using System;

using System.Collections.Generic;

public class BankasKonts

{

private static int nakamaisKontaNumurs = 1000;

private static HashSet<int> izmantotieKontuNumuri = new HashSet<int>();

private int kontaNumurs;

private string ipasnieks;

private decimal atlikums;

public BankasKonts(string vards, decimal saktnauda)

{

this.kontaNumurs = GeneretKontaNumuru();

this.ipasnieks = vards;

this.atlikums = saktnauda;

Console.WriteLine($"\nIzveidots jauns konts #{kontaNumurs} īpašniekam {ipasnieks}");

}

~BankasKonts()

{

Console.WriteLine($"\nKonts #{kontaNumurs} tiek iznīcināts");

izmantotieKontuNumuri.Remove(kontaNumurs);

}

private int GeneretKontaNumuru()

{

int jaunsNumurs = nakamaisKontaNumurs++;

while (izmantotieKontuNumuri.Contains(jaunsNumurs))

{

jaunsNumurs = nakamaisKontaNumurs++;

}

izmantotieKontuNumuri.Add(jaunsNumurs);

return jaunsNumurs;

}

public void Noguldit(decimal summa)

{

if (summa <= 0)

{

Console.WriteLine("Kļūda: Summai jābūt lielākai par 0!");

return;

}

atlikums += summa;

Console.WriteLine($"\nKontā #{kontaNumurs} noguldīti {summa} EUR. Jaunais atlikums: {atlikums} EUR");

}

public void Izņemt(decimal summa)

{

if (summa <= 0)

{

Console.WriteLine("Kļūda: Summai jābūt lielākai par 0!");

return;

}

if (summa > atlikums)

{

Console.WriteLine("Kļūda: Nepietiek līdzekļu kontā!");

return;

}

atlikums -= summa;

Console.WriteLine($"\nNo konta #{kontaNumurs} izņemti {summa} EUR. Atlikums: {atlikums} EUR");

}

public void ParadiInformaciju()

{

Console.WriteLine("\nKonta informācija:");

Console.WriteLine($"Konta numurs: #{kontaNumurs}");

Console.WriteLine($"Īpašnieks: {ipasnieks}");

Console.WriteLine($"Atlikums: {atlikums} EUR");

}

public int DabuKontaNumuru() => kontaNumurs;

public string DabuIpasniekaVardu() => ipasnieks;

public decimal DabuAtlikumu() => atlikums;

}

class Program

{

static List<BankasKonts> konti = new List<BankasKonts>();

static void Main()

{

while (true)

{

Console.WriteLine("\nBankas kontu pārvaldības sistēma");

Console.WriteLine("1. Izveidot jaunu kontu");

Console.WriteLine("2. Noguldīt naudu");

Console.WriteLine("3. Izņemt naudu");

Console.WriteLine("4. Apskatīt konta informāciju");

Console.WriteLine("5. Beigt darbu");

Console.Write("Izvēlieties darbību (1-5): ");

string izvele = Console.ReadLine();

switch (izvele)

{

case "1":

IzveidotKontu();

break;

case "2":

VeiktNoguldijumu();

break;

case "3":

VeiktIzņemšanu();

break;

case "4":

ParadiKontaDatus();

break;

case "5":

Console.WriteLine("\nPaldies par lietošanu!");

return;

default:

Console.WriteLine("\nNepareiza izvēle. Mēģiniet vēlreiz.");

break;

}

}

}

static void IzveidotKontu()

{

Console.Write("\nIevadiet konta īpašnieka vārdu: ");

string vards = Console.ReadLine();

Console.Write("Ievadiet sākotnējo atlikumu (EUR): ");

decimal summa = decimal.Parse(Console.ReadLine());

BankasKonts konts = new BankasKonts(vards, summa);

konti.Add(konts);

}

static BankasKonts AtrastKontu()

{

if (konti.Count == 0)

{

Console.WriteLine("\nNav neviena konta. Vispirms izveidojiet kontu.");

return null;

}

Console.Write("\nIevadiet konta numuru: ");

int numurs = int.Parse(Console.ReadLine());

foreach (var konts in konti)

{

if (konts.DabuKontaNumuru() == numurs)

{

return konts;

}

}

Console.WriteLine("\nKonts ar šādu numuru nav atrasts!");

return null;

}

static void VeiktNoguldijumu()

{

BankasKonts konts = AtrastKontu();

if (konts == null) return;

Console.Write("\nIevadiet noguldāmo summu (EUR): ");

decimal summa = decimal.Parse(Console.ReadLine());

konts.Noguldit(summa);

}

static void VeiktIzņemšanu()

{

BankasKonts konts = AtrastKontu();

if (konts == null) return;

Console.Write("\nIevadiet izņemamo summu (EUR): ");

decimal summa = decimal.Parse(Console.ReadLine());

konts.Izņemt(summa);

}

static void ParadiKontaDatus()

{

BankasKonts konts = AtrastKontu();

if (konts == null) return;

konts.ParadiInformaciju();

}

}