformation آن textbox شرط جستجو را به شی دستور اضافه نمائیم مانند کد زیر :

private void loadInformation()

{

cmd.CommandText = constQuery;

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* قرار دادن شرط برای جستجو در تکست باکس

if (txtName.Text != string.Empty)

{

cmd.CommandText += " AND b\_name LIKE '%" + txtName.Text + "%'";

}

همانطور که مشاهده می نمائید ابتدا با شرط if می گوئیم اگر textbox خالی نباشد (یعنی عبارتی در آن تایپ شده باشد) شرط جستجو را به دستور شی اضافه کن. برای شرط جستجو باید از LIKE استفاده نمود.

در شرط جستجوی بالا می گوئیم اگر ستون name کتاب عبارتی در آن وجود داشت که در textbox تایپ شده باشد شرط را به شی دستور اضافه کن.

کار تمام است و فقط باید در رویداد TextChanged آن textbox تابع loadInformation را فرا خوانی نمود مانند کد زیر :

private void txtName\_TextChanged(object sender, EventArgs e)

{

loadInformation();

}

نکته : رویاد TextChanged را بعد از تابع لود فرم قرار دهید.

* 1. درج عکس

برای درج تصاویر در برنامه ابتدا باید یک PictureBox در فرم قرار بدهیم و یک دکمه (Button) و در رویداد Click دکمه کد زیر را قرار می دهیم :

Bitmap img = null;

private void btnPic\_Click(object sender, EventArgs e)

{

{

openFileDialogImage.Filter = "All Image Formats

(\*.bmp;\*.jpg;\*.jpeg;\*.png;\*.gif;\*.tif)|" +

"\*.bmp;\*.jpg;\*.jpeg;\*.gif;\*.tif|Bitmaps

(\*.bmp)|\*.bmp|PNGs (\*.png)|\*.png|" +

"GIFs (\*.gif)|\*.gif|JPEGs (\*.jpg)|\*.jpg;\*.jpeg|TIFs

(\*.tif)|\*.tif";

openFileDialogImage.FilterIndex = 1;

openFileDialogImage.FileName = string.Empty;

if (openFileDialogImage.ShowDialog() == DialogResult.OK)

{

PixBox.Image =

System.Drawing.Image.FromFile(openFileDialogImage.FileName);

FileInfo fi = new

FileInfo(openFileDialogImage.FileName);

string temp = fi.Name;

if (fi.Name != null)

{

img = (Bitmap)Image.FromFile(openFileDialogImage.FileName);

}

}

}

برای درج تصویر در پایگاه داده نیز باید از کد زیر در هنگام درج استفاده نمود :

byte[] bufferFront = null;

System.IO.MemoryStream msFron = new System.IO.MemoryStream();

img.Save(msFron, System.Drawing.Imaging.ImageFormat.Jpeg);

bufferFront = msFron.GetBuffer();

cmd.Parameters.AddWithValue("@c\_image", bufferFront);

* 1. بررسی خالی نبودن textbox

در برنامه کتابخانه بعضی از فیلد ها نباید خالی باشند. به منظور بررسی خالی نبودن این فیلد ها از شرط if به شکل زیر استفاده می کنیم :

if (txtNumber.Text == string.Empty) // خالی بود txtNumber اگر

{

txtNumber.BackColor = Color.Red; // txtNumber رنگی کردن

MessageBox.Show(" لطفاً شماره کتاب را وارد نمائید ");

txtNumber.Select(); // txtNumber انتخاب کردن

return;

}

* 1. بررسی درج عدد صحیح در textbox

در برنامه کتابخانه در بعضی از فیلد ها حروف درج شوند و باید بجای حروف اعداد درج شوند. به همین منظور از بلوک try / catch به شکل زیر استفاده می کنیم :

try

{

int temp = Convert.ToInt32(txtNumber.Text);

}

catch

{

txtNumber.BackColor = Color.Red;

MessageBox.Show("لطفاً شماره کتاب را درست وارد نمائید");

txtNumber.Text = string.Empty;

txtNumber.Select();

return;

}

دربخشtry ابتدا یک متغیر از نوع صحیح تعریف کرده ایم و این دستور می گوید که سعی کن مقداری که کاربر وارد کرده در داخل txtNumber را تبدیل کن به عدد صحیح ، اگر خطا ندهد برنامه مشکلی ندارد و اگر خطا بدهد برنامه به قسمت catch می رود و در آن قسمت پیام داده می شود که لطفا شماره کتاب را درست وارد نمائید.

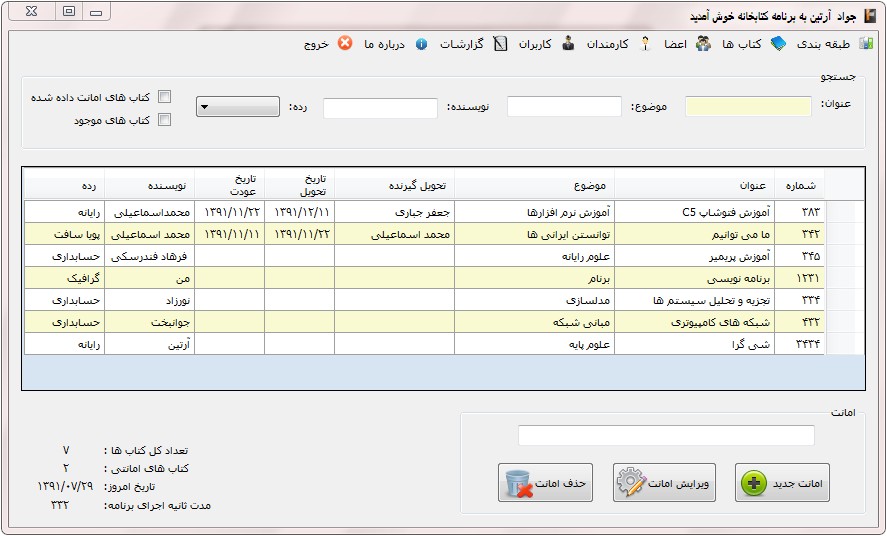
1. راهنمای برنامه
   1. شروع کار برنامه

ابتدا بر روی آیکون برنامه دابل کلیک کرده تا پنجره فرم لاگین باز شود. سپس با وارد کردن نام کاربری و کلمه عبور و کلیک دکمه ورود و یا فشردن دکمه Enter وارد برنامه شوید. ( نام کاربری admin – رمز عبور 1)

****

(تصویر 5-1-1)

**سپس فرم اصلی برنامه باز خواهد شد. در این فرم یک منو در بالا برای دستری به فرم های دیگر و نیز همان منو توسط یک آیکون در کنار ساعت ویندوز قرار گرفته.**

****

(تصویر 5-1-2)

pic_03.jpg

(تصویر 5-1-3)

pic_04.jpg 

(تصویر 5-1-4)

**در قسمت جستجوی فرم اصلی شما می توانید کتاب های مورد نظر خود را بر اساس عنوان کتاب ، موضوع ، نویسنده ، رده ، کتاب های امانت داده شده و کتاب های موجود فیلتر نمائید.**

****

(تصویر 5-1-5)

**در قسمت وسط فرم شما یک جدول مشاهده می نمائید که در آن لیست کتاب ها وجود دارد. این جدول براساس کتاب های امانت داد شده ، نام ، نام خانوادگی ، تاریخ تحویل و تاریخ عودت مرتب شده است.**

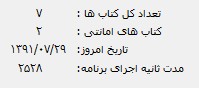
(تصویر 5-1-6)

**در پایین فرم سمت راست قسمت امانت وجود دارد که می توانید در این قسمت برای یک کتاب امانت جدید ثبت کنید و یا اینکه آن امانت را ویرایش و یا حذف نمائید که بعد به آن می پردازیم.**

****

(تصویر 5-1-7)

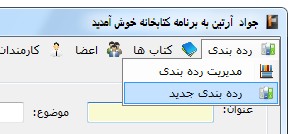
**و در پایین فرم سمت چپ می توانید تعداد کتاب های کتابخانه ، تعداد کتاب های امانت داده شده ، تاریخ روز و مدت ثانیه اجرای برنامه را مشاهده نمائید.**

****

(تصویر 5-1-8)

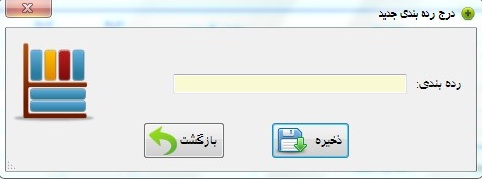
* 1. شروع ثبت کتاب
     1. ثبت رده بندی

**برای ثبت کتاب ابتدا باید یک رده برای کتاب به ثبت برسانید. رده مشخص می کند که کتاب در چه بخش و قفسه ای باید قرار بگیرد. برای اینکار از منوی برنامه قسمت رده بندی را کلیک کرده و در زیر منوی باز شده یا بر روی مدیریت رده بندی و یا بر روی رده بندی جدید کلیک کنید.**

****

(تصویر 5-2-1-1)

**در پنجره باز شده نام رده مورد نظر خود را تایپ و سپس دکمه ذخیره را کلیک نمائید تا رده مورد نظر در پایگاه داده ذخیره گردد.**

****

(تصویر 5-2-1-2)

**در فرم مدیریت رده بندی نیز می توانید لیست رده هایی که به ثبت رسیده اند را مشاهده یا ویرایش یا یک رده جدید به ثبت برسانید و یا نام رده مورد نظر را جستجو نمائید.**

****

(تصویر 5-2-1-3)

* + 1. ویرایش رده

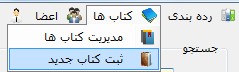
برای ویرایش رده کافیست در فرم مدیریت رده ها ، رده مورد نظر را از جدول انتخاب کرده و دکمه ویرایش را کلیک نمائید. همچنین با دابل کلیک بر روی رده مورد نظر در جدول نیز می توانید به فرم ویرایش بروید.



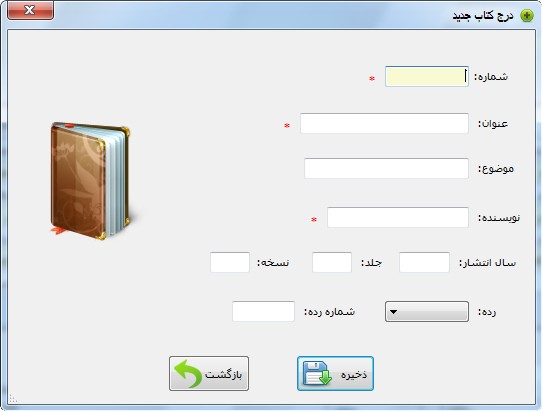
(تصویر 5-2-2-1)

* + 1. ثبت کتاب

پس از ثبت رده نوبت به ثبت کتاب می رسد. برای اینکار از منوی برنامه گزینه کتاب ها و گزینه کتاب جدید را انتخاب نمائید تا فرم ثبت کتاب جدید باز شود.



(تصویر 5-2-3-1)



(تصویر 5-2-3-2)

در فرم ثبت کتاب جدید می بایست گزینه هایی که با علامت ستاره (🞹) مشخص شده اند را پر نمود در غیر اینصورت برنامه به شما اجازه ذخیره کتاب را نخواهد داد.( به عنوان نمونه تصویر شماره 5-2-3-3)



(تصویر 5-2-3-3)

برای مشاهده کتاب ثبت شده و همچنین مدیریت کتاب ها از منوی برنامه گزینه مدیریت کتاب ها را انتخاب نمائید. در فرم باز شده شما می توانید کتاب مورد نظر را بر اساس عنوان کتاب ، موضوع ، نویسنده و رده جستجو نمائید.



(تصویر 5-2-3-4)

* + 1. ویرایش کتاب

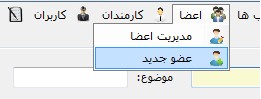
برای ویراش کتاب در فرم مدیریت کتاب کافیست کتاب مورد نظر را انتخاب و دکمه ویرایش را کلیک نماید یا بر روی کتاب مورد نظر دابل کلیک نمائید تا فرم ویرایش باز شود.



(تصویر 5-2-4-1)

* 1. ثبت اعضای کتابخانه

برای ثبت عضو جدید در کتابخانه می بایست از منوی برنامه گزینه اعضا و سپس زیرمنوی عضو جدید را انتخاب کرد تا فرم ثبت عضو جدید باز شود.



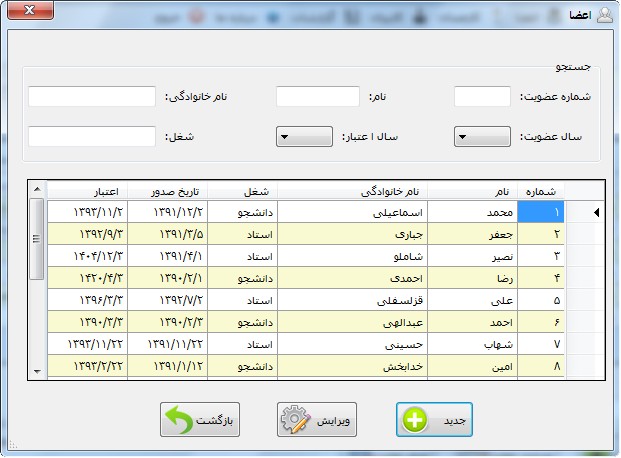
(تصویر 5-3-1)

سپس در فرم باز شده تمامی موارد خواسته شده را پر نمائید. در غیر اینصورت برنامه اجازه ثبت عضو را نخواهد داد. در هنگام وارد نمودن شماره عضو به آخرین شماره عضو در پایین فرم دقت نمائید و عدد بعد از آن را وارد نمائید. در این فرم تاریخ روز عضویت به صورت خودکار توسط برنامه نمایش داده می شود. اگر سال پایان اعتبار کوچکتر از سال صدور باشد برنامه به شما پیغام خطا می دهد. شما می بایست برای هر عضو یک عکس انتخاب نمائید و دقت نمائید عکس انتخابی در هنگام صدور کارت عضویت چاپ می شود.



(تصویر 5-3-2)

برای مشاهده اعضای عضو شده و همچنین مدیریت اعضا از منوی برنامه گزینه مدیریت اعضا را انتخاب می نمائید. در این فرم شما می توانید اعضارا بر اساس شماره عضویت ، نام ، نام خانوادگی ، سال عضویت ،‌ سال اعتبار و شغل جستجو نمائید.



(تصویر 5-3-3)

* 1. ویرایش اعضای کتابخانه

برای ویرایش اعضا کافیست در فرم مدیریت اعضا ، عضو مورد نظر را انتخاب کرده و بر روی دکمه ویرایش کلیک نمائید یا بر روی عضو مورد نظر در جدول دابل کلیک نمائید. در فرم ویرایش نیز باید مواردی که در فرم ثبت عضو گفته شد رعایت شود در غیر اینصورت برنامه به شما اجازه ویرایش نخواهد داد. اما نکته دیگری که در فرم ویرایش وجود دارد امکان چاپ کارت عضویت برای اعضا می باشد. شما با کلیک بر روی دکمه پرینتر می توانید پیش نمایش کارت عضویت را مشاهده نمائید و در صورت نیاز آنرا چاپ بگیرید.



(تصویر 5-4-1)

* 1. چاپ کارت عضویت

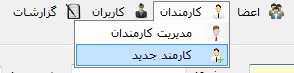
برای چاپ کارت عوضیت کافیست در فرم ویرایش بر روی دکمه پرینتر کلیک نمائید تا پنجره پیش نمایش چاپ باز شود و پس از مشاهده کارت می توانید از آن پرینت بگیرید.



(تصویر 5-5-1)

* 1. ثبت کارمندان

برای ثبت کارمند می بایست از منوی برنامه گزینه کارمندان و زیر منوی کارمند جدید را انتخاب نمائید و در فرم باز شده فیلدهای حاوی (🞹) را پر نمائید.



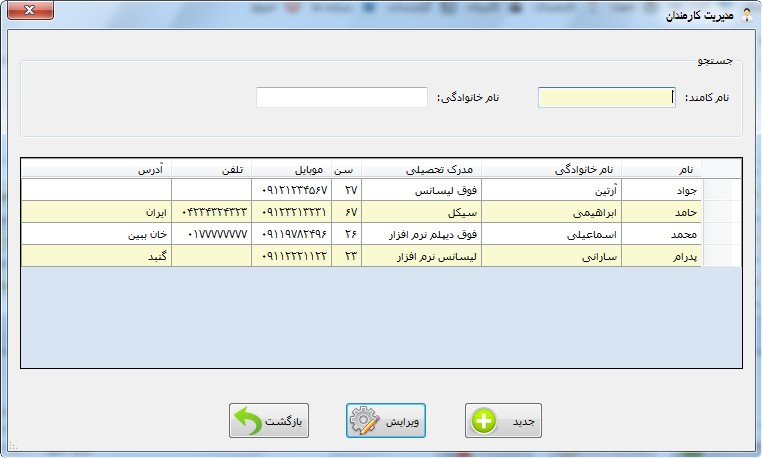
(تصویر 5-6-1)



(تصویر 5-6-2)

در هنگام تایپ شماره تلفن همراه باید دقت شود که شماره وارد شده متعلق به کارمندان دیگر نباشد ، زیرا برنامه اجازه وارد نمودن شماره همراه تکراری را به شما نخواهد داد.

برای مشاهده کارمندان ثبت شده و همچنین مدیریت کارمندان از منوی برنامه گزینه کارمندان و زیر منوی مدیریت کارمندان را انتخاب نمائید. در فرم باز شده شما می توانید کارمند را بر اساس نام و نام خانوادگی جستجو نمائید.



(تصویر 5-6-3)

* 1. ویرایش کارمندان

برای ویرایش کافیس کارمند مورد نظر را انتخاب و دکمه ویرایش را کلیک یا بر روی کارمند مورد نظر در جدول دابل کلیک نمائید.



(تصویر 5-7-1)

* 1. ثبت کاربران

برای ثبت کاربر از منوی برنامه گزینه کاربران و زیر منوی کابر جدید را انتخاب نمائید. در فرم باز شده ضمن انتخاب یک کاربر ، تمام فیلد ها را باید پر نمائید. دقت کنید اگر کلمه عبور اول و دوم باهم متفاوت باشند برنامه به شما اجازه ذخیره نخواهد داد.



(تصویر 5-8-1)

برای مشاهده کاربران و مدیریت کاربران از منوی برنامه گزینه کاربران و سپس زیرمنوی مدیریت کاربران را انتخاب نمائید.



(تصویر 5-8-2)

* 1. ویرایش کاربران

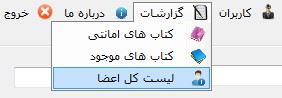
برای ویرایش کاربران کافیست در فرم مدیریت کاربران ، کاربر مورد نظر را انتخاب و بر روی دکمه ویرایش کلیک یا بر روی کاربر مورد نظر در جدول دابل کلیک نمائید.



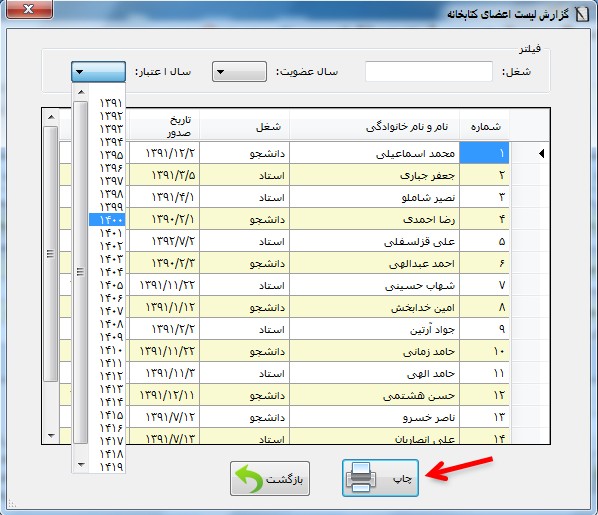
(تصویر 5-9-1)

* 1. گزارش گیری

برنامه کتابخانه طراحی شده این امکان را در خود جای داده است که بتوان از لیست کتاب های امانت داده شده ، لیست کتاب های موجود و لیست اعضا با فیلتر کردن شغل اعضا و یا سال عضویت و سال اعتبار گزارش تهیه نماید و از گزارش تهیه شده چاپ نیز بگیرد. برای اینکار از منوی برنامه گزینه گزارشات را انتخاب و از زیر منو ها گزارش مورد نظر را انتخاب نمائید.



(تصویر 5-10-1)



(تصویر 5-10-2)

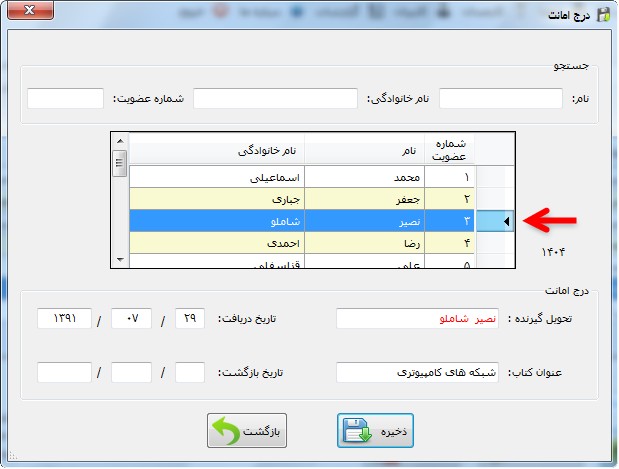
* 1. ثبت امانت

برای ثبت امانت می بایست در جدول فرم اصلی بر روی نام کتابی که قبلا امانت داده نشده است دابل کلیک کرده و بعد از نمایان شدن نام کتاب در TextBox بخش امانت ، بر روی دکمه امانت جدید کلیک کرده تا فرم ثبت امانت باز شود.



(تصویر 5-11-1)

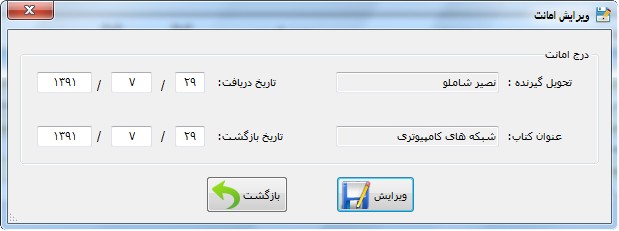
درفرم باز شده می توانید عضوی که میخواهید کتاب را به او امانت دهید را از طریق نام ، نام خانوادگی و شماره عضویت جستجو نمائید. سپس بر روی نام عضو مورد نظر دابل کلیک کرده تا در TextBox تحویل گیرنده نمایش داده شود. همانطور که مشاهده می نمائید نام کتاب و تاریخ روز تحویل قبلا در فرم توسط برنامه ثبت شده است و شما فقط می بایست تاریخ برگشت کتاب را وارد نمائید. اما اگر تاریخ بازگشت کتاب کوچکتر از تاریخ تحویل باشد برنامه به شما اجازه نخواهد داد که امانت را ثبت نمائید. درصورت عدم انتخاب عضو نیز برنامه به شما اجازه نخواهد داد که شما امانت را ثبت نمائید و باید حتما عضو مشخص شده باشد.



(تصویر 5-11-2)

* 1. ویرایش امانت

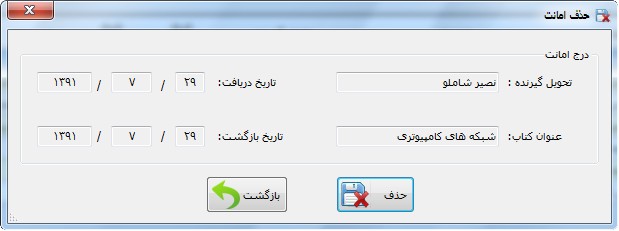
برای ویرایش امانت کافیست از جدول فرم اصلی کتابی که امانت داده شده است را انتخاب نمائید تا نام کتاب در TextBox قسمت امانت درج شود و سپس بر روی دکمه ویرایش کلیک نمائید تا فرم ویرایش باز شود. آنگاه می توانید تاریخ ها را ویرایش نمائید.



(تصویر5-12-1)

* 1. حذف امانت

برای حذف یک امانت کافیست در جدول فرم اصلی بر روی کتابی که امانت داده شده است دابل کلیک نمائید تا نام کتاب در TextBox قسمت امانت نمایش داده شود. سپس بر روی دکمه حذف امانت کلیک کرده تا فرم حذف امانت باز شود. و بعد از کلیک بر روی دکمه حذف امانت مورد نظر حذف خواهد شد.



(تصویر5-13-1)

نتیجه گیری :

با استفاده از برنامه کتابخانه ضمن داشتن آرشیوی کامل و کار آمد با ماندگاری بالاتر از دفاتر ثبت ، سرعت عمل بالاتری نیز در بر خواهد داشت. نمایش کتاب های امانت داده شده در ابتدای ورود به نرم افزار می تواند به مدیریت بهتر کتاب های امانت داده شده کمک بسیار فراوانی کند. با استفاده از این برنامه خطاهای موجود بسیار کاهش می یابد. عضو گیری با دقت بهتر و سریعتری شکل می گیرد و کارت عضویت در کتابخانه در عرض کمتر از یک دقیقه با کیفیت بسیار مطلوب حاضر می شود و در خود برنامه نیز ثبت می گردد. قبلا گرفتن گزارش از تعداد کتاب های امانت داده شده و یا کتاب های موجود و یا اعضا عمری بسیار دشوار و گاهاً غیر ممکن بود اما با این برنامه تمام این موارد با انعطاف پذیری بالای امکان پذیر است.

تمام این مزیت ها روشنگر این هست که استفاده از این دست برنامه ها می تواند کمک بسیار زیادی به کتابدارن نماید.

پیشنهاد :

می توان با اتصال این برنامه با یک برنامه تحت وب ، عملیات جستجوی کتاب ، موجود بودن کتاب مورد نظر در کتابخانه ، رزرو کتاب ، درخواست عضویت ، درخواست تمدید عضویت و ارتباط با کتابدار را فراهم نمود.

فهرست منابع