# Banki Ügyfelek és Bankszámlák Kezelése

## Cél

A feladat célja egy banki ügyfelek és bankszámlák kezelését lehetővé tevő C# konzolalkalmazás létrehozása. Az alkalmazás tartalmaz két osztályt: Customer és BankAccount. Az Customer osztályban tároljuk az ügyfelek adatait, míg a BankAccount osztályban a bankszámlákat kezeljük.

## Customer Osztály

### Adattagok

1. string customerName - Az ügyfél neve.
2. int customerAge - Az ügyfél életkora.
3. string customerAddress - Az ügyfél lakcíme.
4. string customerEmail - Az ügyfél e-mail címe.
5. string customerPhoneNumber - Az ügyfél telefonszáma.

### Metódusok

1. void PrintCustomerInfo() - Kiírja az ügyfél adatait a konzolra.
2. void UpdateCustomerInfo(string newName, int newAge, string newAddress, string newEmail, string newPhoneNumber) - Frissíti az ügyfél adatait az új adatokkal.
3. void PlaceOrder(string product, int quantity) - Feladja az ügyfél rendelését a megadott termékből és mennyiségből.
4. void RequestSupport(string supportRequest) - Rögzíti az ügyfél támogatási kérését.
5. void SendFeedback(string feedback) - Rögzíti az ügyfél visszajelzését.

## BankAccount Osztály

### Adattagok

1. string ownerName - A számlatulajdonos neve.
2. decimal balance - A számla egyenlege.
3. string accountNumber - A bankszámla számlaszáma.
4. string accountType - A bankszámla típusa.
5. DateTime accountOpeningDate - A bankszámla nyitásának dátuma.
6. bool isAccountActive - A bankszámla állapota (aktív vagy inaktív).

### Metódusok

1. void PrintAccountInfo() - Kiírja a bankszámla adatait a konzolra.
2. void CloseAccount() - Lezárja a bankszámlát.
3. void TransferFundsTo(BankAccount destinationAccount, decimal amount) - Pénzátutalást végez a jelenlegi számláról a célbankszámlára a megadott összeggel.
4. void UpdateAccountType(string newAccountType) - Frissíti a bankszámla típusát az új típussal.

## Feladatlépések

1. Hozz létre egy Customer osztályt, és definiáld az adattagokat és metódusokat, amelyek a fenti leírásban vannak felsorolva.
2. Hozz létre egy BankAccount osztályt, és definiáld az adattagokat és metódusokat, amelyek a fenti leírásban vannak felsorolva.
3. Készíts egy konzolalkalmazást, amely létrehoz egy Customer példányt és egy BankAccount példányt.
4. Hozz létre néhány példányosítási és műveleti lépést a Customer és BankAccount példányokkal, és teszteld a metódusokat a megfelelő eredményekkel.
5. Készítsd el a szükséges kimeneti üzeneteket, hogy a felhasználó könnyen követhesse az alkalmazás működését.
6. Győződj meg róla, hogy az osztályok és metódusok helyesen működnek, és a kimeneti üzenetek tartalmazzák az elvárt információkat.
7. Teszteld az alkalmazást a különböző műveletek végrehajtásával, és győződj meg róla, hogy az eredmények helyesek és az alkalmazás megfelel a követelményeknek.
8. Töltsd fel az elkészült kódot egy verziókezelő rendszerbe. (GIT)