**فهرست مطالب**

*عنوان صفحه*

[چکیده 2](#_Toc436945074)

[مقدمه 3](#_Toc436945075)

[فصل اول 4](#_Toc436945076)

[معرفی نرم افزار و زبان های برنامه نویسی 4](#_Toc436945077)

[1-1نرم افزار استفاده شده 5](#_Toc436945078)

[2-1زبانهای برنامه نویسی 6](#_Toc436945079)

[1-2-1 روش‌های برنامه نویسی 6](#_Toc436945080)

[2-2-1 انواع سطح زبان ها 7](#_Toc436945081)

[3-2-1 نوع ترجمه 7](#_Toc436945082)

[فصل دوم 9](#_Toc436945085)

[ویژگی سی شارپ و پروژه 9](#_Toc436945086)

[1-2زبان سی شارپ 10](#_Toc436945087)

[1-1-2 اهداف طراحی زبان 10](#_Toc436945088)

[2-1-2ويژگي‌ها 10](#_Toc436945089)

[فصل سوم 12](#_Toc436945091)

[کدهای استفاده شده در برنامه و عکس ها 12](#_Toc436945092)

[1-3 لیست جداول پایگاه داده و ارتباط بین انها 13](#_Toc436945093)

[تصاویرقسمت های مختلف برنامه و توضیحات 13](#_Toc436945094)

[سورس کدبرنامه و توضیحات 20](#_Toc436945095)

[4-3 نتیجه گیری 35](#_Toc436945096)

**چکیده**

این پروژه راجع به شرکت پخش مواد غذایی یا سوپرمارکت می باشد. میتوان انواع اعمال اصلی مانند حذف و اضافه و نمایش فاکتور را درون این پروژه مشاهده کرد!

همچنین اتصال صحیح ایجاد شده بین این پروژه و فایل های تکست (.txt) باعث شده تا بتوان اطلاعات را درون برنامه ثبت و بازیابی کرد.

**مقدمه**

دنیاي برنامه نویسی دنیاي عجیبی است، برنامه نویسی به شما اجازه می دهد که با کامپیوترتان مانند یک رئیس رفتار کنید، برنامه نویسی می تواند شغل یا تفریح شما باشد اما هدف اصلی از برنامه نویسی ارائه راه حلی براي بر طرف کردن مشکلات مختلف است.

برنامه نویسی هنر برطرف کردن مشکلات با استفاده از کد است!

ما می توانیم با استفاده از زبانهاي برنامه نویسی برنامه بنویسیم، زبان هاي برنامه نویسی زیادي وجود دارند که از این بین چندین زبان از بقیه مشهورترو پرکاربرد تر می باشند. شاید یکی از بهترین زبان هاي برنامه نویسی زبان سی شارپ (C#) باشد.

**فصل اول**

**معرفی نرم افزار و زبان های برنامه نویسی**

**1-1 نرم افزار استفاده شده**

نرم افزار استفاده شده در این پروژه نرم افزار Microsoft Visual Studio است که این نرم‌افزار Visual Studio، نرم‌افزاری توسعه یافته برای برنامه نویسان کامپیوتر است که توسط شرکت نرم‌افزاری [Microsoft](http://fa.wikipedia.org/wiki/%D9%85%D8%A7%DB%8C%DA%A9%D8%B1%D9%88%D8%B3%D8%A7%D9%81%D8%AA) تولید شده است . تمرکز اصلی این نرم‌افزار از اولین نسخه‌های آن تا کنون بر روی خصوصیت [IDE](http://fa.wikipedia.org/wiki/%D9%85%D8%AD%DB%8C%D8%B7_%DB%8C%DA%A9%D9%BE%D8%A7%D8%B1%DA%86%D9%87_%D8%AA%D9%88%D8%B3%D8%B9%D9%87_%D9%86%D8%B1%D9%85%E2%80%8C%D8%A7%D9%81%D8%B2%D8%A7%D8%B1) بودن آن است که به برنامه نویس اجازه می‌دهد تا برنامه‌های کاربردی مستقل، وب‌گاه، برنامه‌های کاربردی وب و یا سرویس‌های وب را که بر روی تعدادی از PlatFormهای پشتیبانی شده توسط Microsoft .NET Framework(البته برای تمام نسخه‌های بعد از ویژوال استودیو 6) همچنین Platform هایی مانند Microsoft Windows servers and workstations, PocketPC Smartphone و World Wide Web browsers اجرا می‌شوند را براحتی ایجاد نماید.

Visual Studio یک مجموعه از برنامه‌هایی است که ارتباط بسیار نزدیک با هم دارند که مایکروسافت آن را به توسعه دهندگان و برنامه نویسان برنامه‌های کاربردی اهدا نمود تا آنها را وادار نماید در محیطی توسعه یافته بر روی پلت فرم‌های ویندوز و دات نت به ساخت برنامه‌های خود بپردازند ، ویژال استدیو می‌تواند برای نوشتن برنامه‌های کنسولی، ویندوزی، سرویس‌های ویندوز، برنامه‌های کاربردی موبایل، برنامه‌های کاربردی ASP.NET و سرویسهای وب ASP.NETبنا به انتخاب شما همراه با زبانهایی مانند #C++, C#, VB.NET, J استفاده شود .

*با Visual Studio واقعاً چه کارهایی می‌توان انجام داد ؟ در زیر تعدادی از کاربردهایی را که برای تولید آنها می‌توان از Visual Studio استفاده نمود معرفی گردیده‌اند:*

 : Console applicationاین کاربرد برای اجرای خطوط دستور البته بدون محیط گرافیکی استفاده می‌شود که از این کاربرد برای برخی از ابزارهای کوچک یا برای اجرا شدن کدها توسط دیگر کاربردها استفاده می‌شود، این دستور ها در خط فرمان اجرا میشود.

:Windows forms applications برای برنامه‌های کاربردی ویندوزی که با استفاده از NET framework. نوشته می‌شوند .

:Windows servicesسرویس‌ها برنامه‌های کاربردی هستند که در پس زمینه ویندوز اجرا می‌شوند .

(ASP.NET applications)  : ASP.NET یک تکنولوژی قدرتمند که برای طراحی و ساخت صفحات وب پویا استفاده می‌شود .

: ASP.NET web service مدل سرویسهای وب را بطور کامل فراهم نموده تا شما براحتی و با سرعت سرویسهای وب را تولید نمایید .

:Windows Mobile applicationsکه می‌تواند بر روی ابزارهایی که شامل framework هستند مانند Pocket PCها و همچنین cell phones هایی که پلت فرم Microsoft Smartphone بر روی آنها اجرا می‌شود، اجرا گردد .

MFC/ATL/Win32 applications: شماهمچنان می‌توانید برنامه‌های سنتی MFC ، ATLیا برنامه‌های Win32را با استفاده از C++ ایجاد نمایید . این برنامه‌های برای اجرا به NET framework. نیاز ندارند اما نمی‌توانند از مزایای NET framework. نیز بهره‌ای ببرند .

:Visual Studio add-insشما می‌توانید از خود ویژوال استودیو برای ساخت توابعی جدید و قابل اضافه شدن به خود ویژوال استودیو استفاده نمایید .

کاربردهای دیگر : ویژوال استودیو همچنین شامل پروژه‌هایی برای توسعه برنامه‌های کاربردی شما، کار با databases، ساخت گزارشها و ... می‌باشد .

**2-1 زبانهای برنامه نویسی**

در حال حاضر زبان های برنامه نویسی متعددی وجود دارند و توسعه دهندگان نرم افزار در سراسر جهان با بهره گیری از آنها برنامه های خود را توسعه می دهند.

زبان‌های برنامه نویسی را می‌توان از پنج دیدگاه متفاوت مورد بررسی قرار داده و تقسیم­بندی کرد:

**1-2-1 روش‌های برنامه نویسی**

1-زیر روالی (Subroutine) یا تابعی (Functional) : روال یا تابع (به آن زیر برنامه هم می گویند) یک قطعه از برنامه است که دارای نام مشخص می باشد و در برنامه اصلی با فراخوانی نام آن، می توان آن را اجرا کرد، برنامه نویسان به کمک این روش علاوه بر افزایش خوانایی برنامه، می توانند از یک قطعه کد بارها استفاده کنند.

2- ساخت یافته : علاوه بر مزایای روش تابعی، کد را در قالب بلوک (Block) می نویسند، هر بلوک نماینگر قسمتی از کد برنامه است که وظایف مشخصی دارد.

3- مدولار یا ماژولار(Modular) : در این روش از ماژول ها استفاده می شود. ماژول قسمتی از کد برنامه است که در یک فایل مجزای دیگر نوشته می شود، در ماژول معمولاً توابع و متغیرهایی (Variables) ایجاد می شوند که در تمامی برنامه عمومیت داشته باشند.

4- شئ گرا (Object Oriented) : به روز ترین روش برنامه نویسی است که در آن همه چیز به دید شیء (Object) نگریسته می شود. شیء از یک کلاس (Class) مشتق می شود، برنامه نویسان با شناسایی مسئله و انتزاع موجودیت های (Entities) آن، کلاس ها را طراحی می کنند. کلاس مجموعه ای از خواص (Properties) ، متدها (Method) و رویدادها (Events) می باشد که یک بار طراحی می شود اما می توان چندین نمونه (Instance) از آن ایجاد کرد.

در حال حاضر روش مورد استفاده برنامه نویسان روش شیء گرایی است!

**2-2-1 انواع سطح زبان ها**

1- سطح پایین : زبان های سطح پایین به زبان ماشین نزدیک هستند و معمولاً در حوزه علم سخت افزار بکار برده می شوند، دستورات این نوع زبان ها بسیار ابتدایی اما یادگیری آن ها مشکل است.

2-سطح میانی : در این نوع زبان ها از دستورات قابل فهم تری نسبت به زبان های سطح پایین استفاده شده است و بیشتر برای کنترل سخت افزار از آنها استفاده می شود، در برنامه نویسی کامپایلرها و سیستم عامل ها به شکل گسترده ای از این نوع زبان ها استفاده شده است. دستورات این نوع زبان ها تا حدودی ابتدایی اما یادگیری آن ها معمولاً مشکل است.

3- سطح بالا : برای برنامه نویسی برنامه های کاربردی مختلف از این نوع زبان های استفاده می شود، قسمت عمده ای از زبان های برنامه نویسی موجود در دنیا در این گروه قرار دارند. دستورات این نوع زبان ها مرکب و دارای پیچیدگی می باشند اما یادگیری آن ها ساده است.

**3-2-1 نوع ترجمه**

1- مفسری : زبان هایی که در حین اجرا خط به خط ترجمه و تفسیر می شوند زبان های مفسری نام دارند. در این نوع زبان ها امکان اشکال زدایی تمامی دستورات برنامه وجود ندارد و اشکالات در حین اجرا مشخص می شوند، همچنین سرعت اجرای آنها پایین است زیرا همزمان با عمل ترجمه و تفسیر دستورات، عمل اجراء نیز انجام می شود.

2-کامپایلری : زبان هایی که ابتدا کل دستورات برنامه را بررسی کرده و اشکالات نحوی(Syntax Error) آنها را شناسایی می کنند سپس آنها را ترجمه می کنند، زبان های کامپایلری نامیده می شوند.

**فصل دوم**

**ویژگی سی شارپ و پروژه**

**1-2زبان سی شارپ**

**1-1-2 اهداف طراحی زبان**

استاندارد ECMA اين اهداف طراحي زبان را براي #C بر آورده مي سازد: سي شارپ يک زبان برنامه سازي ساده، مدرن، براي اهداف عمومي و شي گرا است. به دليل اهميت داشتن موضوع نيرومندي و دوام و بهره وري برنامه نويس، زبان داراي چک کننده Strong Type ، چک کننده مرزهاي آرايه، تشخيص حالتهايي که يک متغير مقدار دهي اوليه نشده‌است، قابليت انتقال کدها و Garbage Collection خودکار است. اين زبان براي استفاده در اجزاي توسعه نرم افزار براي دستيابي به مزاياي سيستم‌هاي توزيعي در نظر گرفته شده‌است و قابليت انتقال برنامه نويس بسيار مهم است،خصوصاً براي آن دسته از برنامه نويساني که با زبان‌هاي C و C++ آشنا هستند. پشتيباني از اين زبان براي بين المللي شدن بسيار مهم است. زبان سي شارپ براي نوشتن برنامه‌ها براي سيستم هاي تعبيه شده و ميزبان در نظر گرفته شده‌است، از سيستم عامل هاي پيچيده بسيار بزرگ گرفته تا توابع اختصاصي بسيار کوچک هر چند برنامه هاي نوشته شده با #C طوري هستند که از لحاظ حافظه و پردازنده مورد نياز مقرون به صرفه باشند، ولي خود زبان از لحاظ اندازه و کارايي به خوبي زبان هاي C و اسمبلي نيست.

**2-1-2ويژگي‌ها**

برخي از تفاوت‌هاي زبان سي شارپ با C و ++C عبارتند از: هيچ تابع يا متغير سراسري(Global) وجود ندارد، تمام متدها و اعضا بايستي در داخل کلاس‌ها تعريف شوند. اين امر ممکن است، هر چند براي استفاده از متغير ها و توابع عمومي بايد از متدها و متغيرها در کلاس هاي عمومي استفاده کرد. متغير هاي عمومي، بر خلاف C و ++C ، نمي توانند بلاک هاي پيوستي را در بر بگيرند.سي شارپ داراي يک نوع داده بولي است (bool)، برخي از عبارت‌ها مانند while و if که شرطي هستند، نيازمند يک عبارت نوع بولي هستند. همان طور که ++C نيز داراي نوع داده بولي است، اين نوع داده به راحتي ميتواند به يا از Integerها تبديل شود.کامپايلر سي شارپ برنامه نويس را در اين شرايط مجبور به استفاده از عباراتي مي‌کند که به درستي يک مقدار bool را برميگردانند. بنابراين دستوري مانند if(a = b) باعث بروز خطا مي‌شوند.(به جاي = بايستي از == استفاده شود.در سي شارپ ، اشاره گرهاي به حافظه بايستي فقط در داخل بلوکهاي unsafe استفاده شوند و برنامه در اين حالت براي اجرا نياز به اجازه از کاربر دارد. بيشتر دسترسي شي از طريق شي امن است که يا هميشه در حال اشاره به شي صحيح موجود است يا يک مقدار Null دارد. اشاره گري به شي به درد نخور يا بلاک حافظه رندم غير ممکن است.اشاره گر نا امن مي تواند به نمونه اي از value-type ، آرايه، رشته يا بلاکي که حافظه به آن داده شده است اشاره نمايد.کدي که به عنوان نا امن علامت نخورده باشد، هنوز مي تواند اشاره گر ها را از سيستم بازيابي يا در آن ذخيره کند ولي نمي تواند مرجع جديدي به آنها اختصاص دهد.حافظه ساماندهي شده نمي تواند صريحا آزاد شود، ولي به طور خودکار به عنوان به درد نخور تلقي مي شود.انتخاب آدرس هاي به درد نخور حافظه نفوذ ناپذير است. هم چنين #C با استفاده از عبارات، پشتيباني مستقيمي از پايان اجباري مي کند.

وراثت چندگانه از کلاس‌ها در اين زبان پشتيباني نمي‌شود. سي شارپ بسيار typesafe تر از C++ است. تنها تبديلات ضمني مثل تبديل نوع داده کوچکتر به بزرگتر يا تبديل نوع مشتق شده به نوع پايه به طور پيش فرض و بدون خطا صورت مي‌پذيرد. هيچ تبديل ضمني اي ميانBoolean ها و Integerها وجود ندارد و هر تبديل user-defined بايستي به صراحت با يکي از کلمات explicit يا implicit نشانه گذاري شود. تبديل b به a در حالتي که a يک Integer و b يک double باشد در زبان C++ مجاز است اما در سي شارپ به يک خطاي زمان کامپايل منجر مي‌شود(بايستي به صورت explicit تعريف شود) اعضاي Enumeration در داخل محدوده شخصي خود قرار دارند. c#قابليت syntactic sugar را براي توابع متداول، اکسسور ها و ماجول هاي کسول شده در يک کلاس به صورت ويژگي ها قرار داده است. اکسسورها که خاصيت نيز گفته مي‌شوند در زبان سي شارپ قادر به کنترل دسترسي اعضا و معتبرسازي داده‌ها هستند.تمام انواع بازتابي(Reflection) و بازيابي(Recovery) قابل استفاده‌است.

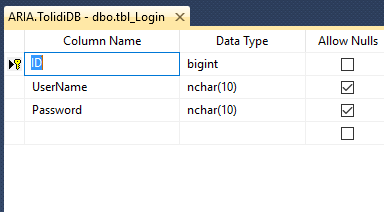
**فصل سوم**

**کدهای استفاده شده در برنامه و عکس ها**

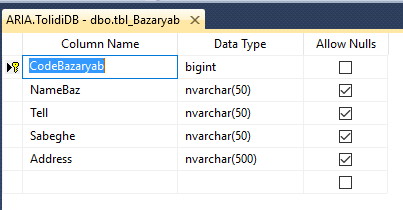
1-3 لیست جداول پایگاه داده و ارتباط بین انها

این پروژه شامل 7 جدول میباشد که به شرح زیر است

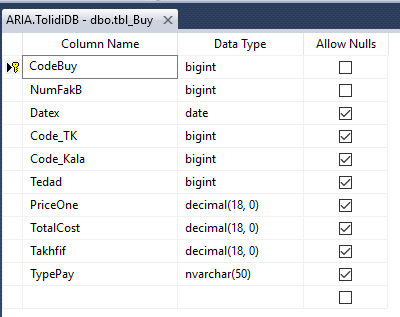
Login.1: شامل نام کاربری و رمز عبور به هر بخش:



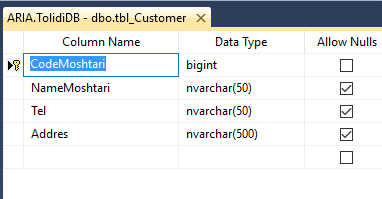
2. tbl\_Bazaryab : در این جدول لیست بازاریابان قرار میگیرد.



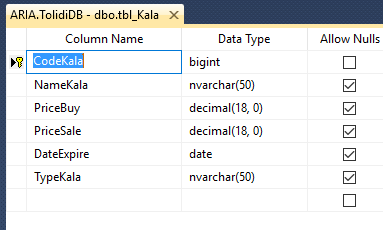
3. tbl\_Buy : در این جدول فاکتورهای خرید را ثبت میکنیم:



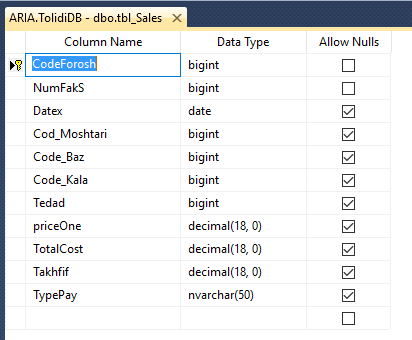
4. tbl\_Customer: در این جدول مشخصات مشتری ها را ثبت میکنیم:



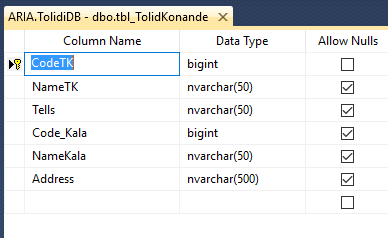
5. tbl\_Kala: در این جدول لیست اقلام را ثبت میکنیم



6. tbl\_Sales: این جدول مربوط به فاکتورهای فروش میباشد.



7. tbl\_TolidKonande:این مشخصات تولید کننده ها را در خود ذخیره میکند



تصاویرقسمت های مختلف برنامه و توضیحات

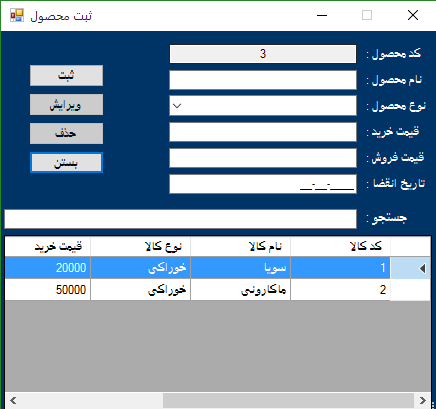
فرم لاگین(ورود): همانگونه که مشخص است این فرم هنگام اجرای برنامه مشاهده میشود که برای وارد شدن به برنامه باید اطلاعات صحیح برا ورود را وارد برنامه کرد.



فرم اصلی برنامه :



فرم ثبت محصولات :



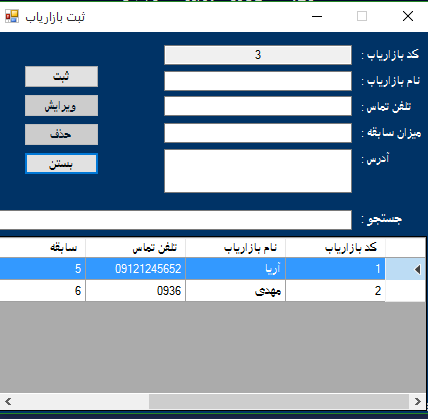
فرم ثبت تولید کننده :



فرم ثبت طرف حساب :



فرم ثبت بازاریاب :



فرم ثبت فاکتور فروش:



فرم لیست فاکتور فروش :



فرم ثبت فاکتور خرید :



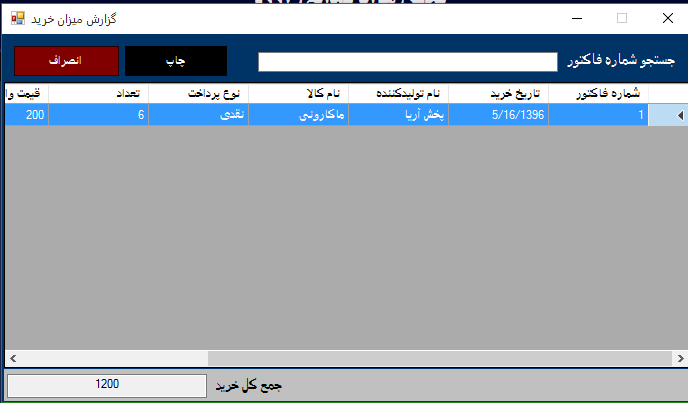
فرم لیست فاکتورهای خرید »



فرم گزارش میزان فروش :



فرم گزارش میزان خرید :



پرینتی از نمایش اطلاعات فروش :



نمونه ای از سورس کدبرنامه

**فرم فاکتور فروش :**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Data.SqlClient;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Tolidi

{

public partial class frmListForosh : Form

{

SqlConnection con = new SqlConnection("Server=(local);Initial Catalog=TolidiDB;Integrated Security=True");

SqlDataAdapter da;

DataTable dt1 = new DataTable();

public frmListForosh()

{

InitializeComponent();

}

public DataTable MySelect(string Sql)

{

SqlCommand cmd = new SqlCommand();

cmd.Connection = con;

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

con.Open();

cmd.CommandText = Sql;

da.Fill(dt);

con.Close();

return dt;

}

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

{

DataTable dt = new DataTable();

dt = MySelect("SELECT NumFakS, Datex, Cod\_Moshtari, Code\_Baz, Code\_Kala, TypePay, Tedad, priceOne, Takhfif, TotalCost FROM tbl\_Sales GROUP BY NumFakS, Datex, Cod\_Moshtari, Code\_Baz, Code\_Kala, priceOne, Takhfif, TypePay, Tedad, TotalCost");

dataGridView1.DataSource = dt;

dataGridView1.Columns[0].HeaderText = "شماره فاکتور";

dataGridView1.Columns[1].HeaderText = "تاریخ فروش";

dataGridView1.Columns[2].HeaderText = "کد مشتری";

dataGridView1.Columns[3].HeaderText = "کد بازاریاب";

dataGridView1.Columns[4].HeaderText = "کد کالا";

dataGridView1.Columns[5].HeaderText = "نوع پرداخت";

dataGridView1.Columns[6].HeaderText = "تعداد";

dataGridView1.Columns[7].HeaderText = "قیمت واحد";

dataGridView1.Columns[8].HeaderText = "تخفیف";

dataGridView1.Columns[9].HeaderText = "قیمت کل";

}

private void button3\_Click(object sender, EventArgs e)

{

frmFrosh x = new frmFrosh();

x.ShowDialog();

}

int NumFakS = 0;

private void button5\_Click(object sender, EventArgs e)

{

NumFakS = Convert.ToInt32(dataGridView1.CurrentRow.Cells["NumFakS"].Value);

if (MessageBox.Show("آیا می خواهید حذف کنید؟", "", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)

{

SqlCommand cmd = new SqlCommand("DELETE tbl\_Sales where NumFakS=@NumFakS", con);

cmd.Parameters.AddWithValue("@NumFakS", NumFakS);

con.Open();

cmd.ExecuteNonQuery();

con.Close();

MessageBox.Show("فاکتور مورد نظر با موفقیت حذف شد");

frmListForosh\_Load(sender, e);

}

else { }

}

private void textBox5\_TextChanged(object sender, EventArgs e)

{

//جستجو

con.Open();

da = new SqlDataAdapter("select \* from tbl\_Sales where NumFakS like N'%" + textBox5.Text + "%' ", con);

dt1 = new DataTable();

da.Fill(dt1);

dataGridView1.DataSource = dt1;

con.Close();

}

}

}

**سورس فرم ثبت محصول :**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Data.SqlClient;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Tolidi

{

public partial class frmKala : Form

{

SqlConnection con = new SqlConnection("Server=(local);Initial Catalog=TolidiDB;Integrated Security=True");

SqlDataAdapter da;

DataTable dt1 = new DataTable();

public frmKala()

{

InitializeComponent();

}

//شماره ثبت

private string getNewID()

{

string AtCnt;

SqlCommand cmd = new SqlCommand();

SqlConnection conn = new SqlConnection("Server=(local);Initial Catalog=TolidiDB;Integrated Security=True");

conn.Open();

cmd.CommandType = CommandType.Text;

cmd.CommandText = "Select MAX(CodeKala) from tbl\_Kala";

cmd.Connection = conn;

AtCnt = Convert.ToString(cmd.ExecuteScalar());

conn.Close();

return AtCnt;

}

public void NumberSabt()

{

textBox1.Text = "";

int NewId = 1;

string strnewid = "";

object obj = getNewID();

if (obj.ToString() != "")

{

NewId = Convert.ToInt32(obj.ToString()) + 1;

strnewid = NewId.ToString();

textBox1.Text = strnewid;

}

else

{

textBox1.Text = "1";

}

}

public DataTable MySelect(string Sql)

{

SqlCommand cmd = new SqlCommand();

cmd.Connection = con;

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

con.Open();

cmd.CommandText = Sql;

da.Fill(dt);

con.Close();

return dt;

}

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

{

button2.Enabled = false;

button3.Enabled = false;

NumberSabt();

DataTable dt = new DataTable();

dt = MySelect("select CodeKala,NameKala,TypeKala,PriceBuy,PriceSale,DateExpire from tbl\_Kala");

dataGridView1.DataSource = dt;

dataGridView1.Columns[0].HeaderText = "کد کالا";

dataGridView1.Columns[1].HeaderText = "نام کالا";

dataGridView1.Columns[2].HeaderText = "نوع کالا";

dataGridView1.Columns[3].HeaderText = "قیمت خرید";

dataGridView1.Columns[4].HeaderText = "قیمت فروش";

dataGridView1.Columns[5].HeaderText = "تاریخ انقضا";

}

private void button1\_Click(object sender, EventArgs e)

{

if (textBox2.Text == "" || textBox3.Text == "" || textBox4.Text == "")

{

MessageBox.Show(".فیلدهای خالی رو پر کنید", "هشدار", MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButton.Button1, MessageBoxOptions.RtlReading);

}

else

{

SqlCommand cmd = new SqlCommand("INSERT INTO tbl\_Kala (CodeKala,NameKala,TypeKala,PriceBuy,PriceSale,DateExpire) VALUES (@CodeKala,@NameKala,@TypeKala,@PriceBuy,@PriceSale,@DateExpire)", con);

cmd.Parameters.AddWithValue("@CodeKala", textBox1.Text);

cmd.Parameters.AddWithValue("@NameKala", textBox2.Text);

cmd.Parameters.AddWithValue("@TypeKala", comboBox1.Text);

cmd.Parameters.AddWithValue("@PriceBuy", textBox4.Text);

cmd.Parameters.AddWithValue("@PriceSale", textBox3.Text);

cmd.Parameters.AddWithValue("@DateExpire", maskedTextBox1.Text);

con.Open();

cmd.ExecuteNonQuery();

con.Close();

MessageBox.Show(".کالا مورد نظر با موفقیت ثبت شد", "ثبت کالا", MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButton.Button1, MessageBoxOptions.RtlReading);

frmKala\_Load(sender, e);

textBox1.Clear();

textBox2.Clear();

textBox3.Clear();

textBox4.Clear();

textBox5.Clear();

NumberSabt();

}

}

private void button2\_Click(object sender, EventArgs e)

{

SqlCommand cmd = new SqlCommand("UPDATE tbl\_Kala Set NameKala=@NameKala,TypeKala=@TypeKala,PriceBuy=@PriceBuy,PriceSale=@PriceSale,DateExpire=@DateExpire Where CodeKala=@CodeKala", con);

cmd.Parameters.AddWithValue("@CodeKala", textBox1.Text);

cmd.Parameters.AddWithValue("@NameKala", textBox2.Text);

cmd.Parameters.AddWithValue("@TypeKala", comboBox1.Text);

cmd.Parameters.AddWithValue("@PriceBuy", textBox4.Text);

cmd.Parameters.AddWithValue("@PriceSale", textBox3.Text);

cmd.Parameters.AddWithValue("@DateExpire", maskedTextBox1.Text);

con.Open();

cmd.ExecuteNonQuery();

con.Close();

MessageBox.Show("کالا مورد نظر با موفقیت ویرایش شد", "ویرایش کالا", MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButton.Button1, MessageBoxOptions.RtlReading);

frmKala\_Load(sender, e);

textBox1.Clear();

textBox2.Clear();

textBox3.Clear();

textBox4.Clear();

textBox5.Clear();

NumberSabt();

}

int CodeKala = 0;

private void button3\_Click(object sender, EventArgs e)

{

CodeKala = Convert.ToInt32(dataGridView1.CurrentRow.Cells["CodeKala"].Value);

if (MessageBox.Show("آیا می خواهید حذف کنید؟", "", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)

{

SqlCommand cmd = new SqlCommand("DELETE tbl\_Kala where CodeKala=@CodeKala", con);

cmd.Parameters.AddWithValue("@CodeKala", CodeKala);

con.Open();

cmd.ExecuteNonQuery();

con.Close();

MessageBox.Show("کالا مورد نظر با موفقیت حذف شد");

frmKala\_Load(sender, e);

textBox2.Clear();

textBox3.Clear();

textBox4.Clear();

textBox5.Clear();

}

else { }

}

private void button4\_Click(object sender, EventArgs e)

{

this.Close();

}

private void textBox5\_TextChanged(object sender, EventArgs e)

{

//جستجو

con.Open();

da = new SqlDataAdapter("select \* from tbl\_Kala where NameKala like N'%" + textBox5.Text + "%' ", con);

dt1 = new DataTable();

da.Fill(dt1);

dataGridView1.DataSource = dt1;

con.Close();

}

private void dataGridView1\_CellClick(object sender, DataGridViewCellEventArgs e)

{

textBox1.Text = dataGridView1.CurrentRow.Cells["CodeKala"].Value.ToString();

textBox2.Text = dataGridView1.CurrentRow.Cells["NameKala"].Value.ToString();

comboBox1.Text = dataGridView1.CurrentRow.Cells["TypeKala"].Value.ToString();

textBox4.Text = dataGridView1.CurrentRow.Cells["PriceBuy"].Value.ToString();

textBox3.Text = dataGridView1.CurrentRow.Cells["PriceSale"].Value.ToString();

maskedTextBox1.Text = dataGridView1.CurrentRow.Cells["DateExpire"].Value.ToString();

button2.Enabled = true;

button3.Enabled = true;

}

}

}

**سورس فرم ورود به برنامه :**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Data.SqlClient;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Tolidi

{

public partial class frmLogin : Form

{

public SqlCommand cmd = new SqlCommand();

SqlConnection con = new SqlConnection("Server=(local);Initial Catalog=TolidiDB;Integrated Security=True");

public SqlDataReader dr;

public frmLogin()

{

InitializeComponent();

}

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

{

Application.Exit();

}

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

{

if (txtPass.Text == "" && txtUser.Text == "")

{

MessageBox.Show("نام کاربری و کلمه عبور را وارد نمایید!", "اخطار", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

else

{

string sql = "select UserName, Password FROM tbl\_Login where UserName='" + txtUser.Text + "' and Password='" + txtPass.Text + "'";

cmd = new SqlCommand(sql, con);

con.Open();

dr = cmd.ExecuteReader();

dr.Read();

if (dr.HasRows)

{

this.Hide();

dr.Close();

con.Close();

new Main().ShowDialog();

}

else

{

MessageBox.Show("کلمه عبور صحیح نیست!", "اخطار", MessageBoxButtons.OK, MessageBoxIcon.Error);

con.Close();

}

}

}

}

}

4-3 نتیجه گیری

پس از نوشتن این برنامه و تکمیل آن میتوان گفت زبان برنامه نویسی سی شارپ یک زبان جامع و کامل و با کمترین پیچیدگی نسبت به بقیه زیان های سطح پایین است و کار با این زبان راحتتر است که میتوان یکی از دلایل این راحتی را شی گرا بودن زبان سی شارپ تلقی کرد.همچنین گرافیکی بودن این برنامه و زبان باعث جذابیت بیشتر این زبان و برنامه شده و به دلیل محبوبیت زیاد این زبان میتوان به راحتی در مراجع و منابع های مختلف به دنبال اطلاعات بیشتر بود و در صورت بروز مشکل آن ار برطرف کرد که این امر باعث راحتی و سهولت بیشتر در برنامه نویسی میشود.