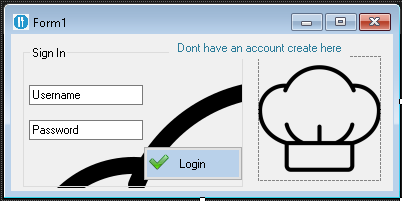
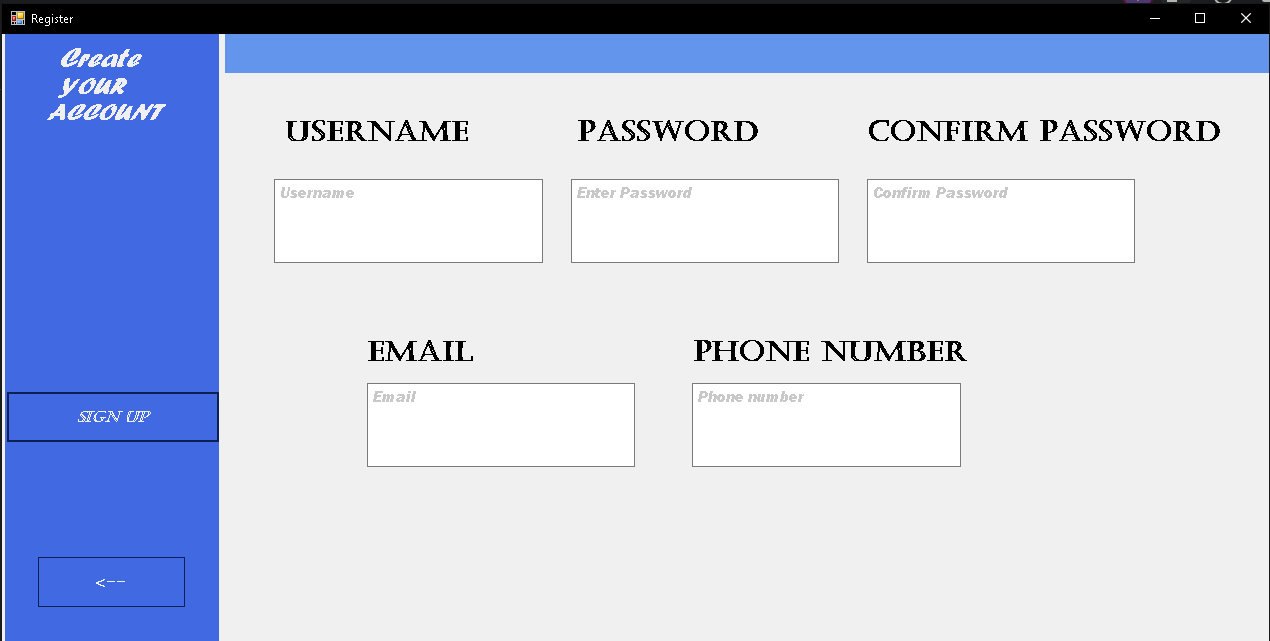
Документация по проект Restaurant System

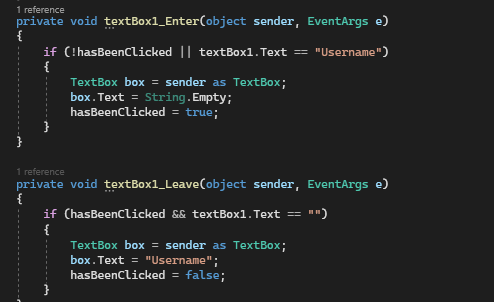
изработено от Райан Райков и Филип Момчилов  
https://github.com/Barabancho/ResturantSystem

Добре дошли в документацията за Restaurant System за управление! Този документ служи като изчерпателно ръководство за разбиране и използване на функциите на системата. Независимо дали сте собственик на ресторант, мениджър или служител, тази документация ще ви осигури необходимата информация, за да работите ефективно с Restaurant System за управление и да оптимизирате операциите във вашия ресторант.

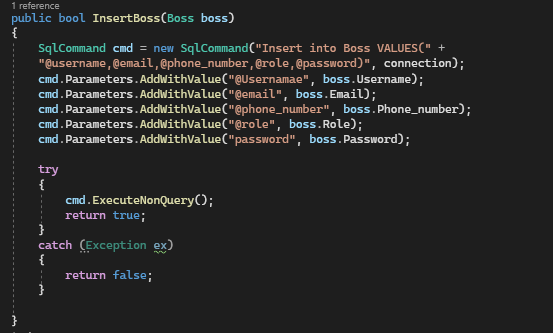
Нека да започнем с отварянето на нашата програма:  


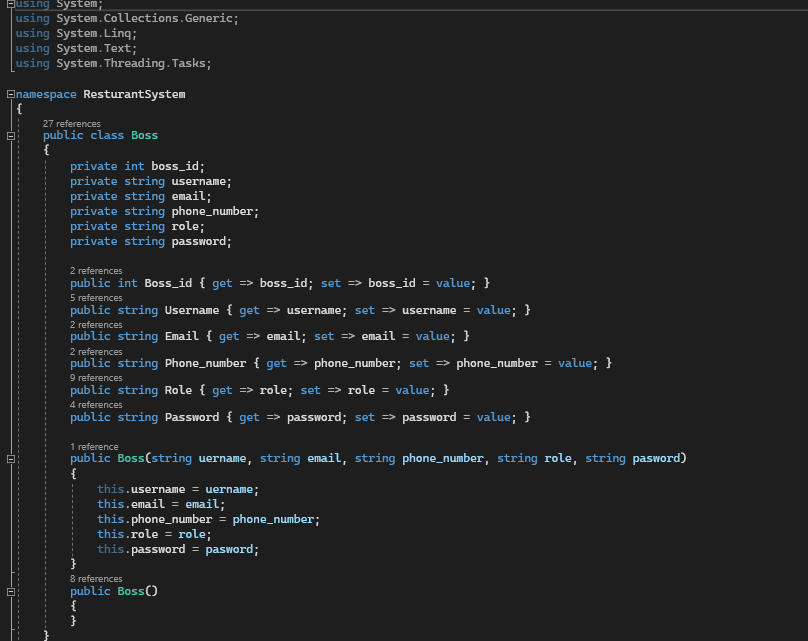
За да можете да влезете в нашата програма е нужно да си направите аккаунт или ако имате вече да влезете с него като въведете исканата информация в полетата Username и Password ,след въвеждането е нужно да кликнете върху Login бутона и ако всичко е наред трябва да сте заредили в нашият Options form.Но ако сте нови в нашата програма както казахме по-рано ви е нужно да си направите аккаунт.Това става след кликане на текста в синьо „Dont have an account create here” след цъкането на този бутон вие ще бъдете препратени на Form на име **REGISTER :**  


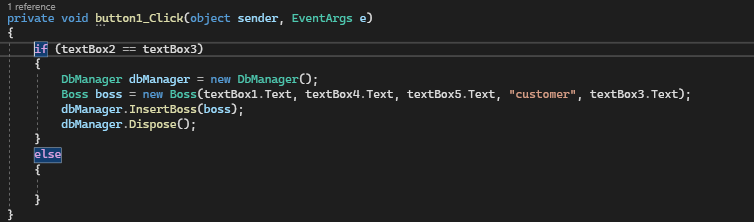
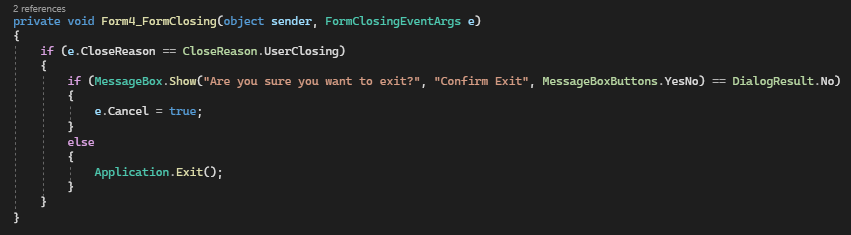
При зареждане на Form Register ще забележите 5-те големи Textbox-a които са Username,Password,Confirm Password ,Email и Phone Number.След въвеждане на исканата от програма информация и цъкане на на Sign In вашият акаунт ще бъде създаден.Но нека да видим кода на всички части от този Form.

Нека да започнем с 5-те Textbox-а (Username,Password,Confirm Password ,Email и Phone Number):  
  
Функцията на TextBox1\_Enter е след цъкане на текста например„Name” той да изчезне това помага за по-бързо въвеждане информация в текст бокса.  
  
Функцията на TextBox1\_Leave прави така ,че след кликане на например„Name”и не се въведе информация в него но след това се кликне върху различен от „Name” името да се върне по зададения от Админа name.

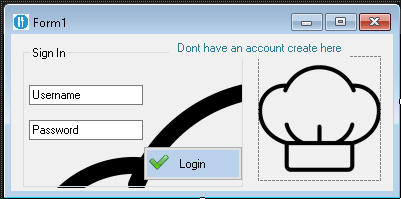
Сега нека да погледнем с **Sign In** бутона :

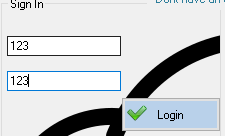
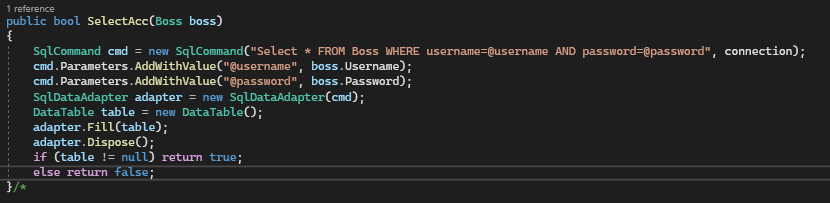
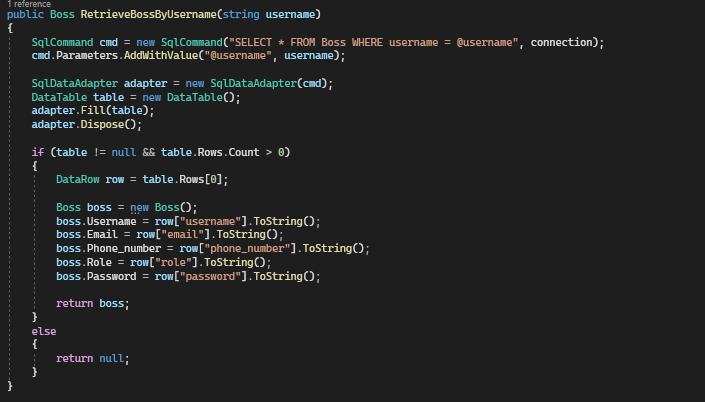
  
С този код правим връзка с базата данни и задаваме стойностите в нея.  
Които както виждате в кода са Username , email,phone number ,role ,password.

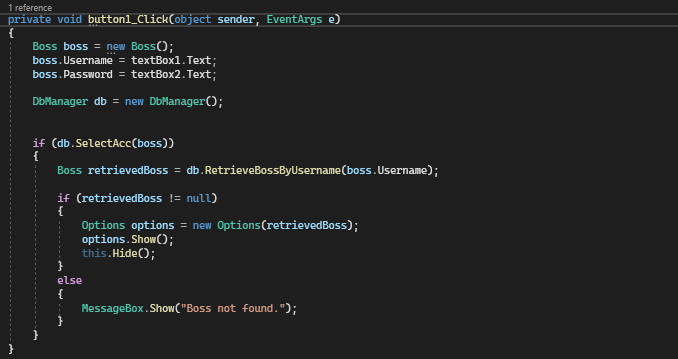
По-надолу виждаме cmd.executeNonQuery() което кара програмата да продължи да функционира при евентуален бъг в нея или ако има неправилна зададена стоиност.  
  


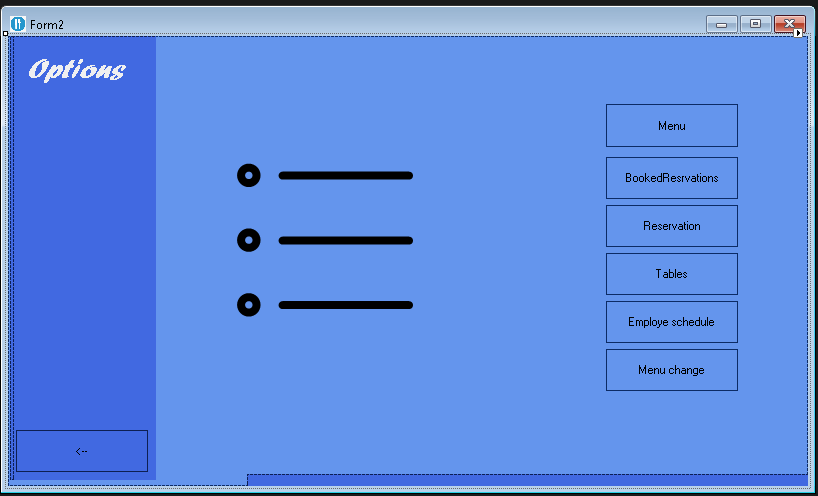
Класа извлича зададената информация от TextBox-ве от Form в DBManager.  
  
  
  
С този скрийншот виждаме какво изпраща бутона Sign in след цъкане.  
С този код разбираме ,че вкарва информацията в класа които показваме малко по-рано.  
  
  
  
  
  
  
  
За да можем успешно да затворим нашата програма след приключване сме добавили код за затваряне .  
  
  
Той е еднакъв при всеки един Form .От него разбираме ,че при затваряне на програмата Usera ще бъде попитан дали иска да я затвори това предотвратява от евентуални грешки и след натискане на бутона X в горния десен ъгъл ще бъде попитан от програмата „Are your sure you want to exit?” и това Usera ще има опция да избере да я затвори или да не я затвори.(,,Confirm Exit“),ако откаже този процес ще бъде прекъснат ,ако ли да програмата ще се изключи.

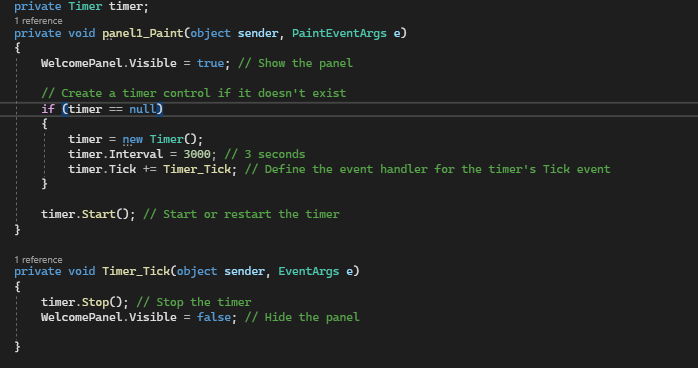
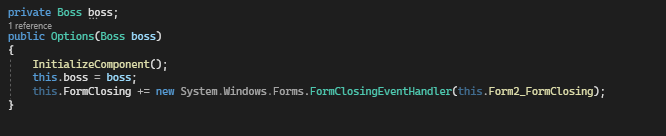
Но нека да се върнем назад до след създаването на акаунт в нашата програма което ще бъде възможно с помощта на този буттон:  
  
Както казахме неговата функция е да ни връща в Login

и закрива сегашния Form.  
  
След като се върнахме на **Login** нека да обясним малко по-подробно за него:   
  
  
  
  
За да влезем в акаунта който създадохме малко по-рано от Register form-а

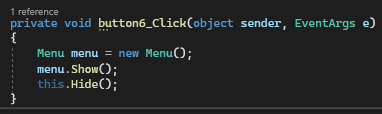
ще е нужно да въведем исканата информация от програмата а тя е Username и Password :   
  
(Това е пример за username и password на акаунт)  
И след въвеждането на исканата информация е нужно да натиснете Login бутона с тикчето на него:  
  
И с тези изпълнени условия вие успешно ще сте влезли в нашата програма!  
Но преди да пристъпим към следващия етап , нека да ви покажем функционалността на нашия код за **Login**:  
  
  
От горните кодове разбираме ,че тяхната функция е да потвърди за легитѝменоста на акаунта(или по просто дали този акаунт съществува в базата данни на програмата).

  
Функциоността на втория код е да изпраща ролята на usera до Options form-a за които ще разберете малко по-напред в документацията.

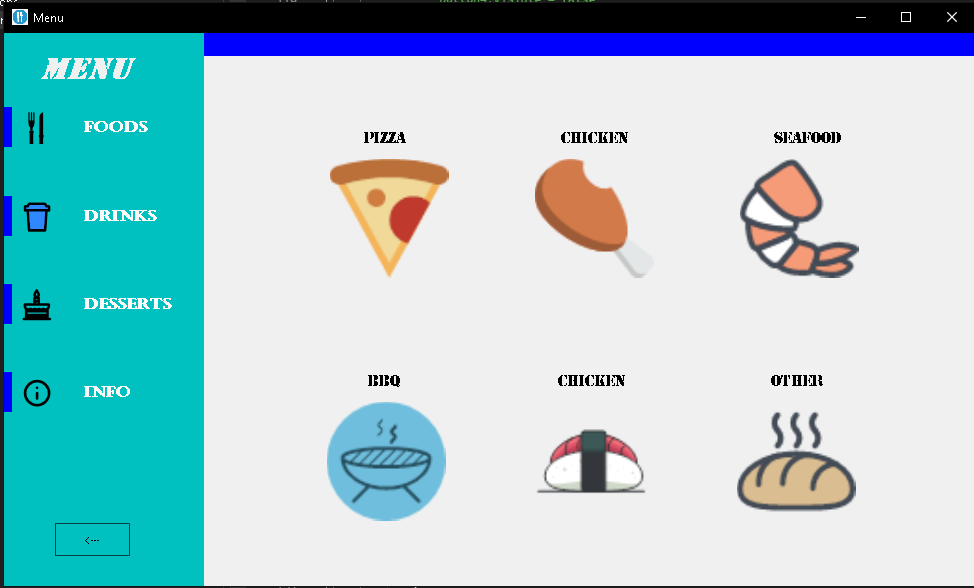
След влизането на вашия акаунт през Login вие ще бъдете изпратени на главния **Option form:**  
  
  


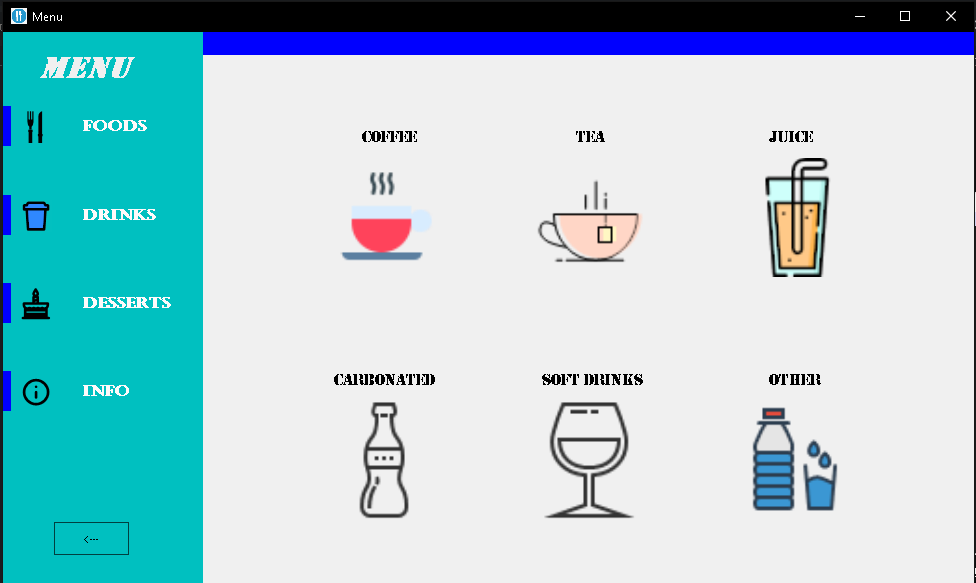
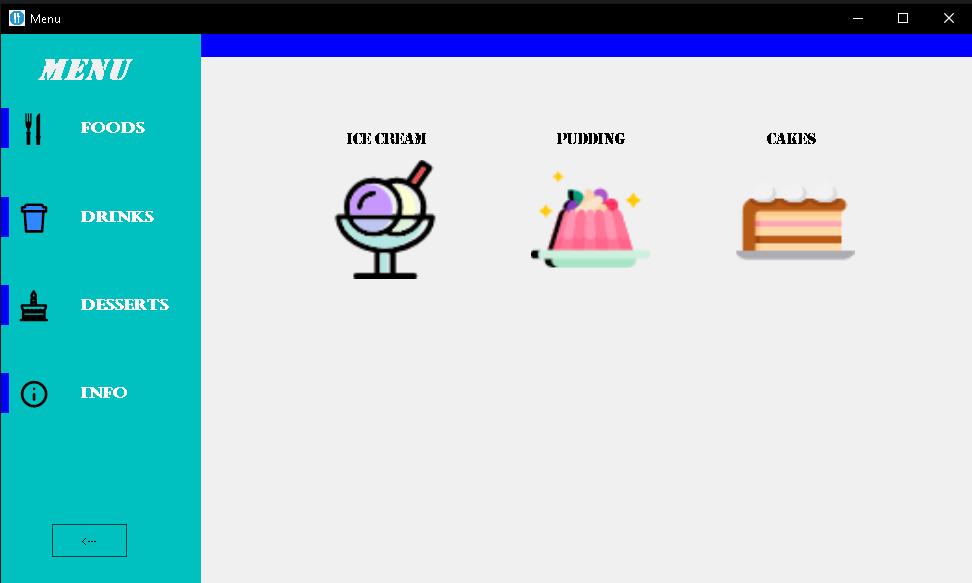
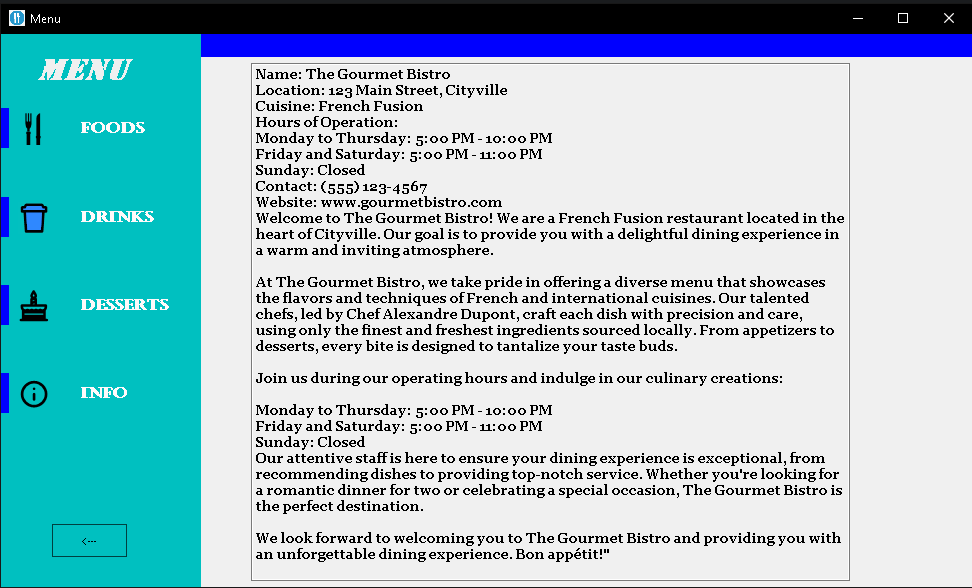
Ще бъдете посрещнати от надпис „Welcome” който ще изчезне след 3 секунди по задаване от админа ето и кода за него:  
  
Нека да продължим с Options  
  
Това са бутоните които вижда customer .Те са Menu и Reservation ,както видяхте по-нагоре има разлика между видимостта на Admina и Customera.  
При админа са бутоните с които се управляват Ресторанта/Програмата и те са BookedReservation,Tables,EmployeeSchedule както и MenuChanges ,това е става възможно с този код.  
  

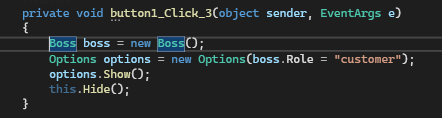

Неговата функция е да скрива или показва бутоните според акаунта в който потребителят е влязъл.Има и друга възможност която ако потребителят е успял да влезе чрез някакъв бъг в програмата да му изписва „Unauthorized access”.

След кликане на избран от вас бутон вие ще бъдете изпратени на дадения Form това е възможно с този код :  
  
Функцията на този код е да отвори в случая нов Menu form и да скрие сегашния form .

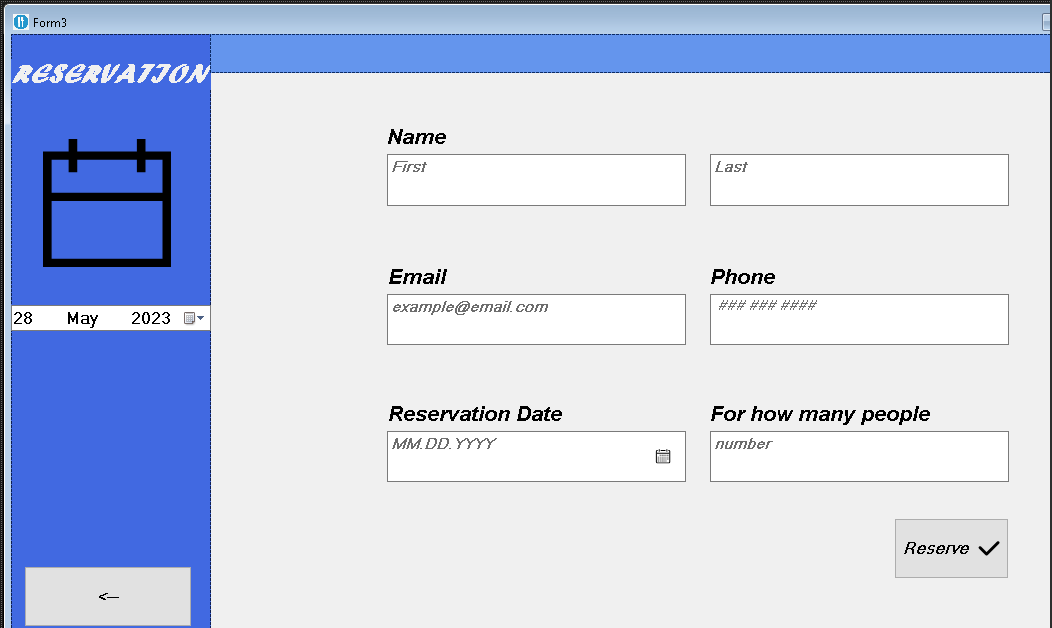
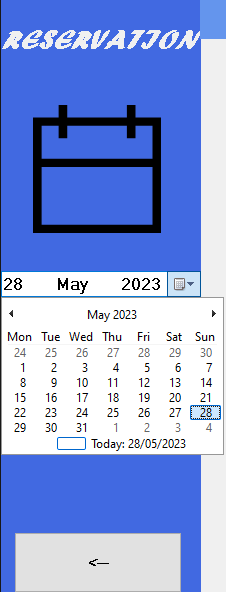
Нека да започнем може би с най-важната част от един ресторант и това е менюто,затова нека да отворим **Menu** Form;

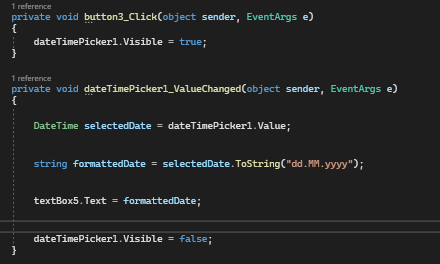


Като във всяко едно меню ние сме изложили нашите продукти за да бъде максимално лесно и достижимо за нашите клиенти .Имаме няколко бутона или по-точно няколко отдела за избиране на продукта които сте си избрали.  
Първия от тях е Foods както виждате по-горе(Основни ястия).  
Втория отдел е за напитки Drinks:  
  
 Третия отдел за десертите които предлагаме Desserts.  
И последния отдел е за информацията за нашия ресторант Info:  


В него се описва за местоположение на Ресторанта ни ,работно време ,телефон за връзка и нашия уебсайт.За повече информация прочетете скрийншота по-горе.  
За да се върнете към главния Options натиснете бутона за връщане:  
  
той работи по същия начин като този който споменахме в началото.   
  
 С кода който ще ви покажа по-долу направихме възможно показването и скриването на Food,Drinks,Desserts както и Info.  


След като сме се върнали на Options ,ще ви покажем и начина за ресерации в нашия ресторант:  


След кликането на бутона Reservation ще бъдете препратени на Reservation from:  
  
Първото нещо което ще видим като заредим е,че този form и Registration form имат textbox-ове в които трябва да се въвежда информация от потребителя.В този случаи се изисква First Name,Last Name,email,phone number,Reservation Date и For how many people(за колко човека е резервацията).За улеснение на потребителя сме сложили бутон за избиране на дата :  
  
след като цъкне малкия календар в дясната част на правоъгълника ще ви се отвори календар с които ще можете да си изберете желаната дата за резервация:  


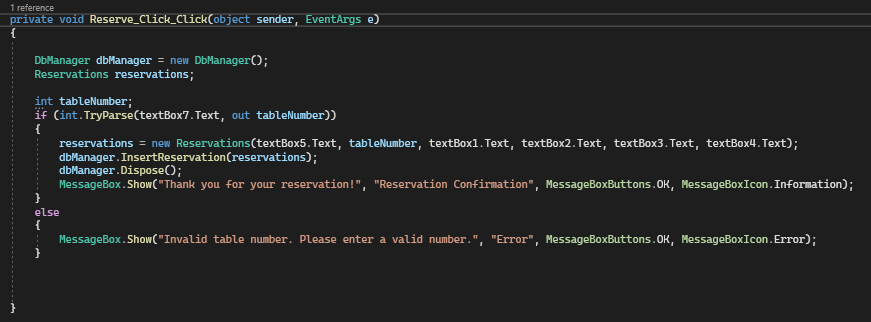
След избиране на дата от ваше желание тя автоматично ще бъде въведена в Reservation Date като например:  
  
Изпълнението на тази функция е възможна с благодарение на този код:  
  
Показваме контрола DateTimePicker след избирането на датата.  
С него извличаме избраната дата от DateTimePicker.  
След това форматираме датата с избрания формат.  
Актуализираме текстовото поле със форматираната дата.

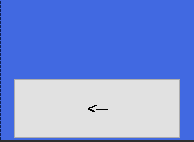
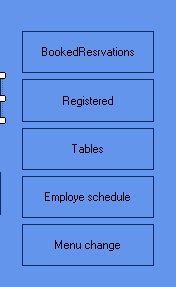
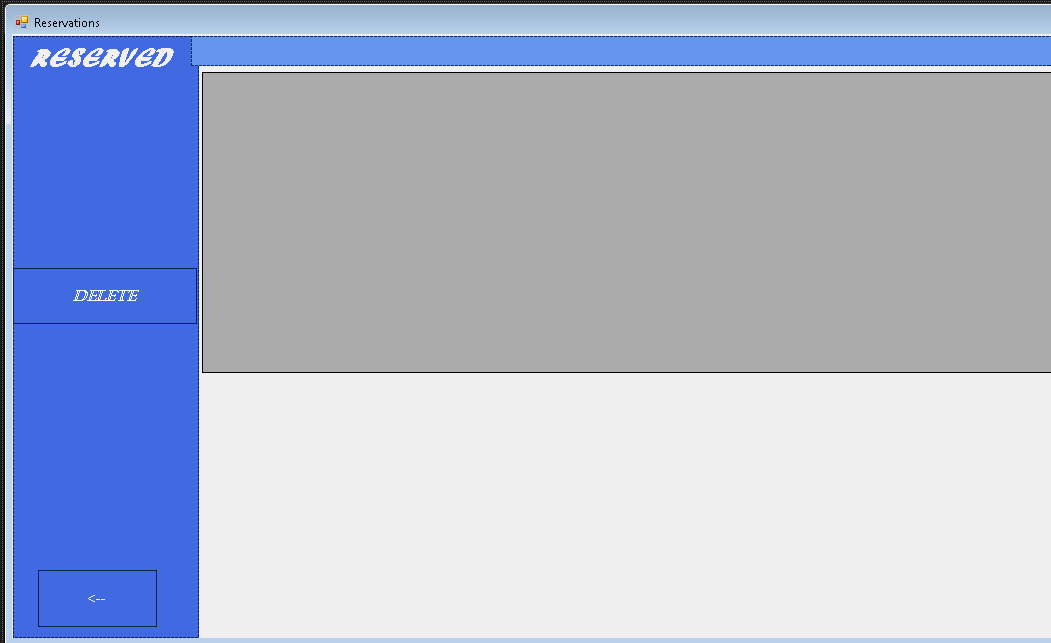
И накрая скриваме контрола DateTimePicker след избирането на датата.

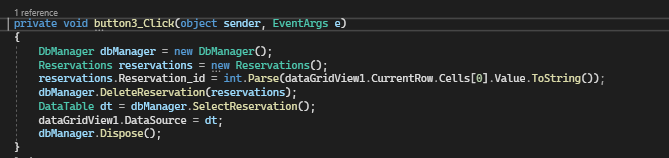
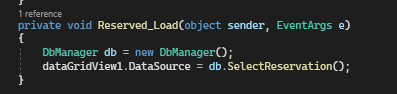
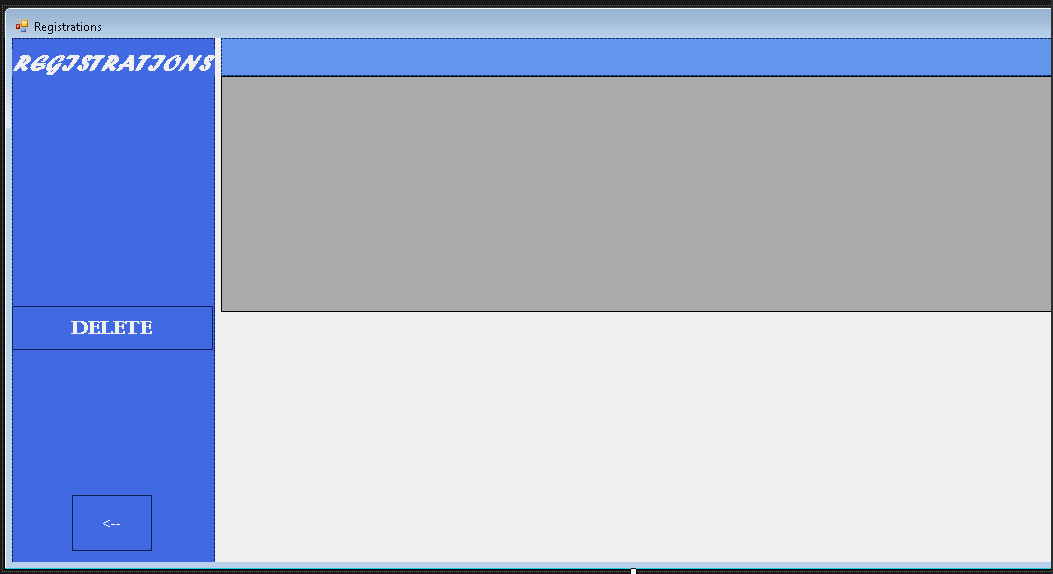
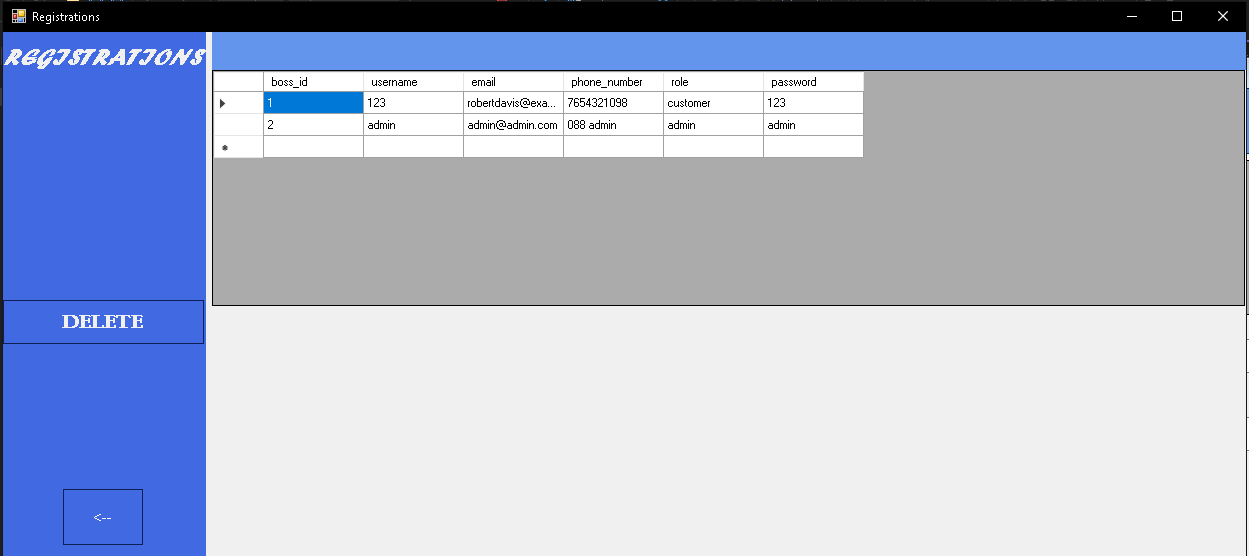
Когато сме готви с нашата резервация натискаме Reserve бутона :  

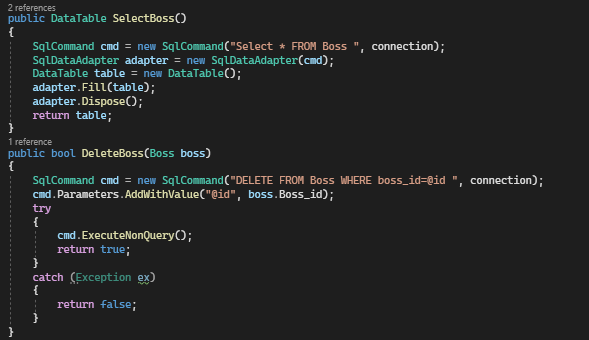
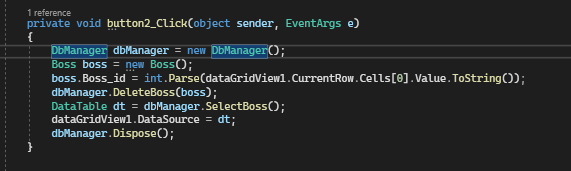
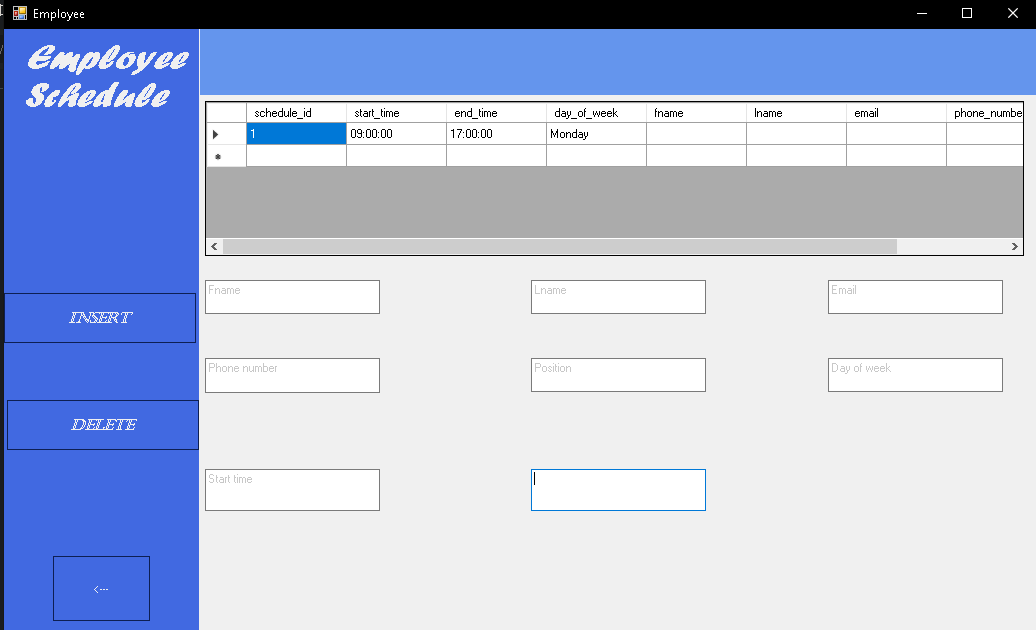

Нека да видим от какво е съставен Reserve бутона :  

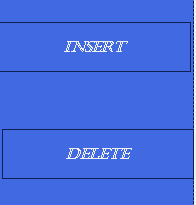
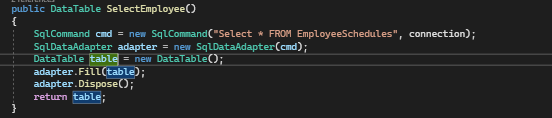

С този код правим връзка с базата данни и задаваме стойностите в нея.

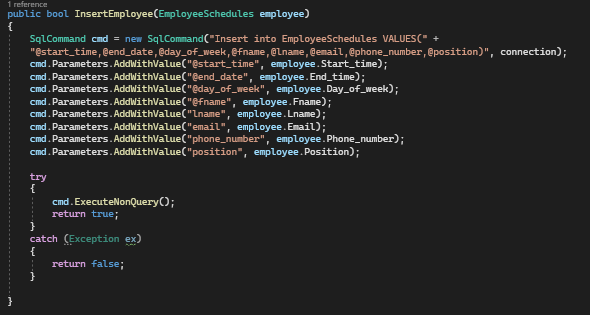
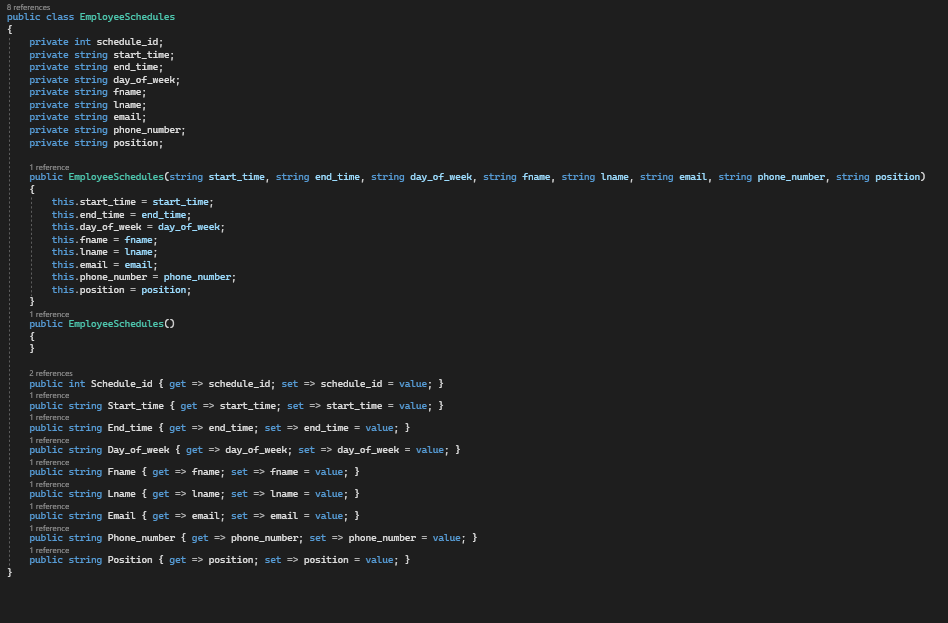
По-надолу виждаме cmd.executeNonQuery() което кара програмата да продължи да функционира при евентуален бъг в нея или ако има неправилна зададена стоиност.  
Ето и класа на Reservations:  
  
Класа извлича зададената информация от TextBox-ве от Form в DBManager.  
И остана какво става след извличането на данните след цъкането на бутона:  


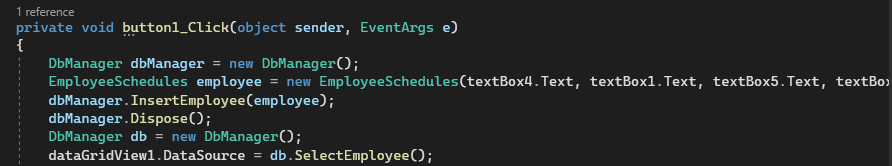
След правенето на успешна резервация от страна на потребителя ,програмата изписва „Thank you for your reservation!”или„Reservation Confirmation”.Забелязваме и ,че след неправилна или непопълнена информация програмата изписва„Invalid table number.Please enter a valid number.”или „Error”.  
След като сме готови с резервацията можем да се върнем с бутона за връщане:  
  
   
И с това приключват опците на customera.  
Но нека да продължим с опциите на Admin-a като например BookedReservations:  
  
  
Сега ще прегледаме кода на BookedReservations и ще обясним неговата   
функционалност:  
  
  
Както виждаме по Form-а ,BookedReservations е изграден от 2 бутон Delete и back button

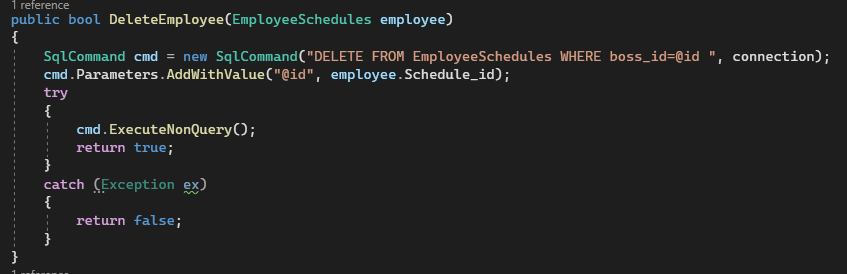
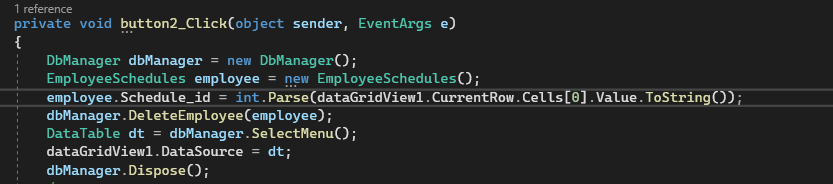
В случая DataGridView който ни служи като монитор поради способността му да показва данните въведени от Admina с помощта на направените от customer резервации както и на бутона Delete.  
  
За бутона **Delete**:  
  
  
Първата част от кода се използва за изписване на информацията в DataGridView,в втората част от кода разбираме ,че функцията му е да изтрива информация която съществува в дата базата по определено ID .  
  
Друг код на които дължим функциоността на бутона Delete:  
  
В скрийншота по-нагоре разбираме кода извлича сотиностие на дадените ID.  
За функциоността на Монитора или DataGridView дължим на:  
  
Този код прави така че след стартиране на Forma да се вижда таблицата.  
  
Нека да се върнем в Options и да видим Регистрацията(Registrations) на customers.  
  
  
Сега ще прегледаме кода на Registrations и ще обясним неговата   
функционалност:  
  
  
Той има абсолютно еднаква посторика с Reserved form-a   
Но все пак ето и какво прави Delete бутона :

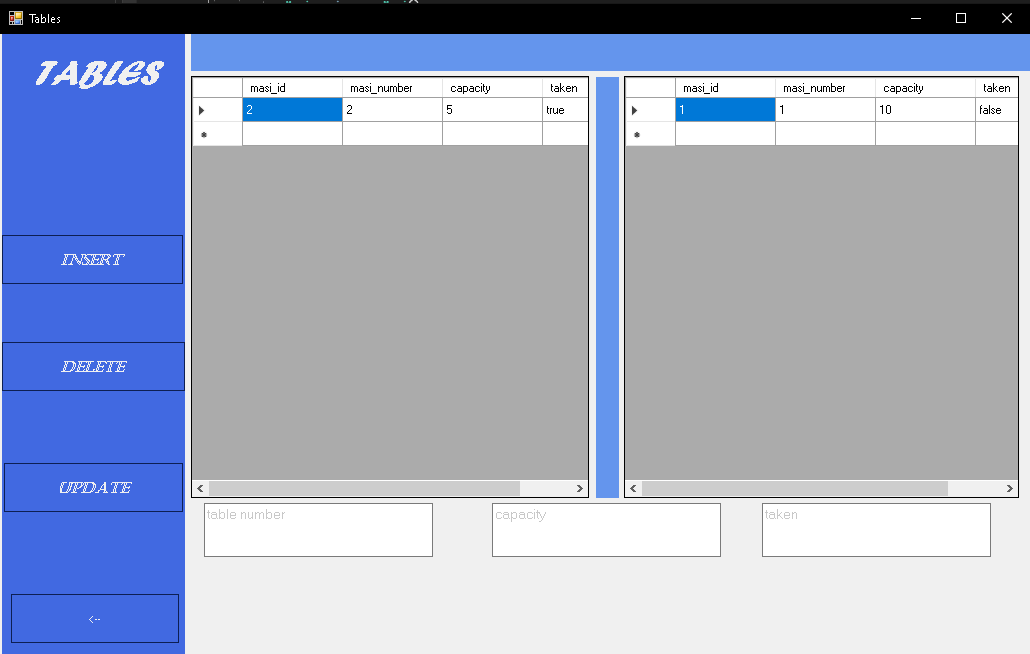
  
Първата част от кода се използва за изписване на информацията в DataGridView,в втората част от кода разбираме ,че функцията му е да изтрива информация която съществува в дата базата по определено ID .  
Друг код на които дължим функциоността на бутона Delete:  
  
И нека се върнем в Options и да изберем друг бутон които само админ може да види и това е **Employee Schedule:  
**Както виждаме по Form-а ,Employee Schedule е изграден от 3 бутона Insert,Delete и back button,

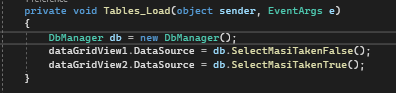
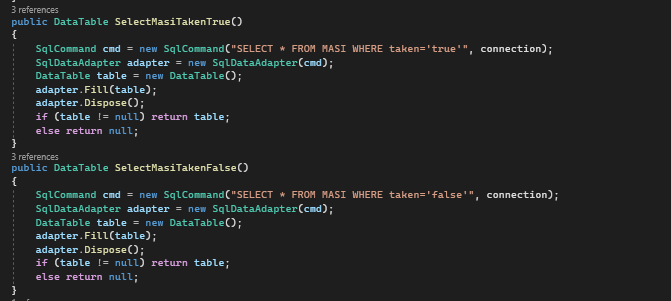
Също така забелязваме 8-те Textbox-a (FName,Lname,Email,Phone number,Position,Day of week,Start time,End Time) и в случая DataGridView който ни служи като монитор поради способността му да показва данните въведени от Admina с помощта на бутоните Insert и Delete.  
  
  
INSERT:  


  
С този код правим връзка с базата данни и задаваме стойностите в нея.  
  
  
Класа извлича зададената информация от TextBox-ве от Form в DBManager.  


С този код разбираме ,че този код вкарва информацията в класа които показваме малко по рано .  


DELETE:  
  
С този код разбираме ,че функцията му е да изтрива информация която съществува в дата базата по определено ID .  
  
  
В скрийншота разбираме кода извлича сотиностие на дадените ID.  


И нека да се върнем в Options и да изберем Tables.  
  
  
  
  
  
  
  
  
  
  
  
  
Сега ще прегледаме кода на Tables и ще обясним неговата   
функционалност:  


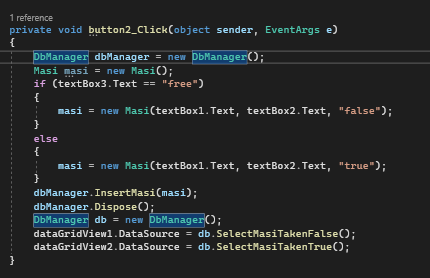
След отваряне на Tables забелязваме разликата от другите Forms  
и това се дължи на двата DataGridView-a на които както виждаме има две неща.Тоест DataBase-а ,че сортира на две отделни таблици .Първата таблица отговаря за заетите маси в ресторанта,а пък втората таблица отговаря точно за обратното,тоест свободните маси в ресторанта ни.  
Това се дължи на този код:  
  


От тези скрийншоти разбираме ,че SelectMasiTakenTrue

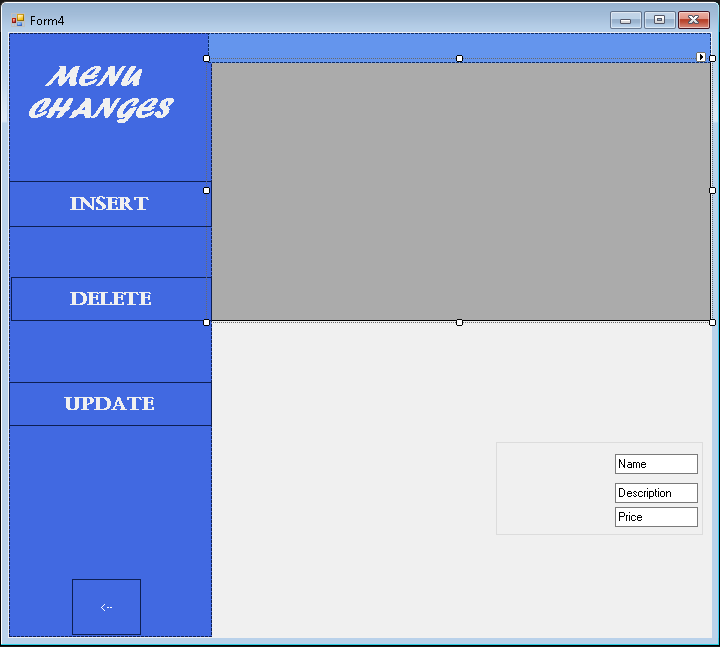
избира дадените маси със стойност taken=true.

За по-долния код е важи същото само дето  
избира дадените маси със стойност taken=false.

Както видяхте малко по-нагоре има и 3 бутона които са:  

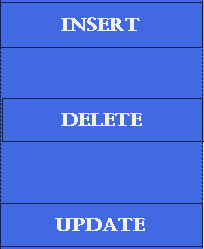

**INSERT:  
**Inserta задава стойности на DBManeger:  


Този код записва информацията от по-горния код в дата базата.

Сега ще прегледаме кода на MenuChanges и ще обясним неговата   
функционалност:

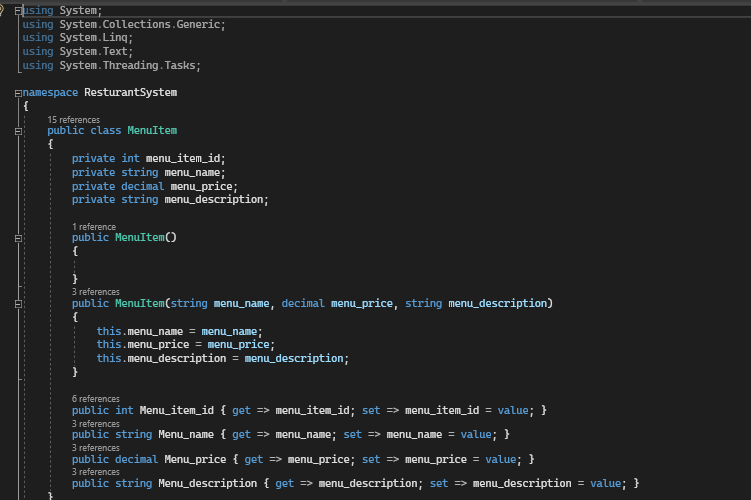
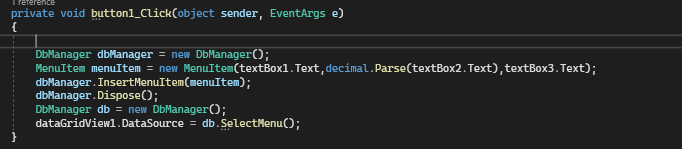
Както виждаме по Form-а ,MenuChanges е изграден от 4 бутона Insert,Delete,back button и Update.

Също така забелязваме 3-те Textbox-a (Name,Description,Price) и в случая DataGridView който ни служи като монитор поради способността му да показва данните въведени от Admina с помощта на

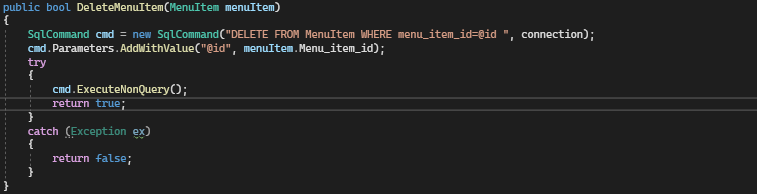
бутоните Insert,Delete и Update.  


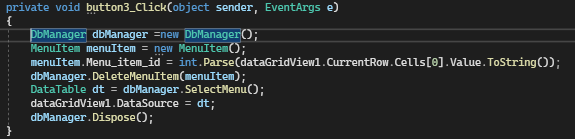
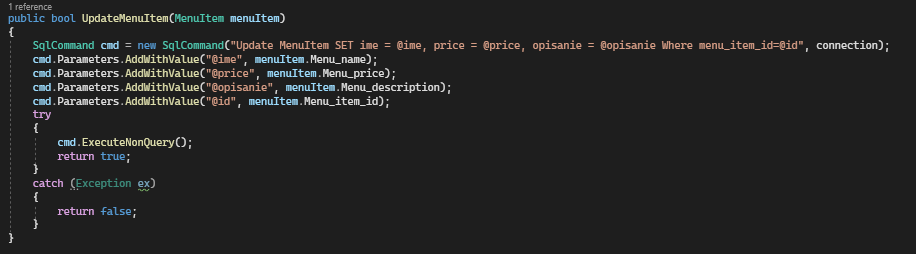
това се случва с помощта на този кодовете които ще ви покажем малко по-долу.  
INSERT:  

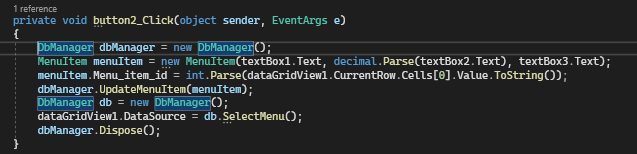

С този код правим връзка с базата данни и задаваме стойностите в нея.

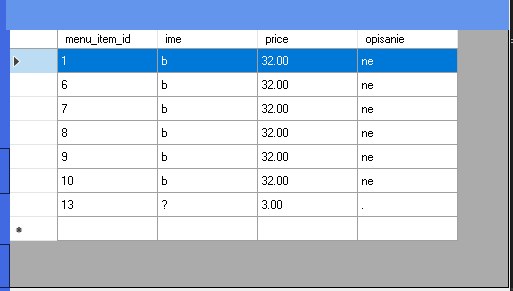
По-надолу виждаме cmd.executeNonQuery() което кара програмата да продължи да функционира при евентуален бъг в нея или ако има неправилна зададена стоиност.  
  
  
Класа извлича зададената информация от TextBox-ве от Form в DBManager.  
  


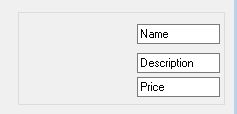
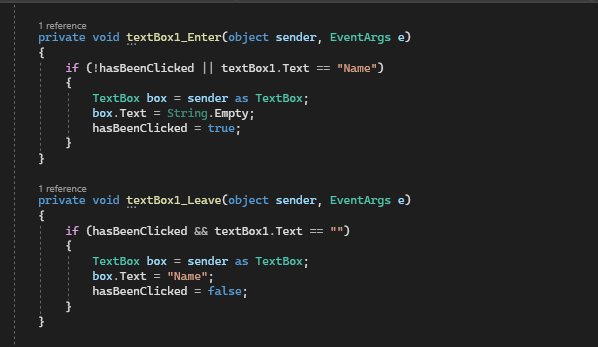
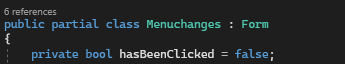
С този код разбираме ,че този код вкарва информацията в класа които показваме малко по рано .

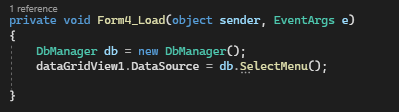
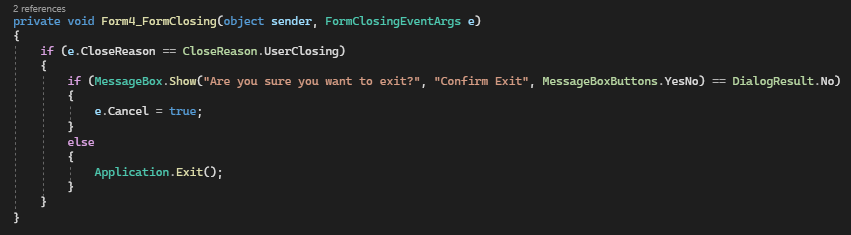
Извода е ,че INSERTA вкарва зададената информация от TextBox-ете в таблицата.  
  
DELETE:  


С този код разбираме ,че функцията му е да изтрива информация която съществува в дата базата по определено ID .По надолу виждаме пак cmd.executeNonQuery() което извършва същата функция като при INSERT-a.В класа се извършва една и съща дейност както при Inserta.  
  
  
В скрийншота по-нагоре разбираме кода извлича сотиностие на дадените ID.  
UPDATE:  
  
При UPDATE забелязваме еднаквости които се намират в INSERT и DELETE.Защото

при UPDATE избираме дадената информация изтриваме я както при DELETE и въвеждаме новата информация както при INSERT.  
cmd.executeNonQuery() извършва същата функция както при INSERT и DELETE.  
В класа е еднакво също.  


В скрийншота по-нагоре разбираме кода извлича сотиностие на дадените ID, и INSERT-ва в една команда зададената информация от TextBox-ете в таблицата.  


В TextBox-вете също има код и тои е :  
  
  
  
Функцията на TextBox1\_Enter е след цъкане на текста например„Name” той да изчезне това помага за по-бързо въвеждане информация в текст бокса.  
  
Функцията на TextBox1\_Leave прави така ,че след кликане на например„Name”и не се въведе информация в него но след това се кликне върху различен от „Name” името да се върне по зададения от Админа name.  
  
Тази променлива следи дали е кликнат дадения текст бокс.

  
С този код разбираме ,че при отваряне на Form показва стойностите от базата данни на DataGrid което както казах ни служи като монитор .Тази функция също се използва в INSERT,DELETE и UPDATE като нейната цел е да рефрешва стойностите на базата данни в DataGridView.  
  
За да можем успешно да затворим нашата програма след приключване сме добавили код за затваряне .  
  
  
Той е еднакъв при всеки един Form .От него разбираме ,че при затваряне на програмата Usera ще бъде попитан дали иска да я затвори това предотвратява от евентуални грешки и след натискане на бутона X в горния десен ъгъл ще бъде попитан от програмата „Are your sure you want to exit?” и това Usera ще има опция да избере да я затвори или да не я затвори.(,,Confirm Exit“),ако откаже този процес ще бъде прекъснат ,ако ли да програмата ще се изключи.

А пък ако сте решили да се върнете на Options(който ще го има по напред в документацията)сме направили бутон за връщане (back button)  
  
Както казахме неговата функция е да ни връща в Options

и закрива сегашния Form.  
  
