**Debreceni SZC Beregszászi Pál Technikum**

# *címer szakgim kicsi (1)*

*(:52/ 503-150 Fax: 52/314-204*

[*http://www.dszcberegszaszi.hu*](http://www.dszcberegszaszi.hu)

*E-mail:*

[*titkarsag@dszcberegszaszi.hu*](mailto:titkarsag@dszcberegszaszi.hu)

4032 Debrecen, Jerikó u. 17.

OM azonosító: 203033

**Szakképesítés megnevezése: Szoftverfejlesztő  
OKJ száma: 54 213 05**

**ZÁRÓDOLGOZAT**

**Patikai Kassza Menedzsment és Karbantartó Rendszer**

**Készítette**: **Konzulens:**  
Törő László András Boros Sándor

**Debrecen, 2023**

**-1.Bekezdés**

**- 2.Használt programok és azok rövid ismertetése:**

**-Visual Studio(Windows Forms)**

**-XAMP**

**- 3.Bejelentkezés**

**-Adatbázishoz való csatlakoztatás**

**- 4.Adminisztrátor menü**

**- 5.Adminisztrátori UserControlok ismertetése:**

**-Admin Műszertábla**

**-Felhasználó Hozzáadása**

**-Felhasználó Megtekintése/Törlése/Módosítása**

**-Saját Profil**

**-6.Patikusi szint és annak ismertetése:**

**-Gyógyszerek Validitása**

**-Gyógyszerek eladása, DGVprinter használata**

**-Gyógyszerek frissítése**

**-Patikusi MűszerTábla**

**-Gyógyszerek Megtekíntése**

**1.Bekezdés**

A patikai kassza rendszer elkészítéséhez számos programozási nyelv és platform közül választhattam volna, azonban a C# Windows Form mellett döntöttem. Az egyik legfőbb ok az volt, hogy a C# nyelv és az ehhez tartozó Windows Forms keretrendszer segítségével könnyedén és hatékonyan készíthetők felhasználói felületek. A Windows Forms lehetővé teszi az egyszerű és intuitív felhasználói élmény létrehozását, ami fontos a patikai kassza rendszer használhatósága szempontjából.

A C# nyelv nagy előnye az objektumorientált programozás (OOP) támogatása, amely lehetővé teszi a programozók számára a moduláris kódolás és az újrafelhasználhatóság. Az OOP segítségével a fejlesztők könnyedén oszthatják fel az alkalmazásokat különböző osztályokra és modulokra, ami javítja a kód olvashatóságát és karbantarthatóságát.

A patikai kassza rendszer adatbázis-kezelési funkcióihoz a MySql adatbázis-kezelő rendszert választottam. A C#-hoz számos MySql-kapcsoló elérhető, amelyek lehetővé teszik az alkalmazás és az adatbázis közötti könnyű kommunikációt. Az adatbázis-kezelési lehetőségek széles skálája segít abban, hogy az alkalmazás hatékonyan kezelje a készletnyilvántartást, az értékesítést és az ügyfelek adatait.

Összességében a C# Windows Form és a MySql kombinációja ideális választás a patikai kassza rendszer készítéséhez. A két technológia együttesen lehetővé teszi az egyszerű és hatékony felhasználói felület, valamint a hatékony adatbázis-kezelés létrehozását.

**2.Használt programok és azok rövid ismertetése:**

**- C#**

A C# (C Sharp) egy modern, objektumorientált programozási nyelv, amelyet az Microsoft fejlesztett ki. A C# nyelv a C és a C++ nyelvek tulajdonságait ötvözi, de egy sokkal könnyebben érthető és olvasható nyelvvé változik. A C# nyelv lehetővé teszi a fejlesztők számára a hatékonyabb és strukturáltabb kódolást, valamint az újrafelhasználhatóság és a kódolási idő csökkentését.

A C# nyelv számos fejlesztési platformot támogat, többek között a Windows Forms (WinForms) keretrendszert is. A Windows Forms lehetővé teszi a fejlesztők számára az egyszerű és hatékony felhasználói felületek létrehozását, amelyeket a Windows operációs rendszeren lehet futtatni. A Windows Forms lehetővé teszi a fejlesztők számára az egyszerű és intuitív felhasználói élmény kialakítását a különböző vezérlőelemek, például gombok, címkék és listák segítségével.

A Windows Forms használata előnyös lehet az olyan alkalmazások esetében, amelyeknek a felhasználói felülete fontos szerepet játszik, mint például a kereskedelmi vagy adminisztratív rendszerek, az irodai szoftverek és a játékok. A Windows Forms segítségével könnyedén létrehozhatók a modern felhasználói felületek, amelyek képesek megfelelni az aktuális trendeknek és az igényeknek.

Összességében a C# nyelv és a Windows Forms keretrendszer számos előnnyel rendelkezik a modern alkalmazások fejlesztése során. A C# nyelv lehetővé teszi a hatékonyabb és strukturáltabb kódolást, míg a Windows Forms keretrendszer lehetővé teszi az egyszerű és hatékony felhasználói felületek létrehozását. A C# és a Windows Forms kombinációja segíti a fejlesztőket abban, hogy hatékonyabb, intuitívabb és hasznosabb alkalmazásokat hozzanak létre.

**-XAMP és MySql**

A XAMPP egy olyan ingyenes, nyílt forráskódú szoftvercsomag, amelyben az Apache web szerver, a MySQL adatbázis kezelő, valamint a PHP és a Perl programozási nyelvek találhatóak. Az XAMPP azoknak az alkalmazásfejlesztőknek nyújt megoldást, akik szeretnének helyi környezetben dolgozni és tesztelni weboldalakat, alkalmazásokat és adatbázisokat. Az XAMPP lehetővé teszi a helyi fejlesztői környezet gyors és egyszerű telepítését és konfigurálását.

Az XAMPP-ban található MySQL adatbázis kezelő rendszer az egyik legnépszerűbb relációs adatbázis-kezelő, amely lehetővé teszi az adatbázisok hatékony kezelését. A MySQL nagy terhelést bír el, és lehetővé teszi a felhasználók számára az adatbázisok egyszerű kezelését, valamint a megbízható adatvédelmet.

A C# programozási nyelv lehetővé teszi a fejlesztők számára, hogy az XAMPP-ban található MySQL adatbázis-kezelőt használják. A MySQL adatbázis-kezelő könnyen integrálható a C# fejlesztői környezetekkel, például a Visual Studio-val, és lehetővé teszi az adatbázisok hatékony és biztonságos kezelését. A MySQL adatbázis-kezelő használata lehetővé teszi a C# fejlesztők számára, hogy hatékony és megbízható alkalmazásokat fejlesszenek, amelyeknek adatbázis-kezelési funkciókra van szükségük.

Összefoglalva, az XAMPP és a MySQL olyan fontos eszközök, amelyek lehetővé teszik a helyi fejlesztői környezet gyors és egyszerű telepítését és konfigurálását. Az XAMPP-ban található MySQL adatbázis-kezelő használata könnyen integrálható a C# fejlesztői környezetekkel, és lehetővé teszi a hatékony és biztonságos adatbázis-kezelést az alkalmazások számára. Az XAMPP és a MySQL használata segíti a C# fejlesztőket abban, hogy hatékonyabb, megbízhatóbb és jobb alkalmazásokat hozzanak létre.

**3.Bejelentkezés**

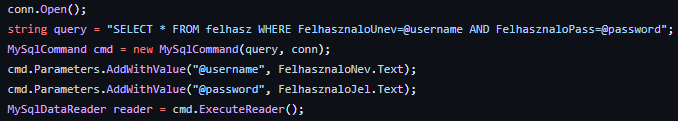
Az alkalmazás a MySQL adatbázis-kezelő rendszerrel kommunikál, így a felhasználói adatokat az adatbázisból ellenőrzi és azok alapján engedélyezi vagy tiltja a belépést.

Az osztály az "Form1" nevet viseli, és a felület kezelését végzi, például a felhasználó által beírt felhasználói név és jelszó lekérdezését az adatbázisból. A "Bejelentkezes\_Click" metódus indítja el az adatbáziskapcsolatot és végzi el a felhasználói bejelentkezés ellenőrzését.

------------------------------------------------------------------------------------------------------------

Az alkalmazás a "MySql.Data.MySqlClient" névtérben található "MySqlConnection" és "MySqlCommand" osztályokat használja az adatbázis-kapcsolat felépítéséhez és a lekérdezések végrehajtásához. Az adatbázisba való kapcsolódás a "conn.Open()" függvénnyel történik, majd a megfelelő lekérdezés összeállítása és a felhasználó által megadott adatokkal való paraméterezése következik.





-----------------------------------------------------------------------------------------------------------

Ha az adatbázisban a felhasználói név és jelszó egyezik, akkor a program eltávolítja a jelenlegi ablakot és megnyitja az adminisztrátori vagy a patikusi felületet a felhasználó szerepkörétől függően. Ha a felhasználói név vagy jelszó nem megfelelő, akkor a program hibaüzenetet jelenít meg a felhasználó számára.



-----------------------------------------------------------------------------------------------------------

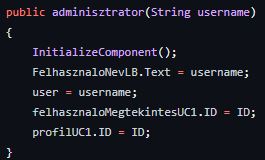
Összességében a kódrészlet egy egyszerű, de hatékony felhasználói bejelentkező felületet valósít meg, amely lehetővé teszi a felhasználók számára az adatok biztonságos kezelését.

**4.Adminisztrátor menü**

Az adminisztrátor nevű cs azonos nevű formot definiál a Zaródolgozat alkalmazásban, amely az adminisztrátor szinten elérhető tevékenységeket kezeli. A felhasználó az alkalmazásba történő belépéskor a felhasználói nevét megadva jut el erre a formra.

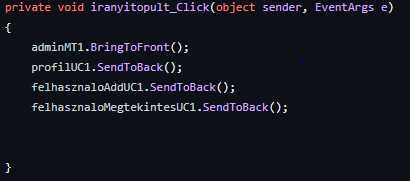
-----------------------------------------------------------------------------------------------------------

Az adminisztrátor formnak van egy String típusú "user" változója, amely a felhasználói nevet tárolja. Az ID tulajdonság ezt a felhasználói nevet adja vissza string formában. Az adminisztrátor formot lehet egy felhasználói névvel inicializálni. Ekkor a "FelhasznaloNevLB" címke megjeleníti a felhasználó nevét, és az "user" változóban eltárolja azt. Az "ID" tulajdonság értéke ezt az eltárolt felhasználói nevet adja vissza.



-----------------------------------------------------------------------------------------------------------

A formon van egy menüsor, amelynek gombok segítségével lehet navigálni a formon belül. A menüsor az "adminMT1" nevű UserControl-t tartalmazza, amely az irányítópult gombjának megnyomásával jelenik meg. Az "iranyitopult" gomb megnyomásával az irányítópult jelenik meg, és a többi UserControl elrejtődik.

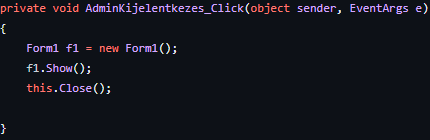


Mindenhól ugyan ez a kódrészlet játszódik le ha UserForm váltásról van szó.

-----------------------------------------------------------------------------------------------------------

A felhasználók kezelésére szolgáló UserControl-ok közül a "felhasznaloAddUC1" és a "felhasznaloMegtekintesUC1" között lehet váltani a "Felhaszadd" és a "FelhasznaloMegB" gombok segítségével. A "profilUC1" UserControl az adminisztrátor profilját kezeli, és a "Profil" gombbal érhető el.

Az "AdminKijelentkezes" gomb megnyomásával az alkalmazás visszalép az "LoginForm" formra, és a jelenlegi adminisztrátor kijelentkezik.



-----------------------------------------------------------------------------------------------------------

A kódrészlet használja a System, a System.Collections.Generic, a System.ComponentModel, a System.Data, a System.Drawing, a System.Linq, a System.Text, a System.Threading.Tasks és a System.Windows.Forms névtereket. Ezek a névterek a .NET keretrendszer részét képezik, és az ablakos alkalmazások fejlesztéséhez használatosak. Az "adminisztrator" form az "adminisztrator.Designer.cs" fájlban található, és a designer által generált kódot tartalmazza. Az eseménykezelők és az egyéb funkciók a "adminisztrator.cs" fájlban találhatók.

**5.Adminisztrátori UserControlok ismertetése:**

AdminMT:

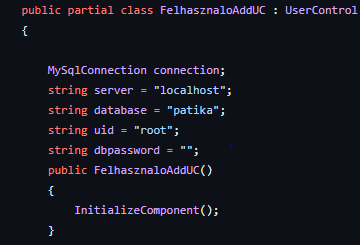
FelhasznaloADD:

A kód első része a szükséges namespace-ek importálását végzi el. A MySql.Data.MySqlClient namespace tartalmazza a MySQL adatbázis-kezelő .NET kliens osztályait. A többi namespace az alapvető C# típusok és az User Interface elemekhez szükséges namespace-ek.



-----------------------------------------------------------------------------------------------------------

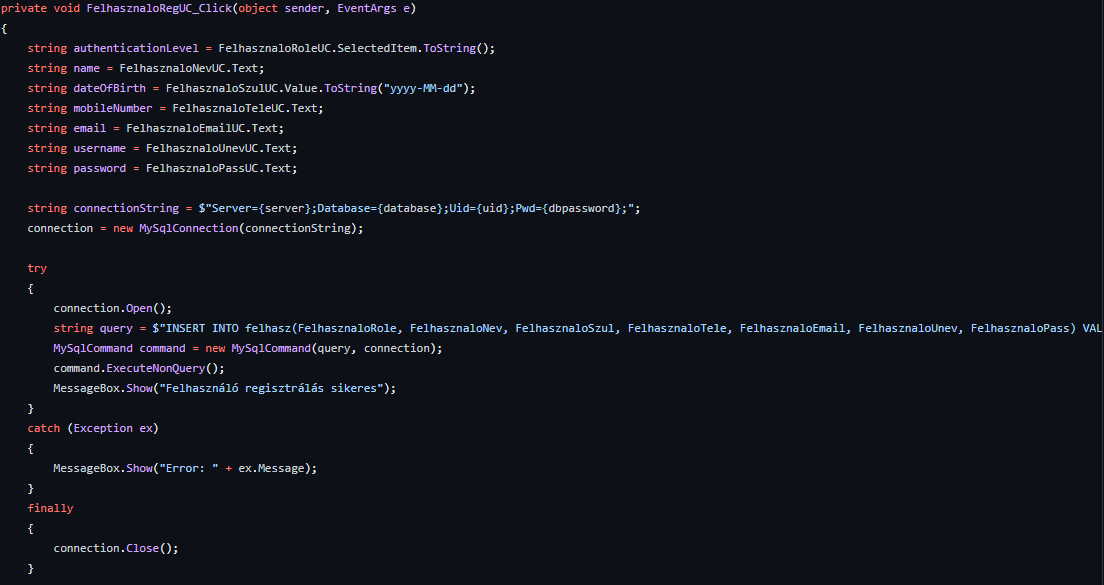
A kód következő része egy "FelhasznaloAddUC" nevű osztály definiálása, amely az előbb említett UserControl-t reprezentálja. Az osztály egy MySQL adatbázis-kezelővel való kapcsolatot hoz létre a felhasználó regisztrálásához.



A "FelhasznaloAddUC" osztály konstruktora inicializálja a MySQL adatbázis-kezelővel való kapcsolatot a "server", "database", "uid" és "dbpassword" változókból. Ezután a konstruktor inicializálja a UserControl megjelenítéséhez szükséges elemeket.

-----------------------------------------------------------------------------------------------------------

A "FelhasznaloRegUC\_Click" metódus végzi el a felhasználó regisztrálását a rendszerben. Az eljárás beolvassa a felhasználói felületen megadott adatokat, majd egy MySQL "INSERT INTO" utasítással elmenti azokat az adatbázisban.

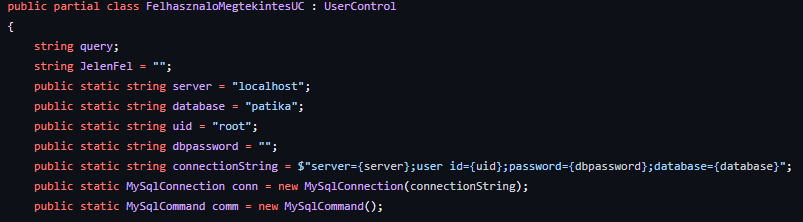


-----------------------------------------------------------------------------------------------------------

Ha a regisztráció sikeres, akkor a "MessageBox.Show" metódus segítségével üzenet jelenik meg az adminisztrátor számára, amely jelzi, hogy a regisztráció sikeres volt. Ha a regisztráció sikertelen, akkor az "Exception" üzenet jelenik meg a hiba részleteivel.

FelhasznaloMegtekintesUC:

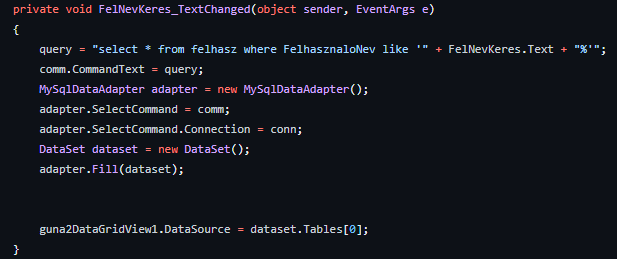
Az osztály tartalmaz néhány adattagot és metódust, amelyek lehetővé teszik a felhasználók megjelenítését, keresését és törlését egy adatbázisból. Az adatbázis kapcsolat beállításait a "server", "database", "uid", "dbpassword" és "connectionString" adattagok tárolják.



Mindenhol ugyan úgy csatlakozom az adatbázishoz.

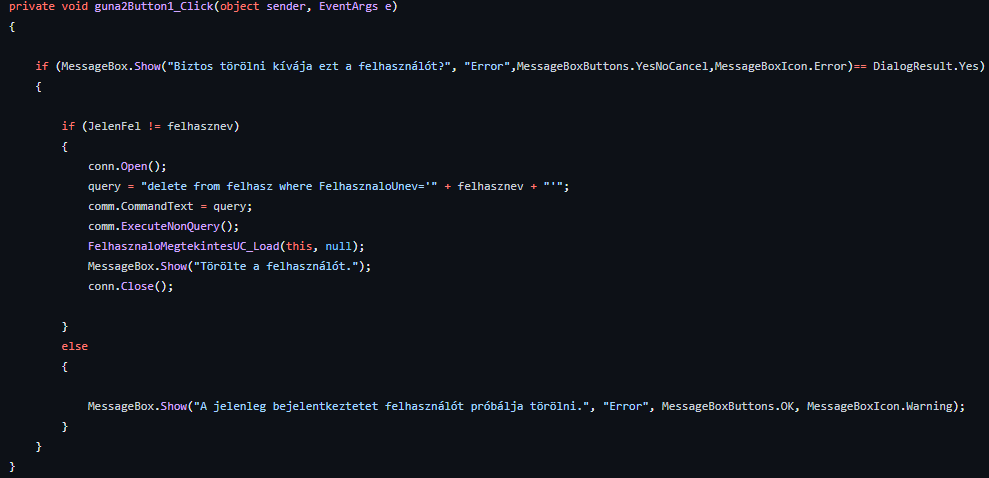
-----------------------------------------------------------------------------------------------------------

Az "Felhasznev" adattag a kereséshez használt felhasználónév, amelyet a "FelNevKeres\_TextChanged" metódusban lehet megadni. A "ID" tulajdonság beállítja a "JelenFel" adattagot.

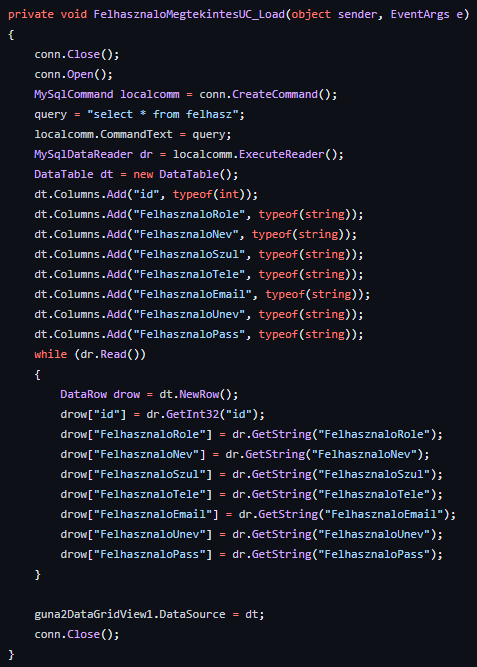


-----------------------------------------------------------------------------------------------------------

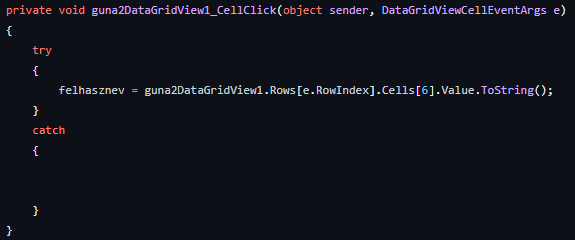
A "guna2Button1\_Click" metódus törli a kiválasztott felhasználót az adatbázisból. A "FelhasznaloMegtekintesUC\_Load" metódus betölti a felhasználók listáját az adatbázisból a "guna2DataGridView1" DataGridView-be.



-----------------------------------------------------------------------------------------------------------

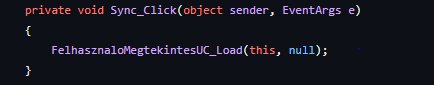


A "guna2DataGridView1\_CellClick" metódus megkapja a kattintott cella értékét a "felhasznev" változóba, hogy később használja a törlési folyamatban.



-----------------------------------------------------------------------------------------------------------

A "Sync\_Click" metódus újratölti az adatokat a DataGridView-ben, amikor a felhasználó rákattint az "Újratöltés" gombra.

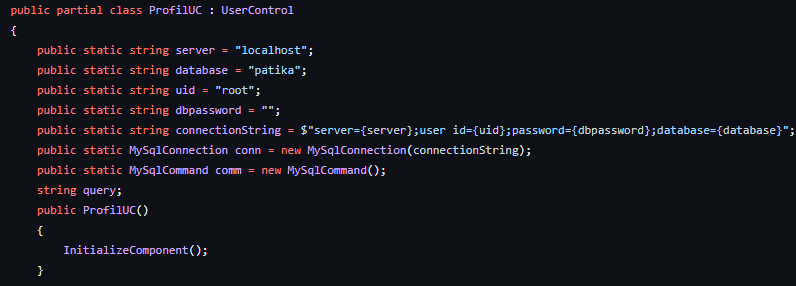


-----------------------------------------------------------------------------------------------------------

Összességében ez az osztály lehetővé teszi az adatbázisból származó felhasználók megtekintését, keresését és törlését a felhasználói felületen.

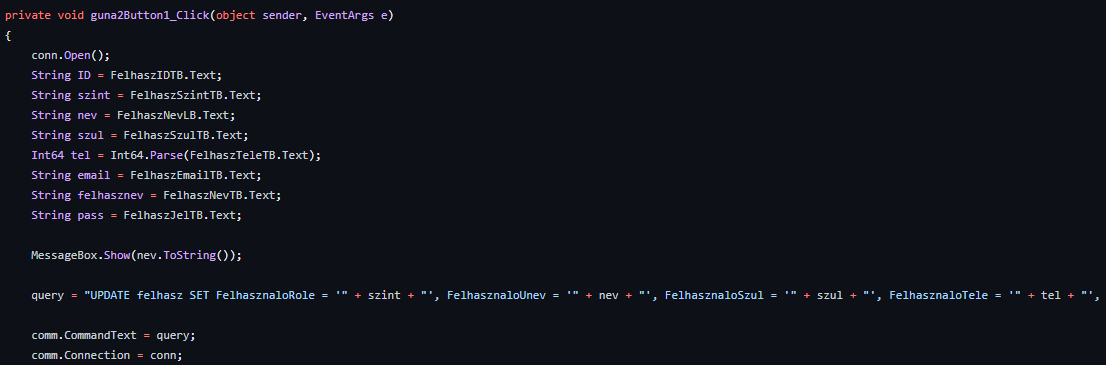
ProfilUC:

Ez egy C# programkód, amely kapcsolatot hoz létre egy MySQL adatbázissal és lehetővé teszi egy felhasználói felületen található profil szerkesztési funkció működését. Az adatbázis eléréséhez szükséges adatokat a programkód elején található változókban lehet beállítani. Az adatbáziskapcsolat kezelése a MySqlConnection és a MySqlCommand objektumok használatával történik.

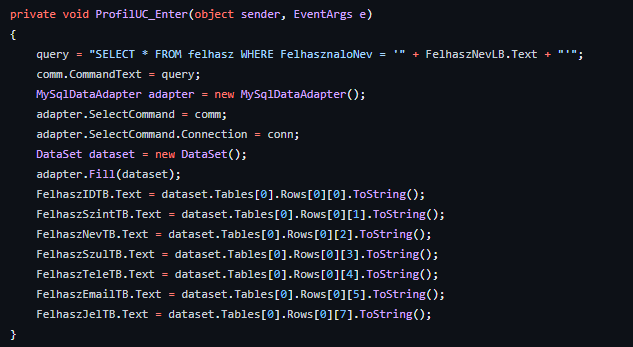


-----------------------------------------------------------------------------------------------------------

Az adatok lekérdezése és frissítése az SQL lekérdezések használatával történik. A profil szerkesztése során a felhasználó által megadott adatokat az SQL UPDATE utasítással frissítik az adatbázisban. A felhasználói felületen található TextBox és Label vezérlők a felhasználói felületre kerülnek és az adatbázisból való lekérdezéskor az adatokat ezekbe a vezérlőkbe helyezik.

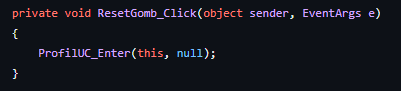


-----------------------------------------------------------------------------------------------------------



Itt mitán az adatbázisból kiszedtem a szükséges információkat, és megadtam a kellő MySql commandokat, minden információt sorban kiiratattam a felhasználó számára.

-----------------------------------------------------------------------------------------------------------



Ezzel a gombal csak vissza resetelteti a felhasználó az oldalt, hogyha esetleg más profilt akarna megtekinteni.

**6.Patikusi szint**

A patikus részlege a kassza programomnak olyan szakasza, amely lehetővé teszi a gyógyszerek, táplálékkiegészítők és más egészségügyi termékek kezelését. A patikus felhasználói szint lehetővé teszi, hogy a patikusok regisztrálhassanak, beléphessenek, és hozzáférhessenek a rendszerben tárolt információkhoz.

A patikusok képesek lesznek kezelni a gyógyszerek állományát, új gyógyszereket hozzáadni a rendszerhez, és nyomon követni a gyógyszerek készletét. Emellett lehetőségük lesz a receptek kezelésére és a vásárlókkal való kapcsolattartásra.

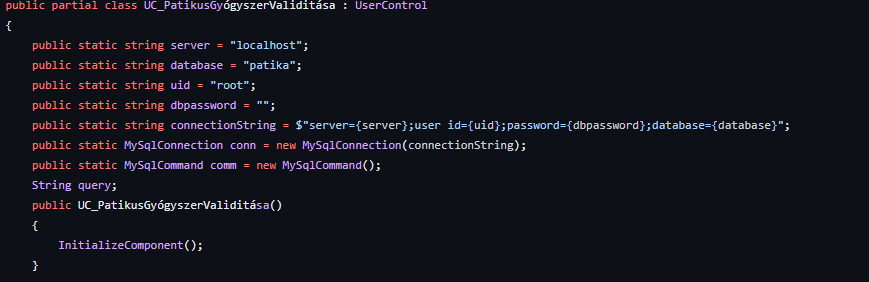
Az Ön programja lehetőséget biztosít arra, hogy a patikusok könnyen kereshessék meg a különböző termékeket az állományban, és nyomon kövessék a gyógyszerek lejárati dátumát. A patikusoknak lehetőségük lesz az ügyféladatok kezelésére, beleértve az ügyfelek adatainak tárolását és szerkesztését.

Az egészségügyi iparban fontos, hogy a gyógyszerek és más termékek kezelése pontos és hatékony legyen. A patikus részleg segítségével a kasszarendszer lehetővé teszi a patikusok számára, hogy hatékonyan és biztonságosan kezeljék az állományt és a recepteket, így a vásárlók biztonságos és minőségi szolgáltatásban részesülnek.

**7.Patikusi szint megismertetése**

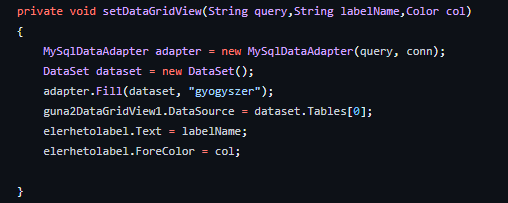
Gyógyszerek Validitása:

Az adatbáziskapcsolatot és adatlekérdezéseket kezelő kódról van szó. A kódrészlet a MySql.Data.MySqlClient és System névtereket használja, amelyek tartalmazzák az adatbáziskapcsolathoz és az adatok lekérdezéséhez szükséges osztályokat és metódusokat.



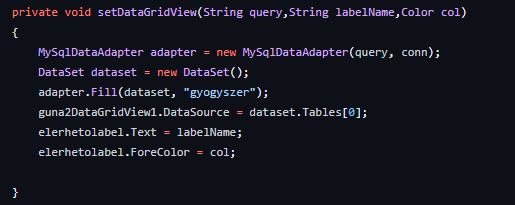
-----------------------------------------------------------------------------------------------------------

A kódban definiálva vannak a szerver, adatbázis, felhasználónév és jelszó adatai, amelyekkel az adatbáziskapcsolat létrejön. Az adatbáziskapcsolatot a connectionString változóban tárolják. A MySqlConnection objektum az adatbáziskapcsolatot reprezentálja, amelyet a kódban a conn változón keresztül érhetünk el.



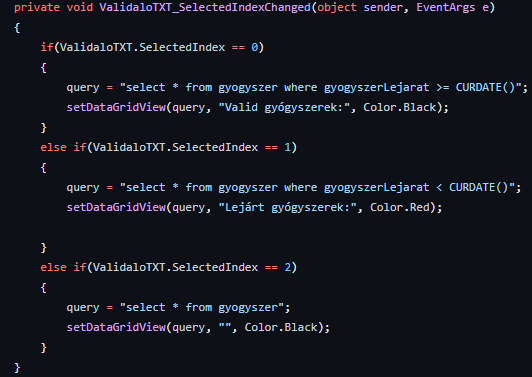
-----------------------------------------------------------------------------------------------------------

Az adatbáziskapcsolatot a setDataGridView metódus használja az adatok lekérdezéséhez. Az adatok lekérdezéséhez használt query változó tartalmazza az SQL utasítást. A metódus a MySqlDataAdapter objektumot használja az adatok lekérdezésére, amelyet a dataset objektummal tölt fel. Az eredményeket a guna2DataGridView1 DataGridView-be helyezi el. Az elérhetolabel feliratot és színét a metódus a paraméterei alapján állítja be.



-----------------------------------------------------------------------------------------------------------

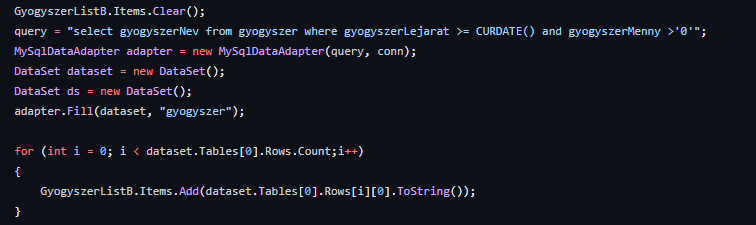
Az ValidaloTXT\_SelectedIndexChanged eseménykezelő az aktuális adatok megjelenítését vezérli a guna2DataGridView1 DataGridView-ben. Az eseménykezelő ellenőrzi, hogy melyik opció van kiválasztva a ValidaloTXT ComboBox-ból, majd az SQL-lekérdezést a kiválasztott opció alapján állítja be, majd meghívja a setDataGridView metódust az eredmények megjelenítéséhez.



-----------------------------------------------------------------------------------------------------------

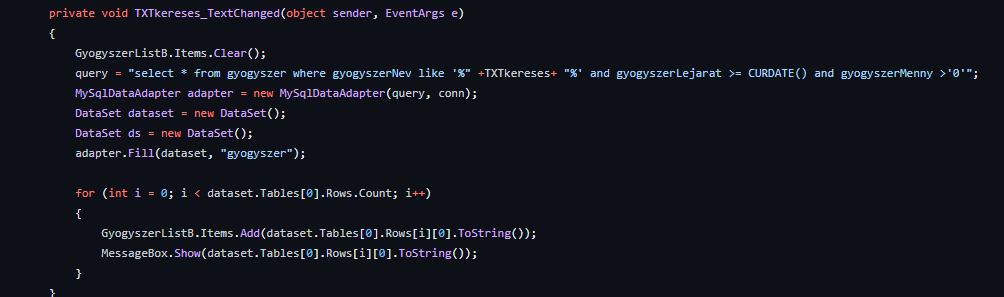
Gyógyszer Eladása:

Az alkalmazásban a patikusok képesek listázni azokat a gyógyszereket, amelyek még nem jártak le és amelyekből még van raktáron.



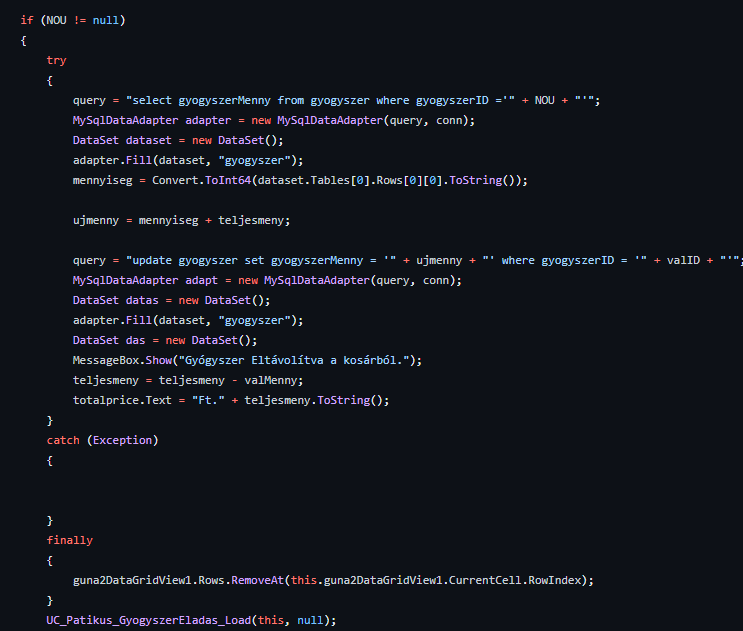
-----------------------------------------------------------------------------------------------------------

A listát lehet keresni a gyógyszer nevére történő kereséssel, és a listában kiválasztott gyógyszer adatai (gyógyszer azonosító, ár, stb.) megjelennek a megfelelő mezőkben.



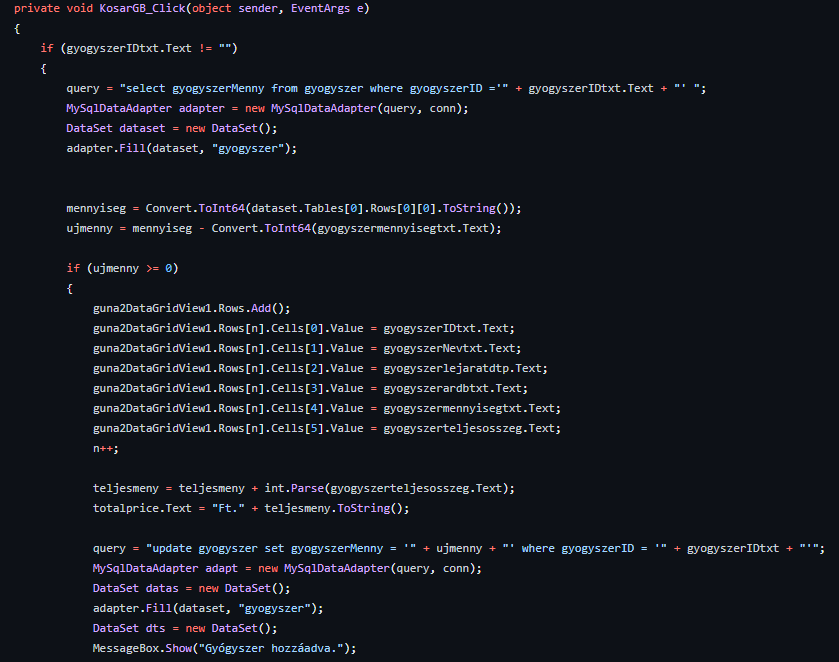
-----------------------------------------------------------------------------------------------------------

A patikus a kijelölt gyógyszerhez megadhatja a vásárolni kívánt mennyiséget, ami automatikusan kiszámolja a vásárlás teljes összegét.



-----------------------------------------------------------------------------------------------------------

Ha a gyógyszerből elegendő mennyiség áll rendelkezésre, akkor a patikus a kosár gomb megnyomásával hozzáadhatja a gyógyszert a kosárhoz, amely megjeleníti a kiválasztott gyógyszerek listáját.



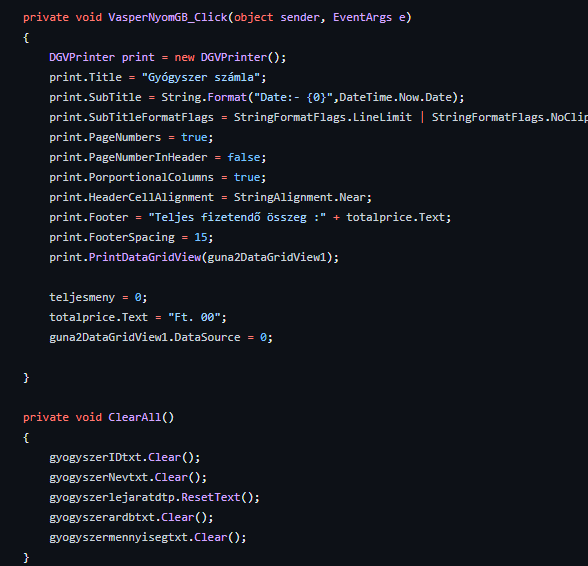
-----------------------------------------------------------------------------------------------------------

A patikus képes a kosár tartalmát kinyomtatni a DGVPrinterHelper használatával.

Itt, a dokumentáció legvégén csatolom majd a használt Printer GitHub linkjét.

-----------------------------------------------------------------------------------------------------------

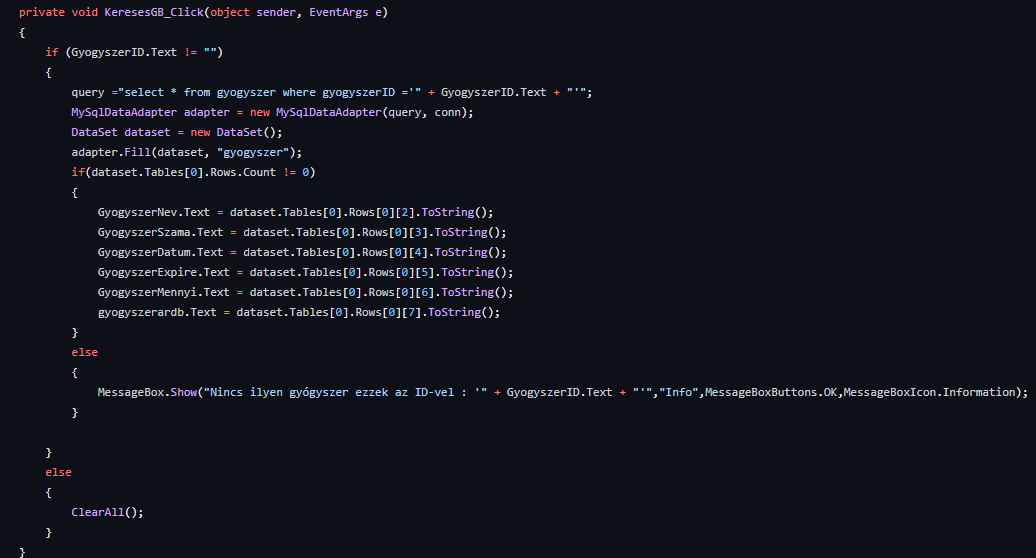
A kód több részre van osztva, és magában foglalja a kapcsolódást az adatbázishoz, a listák kezelését, a gyógyszer adatok megjelenítését a felhasználói felületen, valamint a kosár és a számla nyomtatását.



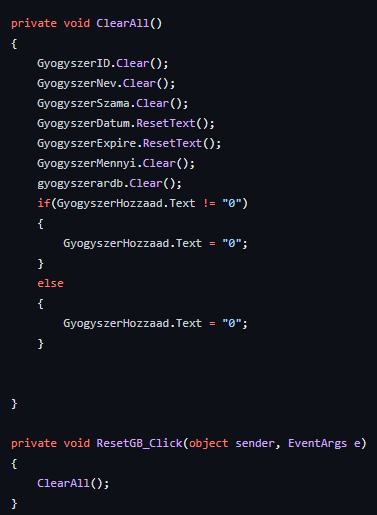
-----------------------------------------------------------------------------------------------------------

Gyógyszer Update(frissítése)

A felhasználói felületen található gombok eseményeit kezeli a kód. A KeresesGB\_Click metódusban a felhasználó által megadott gyógyszer azonosító alapján lekérdezi az adatbázisból a megfelelő gyógyszer adatait, majd kitölti a megfelelő mezőket a felhasználói felületen. Ha az adatbázis nem tartalmazza az adott azonosítójú gyógyszert, akkor egy információs üzenet jelenik meg.

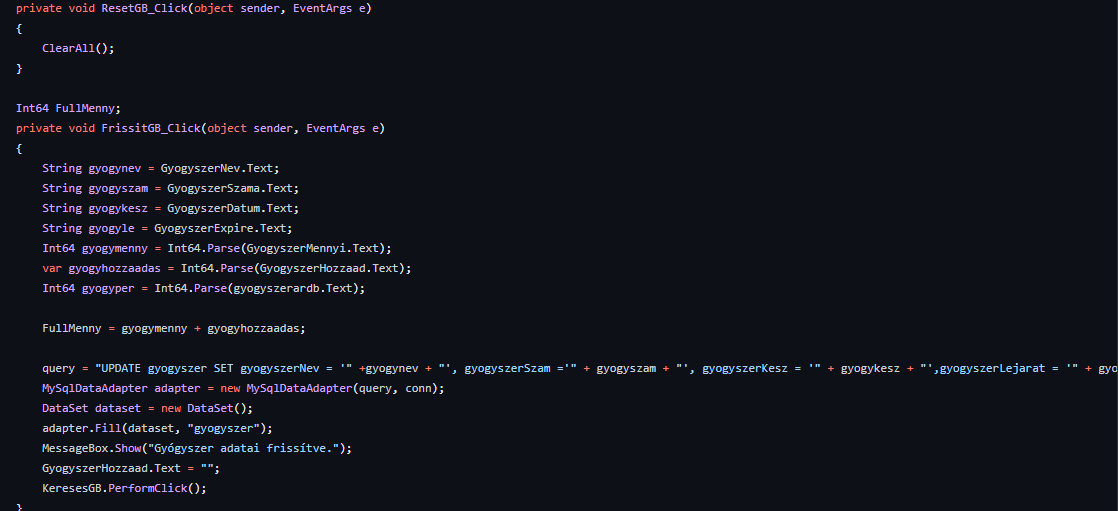
-----------------------------------------------------------------------------------------------------------

A ClearAll metódus kitölti az összes mezőt a felhasználói felületen. A ResetGB\_Click metódus meghívja ezt a metódust.



-----------------------------------------------------------------------------------------------------------

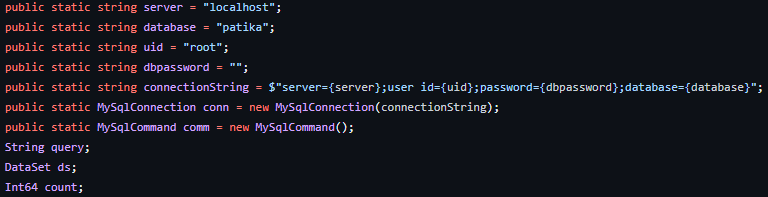
A FrissitGB\_Click metódus a felhasználó által megadott adatok alapján frissíti a megfelelő gyógyszer adatait az adatbázisban. A query stringben megadott SQL utasítással frissíti az adatokat, majd visszajelzést ad a felhasználónak, hogy az adatok frissítése megtörtént.

-----------------------------------------------------------------------------------------------------------

Összességében ez a kód egy egyszerű alkalmazás, amely lehetővé teszi a felhasználó számára a gyógyszerek adatbázisának kezelését, valamint az adatok megjelenítését és frissítését a felhasználói felületen keresztül.

-----------------------------------------------------------------------------------------------------------Patikusi MűszerTábla

Itt, ugyan úgy mint mindegyik másik UserControlnál, hozzá kapcsolódok az adatbázishoz:

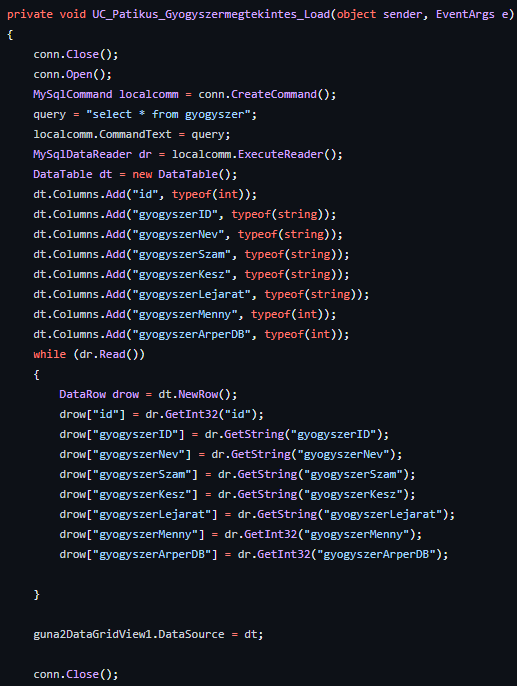


És az ezen az oladoln található oszlopdiagrammot ugyan úgy rácsatlakoztatom.

-----------------------------------------------------------------------------------------------------------

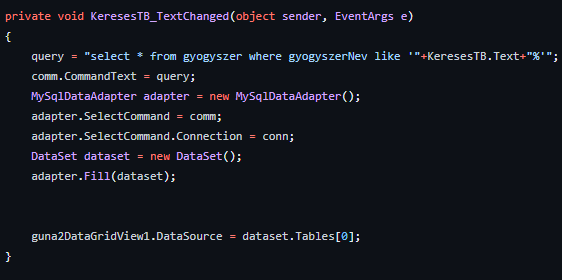
Gyógyszer megtekíntése:

Az alkalmazás egy UserControlt (felhasználói vezérlőt) tartalmaz, amely megjeleníti az összes gyógyszer rekordot egy DataGridView-ban (táblázatos nézetben). Az adatokat az "UC\_Patikus\_Gyogyszermegtekintes\_Load" függvény betölti az adatbázisból a "gyogyszer" táblából. Az adatbáziskapcsolat felépítésehez a "MySql.Data.MySqlClient" és a "System" névtérben definiált szükséges osztályokat használja.



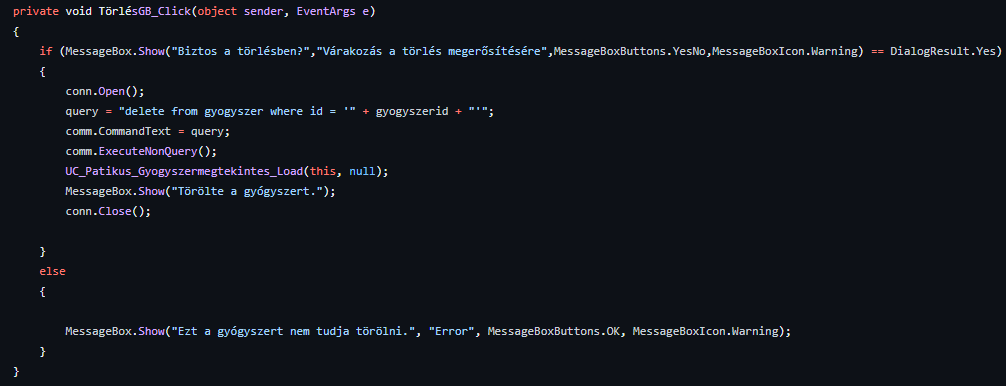
-----------------------------------------------------------------------------------------------------------

Az alkalmazás keresési funkciót is biztosít. A "KeresesTB\_TextChanged" függvény kezeli a keresési mezőben történő változásokat, és frissíti a DataGridView tartalmát a keresési eredményekkel.



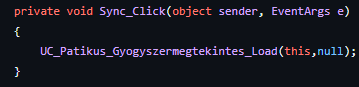
-----------------------------------------------------------------------------------------------------------

Az "TörlésGB\_Click" függvény kezeli a törlési funkciót. Amennyiben a felhasználó megerősíti a törlési szándékát, akkor az adott rekordot törli az adatbázisból, majd frissíti a DataGridView tartalmát a módosított adatokkal.



-----------------------------------------------------------------------------------------------------------

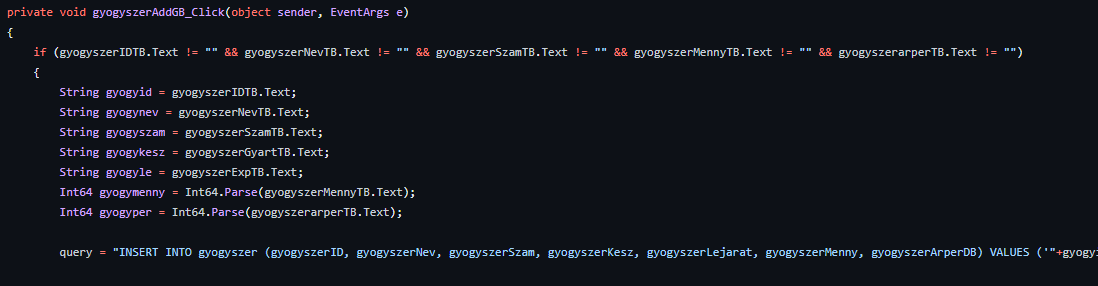
Az "Sync\_Click" függvény az adatok újratöltését végzi el a DataGridView-ban, amely akkor hasznos, ha a felhasználó bármilyen adatmódosítást végez az adatbázisban.



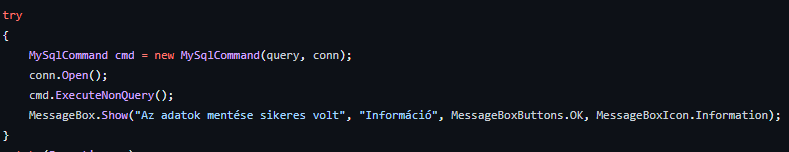
-----------------------------------------------------------------------------------------------------------

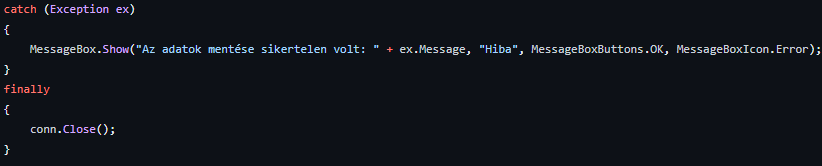
Gyógyszer Hozzáadása:

A program az adatbázis kapcsolatot definiálja a MySQL szerverrel. Az adatok mentéséhez INSERT utasítást használ az adatbázis táblába. Az adatok mentése után a program üzenetet jelenít meg a felhasználó számára az eredményről.

-----------------------------------------------------------------------------------------------------------

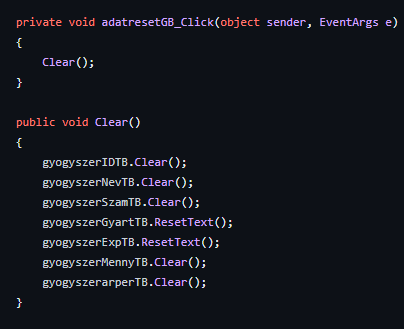
Az adatok helyes mentése esetén az alkalmazás megjeleníti a "Az adatok mentése sikeres volt" üzenetet, ha az adatok mentése nem sikerült, akkor pedig a "Az adatok mentése sikertelen volt" hibaüzenetet jeleníti meg a felhasználó számára.





-----------------------------------------------------------------------------------------------------------

A program az adatok törléséhez a Clear() metódust használja, amely a mezők tartalmát törli.



-----------------------------------------------------------------------------------------------------------

Levezetés

Az alkalmazás kódját az egyes funkciókhoz létrehozott C# osztályok tartalmazzák, amelyek közül sok használja a MySQL adatbázisba való adateléréshez a MySql.Data.MySqlClient névtérben található osztályokat.

A dokumentáció az alkalmazás különböző részeit ismerteti, és részletesen bemutatja a felhasználói felület funkcióit. A dokumentációban ismertetett fontosabb funkciók közé tartozik a gyógyszerek hozzáadása, eltávolítása és módosítása, az áruk kezelése és az értékesítés nyomon követése.

Az alkalmazás fejlesztése során a fejlesztőknek figyelembe kellett venniük a szoftver skálázhatóságát és karbantarthatóságát. Az adatokhoz való hozzáférés és azok kezelése a MySQL adatbázis használatával biztosítja az alkalmazás stabil és hatékony működését. Az alkalmazás azonban csak akkor működik megfelelően, ha a MySQL adatbázis rendelkezésre áll és megfelelően van konfigurálva.

Összességében a dokumentáció részletes áttekintést nyújt az alkalmazás különböző részeiről, működéséről és használatáról, valamint a felhasználók számára számos funkciót kínál a gyógyszerek kezeléséhez és nyomon követéséhez.

**Forrás(ok)**

Nyomtató modul: <https://github.com/vijaythapa333/DGVPrinter>

**Az alkalmazást készítette:**

**Törő László András**