```
using System;
using System.IO;
using System.Linq;

class Program
{
    static string filePath = "accounts.csv";

    static void Main()
    {
        Console.WriteLine("=== Simple Bank System ===");

        while (true)
        {
            Console.WriteLine("\n1. Create Account\n2. Deposit\n3. Withdraw\n4. Check Balance\n5. Exit");
            Console.Write("Your choice: ");
            string choice = Console.ReadLine();

            switch (choice)
            {
                case "1":
                    CreateAccount();
                    break;
                case "2":
                    Deposit();
                    break;
                case "3":
                    Withdraw();
                    break;
                case "4":
```

```
                CheckBalance();
                break;
            case "5":
                Console.WriteLine("Exiting...");
                return;
            default:
                Console.WriteLine("Invalid choice. Please try again.");
                break;
        }
    }
}

static void CreateAccount()
{
    Console.Write("Enter your name: ");
    string name = Console.ReadLine();
    int accountNumber = new Random().Next(100000, 999999);

    using (StreamWriter sw = File.AppendText(filePath))
    {
        sw.WriteLine($"{accountNumber},{name},0");
    }

    Console.WriteLine($"Account created! Your account number is: {accountNumber}");
}

static void Deposit()
{
    Console.Write("Enter account number: ");
    if (!int.TryParse(Console.ReadLine(), out int accountNumber))
    {
```

```
    Console.WriteLine("Invalid account number.");
    return;
}

Console.Write("Enter amount to deposit: ");
if (!decimal.TryParse(Console.ReadLine(), out decimal amount) || amount <= 0)
{
    Console.WriteLine("Invalid amount.");
    return;
}

var lines = File.ReadAllLines(filePath).ToList();
bool found = false;

for (int i = 0; i < lines.Count; i++)
{
    var parts = lines[i].Split(',');
    if (int.Parse(parts[0]) == accountNumber)
    {
        decimal balance = decimal.Parse(parts[2]) + amount;
        lines[i] = $"{parts[0]},{parts[1]},{balance}";
        found = true;
        break;
    }
}

if (found)
{
    File.WriteAllLines(filePath, lines);
    Console.WriteLine("Deposit successful.");
}
```

```
    else
    {
        Console.WriteLine("Account not found.");
    }
}

static void Withdraw()
{
    Console.Write("Enter account number: ");
    if (!int.TryParse(Console.ReadLine(), out int accountNumber))
    {
        Console.WriteLine("Invalid account number.");
        return;
    }

    Console.Write("Enter amount to withdraw: ");
    if (!decimal.TryParse(Console.ReadLine(), out decimal amount) || amount <= 0)
    {
        Console.WriteLine("Invalid amount.");
        return;
    }

    var lines = File.ReadAllLines(filePath).ToList();
    bool found = false;

    for (int i = 0; i < lines.Count; i++)
    {
        var parts = lines[i].Split(',');
        if (int.Parse(parts[0]) == accountNumber)
        {
            decimal balance = decimal.Parse(parts[2]);
            if (balance >= amount)
            {
                balance -= amount;
                lines[i] = $"{parts[0]},{parts[1]},{balance}";
                found = true;
                break;
            }
            else
            {
                Console.WriteLine("Insufficient balance.");
                return;
            }
        }
    }

    if (found)
    {
        File.WriteAllLines(filePath, lines);
        Console.WriteLine("Withdrawal successful.");
    }
    else
    {
        Console.WriteLine("Account not found.");
    }
}

static void CheckBalance()
{
    Console.Write("Enter account number: ");
    if (!int.TryParse(Console.ReadLine(), out int accountNumber))
    {
            Console.WriteLine("Invalid account number.");
            return;
        }

        if (!File.Exists(filePath))
        {
            Console.WriteLine("No account records found.");
            return;
        }

        var lines = File.ReadAllLines(filePath);
        foreach (var line in lines)
        {
            var parts = line.Split(',');
            if (int.Parse(parts[0]) == accountNumber)
            {
                Console.WriteLine($"Account Holder: {parts[1]} | Balance: {parts[2]}");
                return;
            }
        }

        Console.WriteLine("Account not found.");
    }
}
```