**Michał Kasperek – Sprawozdanie – Aplikacja dla małej kawiarni**

Stworzony przeze mnie program umożliwia, złożenie zamówienia w małej kawiarni.

Na początku użytkownik podaje swoje imię, następnie przystępuje do składania zamówienia – wybiera spośród podanych, interesujący go rodzaj kawy. Program ma za zadanie przedstawić w tabelce, rodzaj wybranego produktu, jego ilość oraz cenę. Następnie policzyć wymaganą kwotę do zapłaty. Po zaksięgowaniu zapłaty, dane zamówienia są przesyłane do tabelki z historią ostatnich zamówień.

Pierwsze uruchomienie aplikacji prezentuje się następująco:

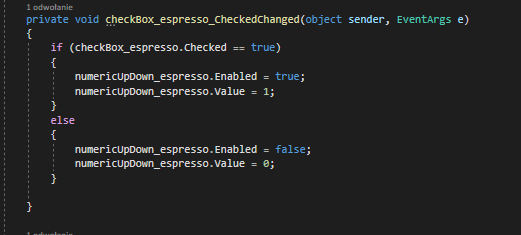


W pierwszej kolejności użytkownik ma za zadanie wpisać swoje imię w przeznaczonym do tego polu.

Następnie mamy możliwość wyboru kawy. Jej wybór odbywa się za pomocą checkbox’ów (domyślnie są odznaczone). Jeżeli użytkownik zdecyduje się zaznaczyć któryś z produktów, obok nazwy kawy odblokuje się możliwość podania ilości jaką chcemy zamówić.

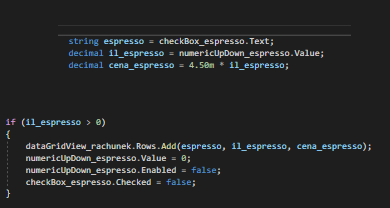
Zrobiłem to za pomocą funkcji if. Jeżeli użytkownik zaznaczy np. kawę espresso, jej ilość zostanie automatycznie zmieniona do 1. Dane są też automatycznie odznaczane po zatwierdzeniu.

Przykładowy kod:

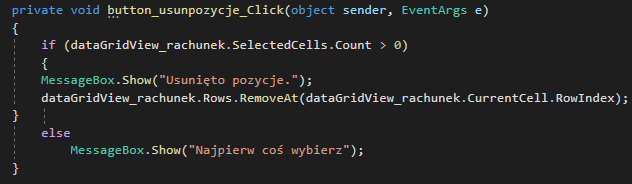


Po zadecydowaniu który produkt chcemy zamówić, klikamy przycisk "Zatwierdź". Ma on za zadanie przepisanie produktów tj. ich nazw, ilości oraz ceny do jednej tabelki.

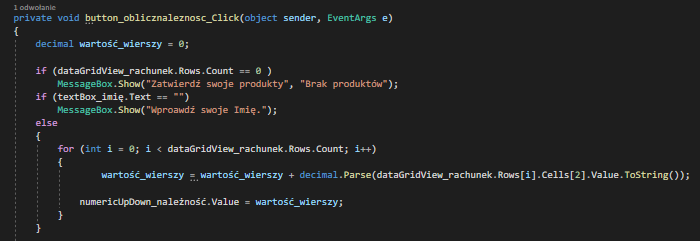
Dane w tabelce wyświetlane są za pomocą funkcji if. Jeżeli użytkownik wybrał jakiś produkt to pojawia się on na liście. Dane są pobierane z checkbox’a oraz z kontrolek numericupdown. A następnie dodane do tabelki za pomocą komendy: **dataGridView\_rachunek.Rows.Add(espresso, il\_espresso, cena\_espresso);**



W tabelce istnieje możliwość usunięcia wybranej pozycji, jeżeli użytkownik pomylił się w trakcie wyboru kawy. Odbywa się to za pomocą przycisku **"Usuń wybraną pozycje".**



Kolejny przycisk – **"Oblicz należność"** za pomocą funkcji if sprawdza czy tabelka nie jest pusta, jeżeli nie to za pomocą funkcji for liczy wybrany wiersz. Następnie obliczona wartość zamówienia wpisywana jest w odpowiednie miejsce.

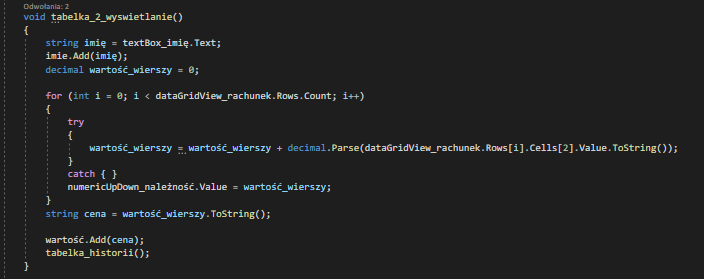


Przycisk ten posiada także odpowiednie zabezpieczania, aby uniknąć pustych wierszy tabelki czy pustego pola tekstowego.

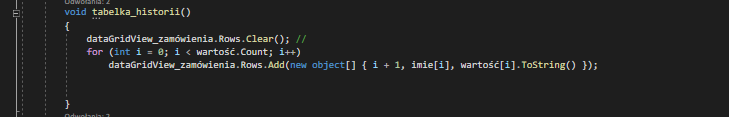
Po obliczeniu należności wymagane jest zapłacić odpowiednią sumę tj. wyświetlaną jako „**do zapłaty”.** Jeżeli podana kwota wpłaty będzie większa bądź równa kwocie do zapłaty, wtedy zamówienie jest opłacone i pojawia się w tabelce historii zamówień wraz z jej wartością. W celu dodawania. Aby dane zostały zapisane w drugiej tabelce stworzyłem do tego metodę tabelka\_2\_wyswietlanie().

*(domyślnie ustawienia wartości kontrolki do zapłaty ustawione są jako zablokowane tzn. nie można jej w żaden sposób edytować.: numericUpDown\_należność.Enabled = false;)*

****



Podana wyżej metoda ma za zadanie pobranie wartości z textobox’a imię a następnie za pomocą funkcji for policzyć liczbę wierszy z pierwszej tabelki. Następnie wywoływana jest kolejna metoda – tabelka\_historri(), która odpowiada za dodanie wyżej wymienionych wartości do tabelki.



Dane z tabelki zamówień są serializowane. W tym celu wykorzystałem kod poznany na zajęciach i odpowiednio go zmodyfikowałem tak aby wszystkie odpowiednie dane zostały zapisane wraz z zakończeniem pracy programu oraz zostały otworzone przy ponownym uruchomieniu aplikacji.

