| NAME | RAJ SANJAY JADHAV |
| --- | --- |
| CLASS | BE-3-CSE |
| BATCH | B |
| ROLL NO | 512024 |
| PRN | 2019033800129402 |

**PRACTICAL-6(B)**

**GIT REPOSITORY :** [**PRACTICAL-6**](https://github.com/Rajjadhav1710/.NET_Practical/tree/main/Practical-6)

**GENERIC SORTER**

**CODE:**

**using System;**

**class Sorter{**

**public static void selectionSort<T>(T []arr,Func<T,T,bool> compare)**

**{**

**int n = arr.Length;**

**for (int i = 0; i < n - 1; i++)**

**{**

**int min\_idx = i;**

**for (int j = i + 1; j < n; j++)**

**if (compare(arr[j],arr[min\_idx]))//arr[j] < arr[min\_idx]**

**min\_idx = j;**

**T temp = arr[min\_idx];**

**arr[min\_idx] = arr[i];**

**arr[i] = temp;**

**}**

**}**

**}**

**class Employee{**

**string empName;**

**int empId;**

**double empSalary;**

**int age;**

**public Employee(int empId,string empName,double empSalary,int age){**

**this.empId=empId;**

**this.empName=empName;**

**this.empSalary=empSalary;**

**this.age=age;**

**}**

**public static bool compareEmpsById(Employee x,Employee y){**

**if(x.empId<y.empId)**

**return true;**

**else**

**return false;**

**}**

**public static bool compareEmpsBySalary(Employee x,Employee y){**

**if(x.empSalary<y.empSalary)**

**return true;**

**else**

**return false;**

**}**

**public static bool compareEmpsByAge(Employee x,Employee y){**

**if(x.age<y.age)**

**return true;**

**else**

**return false;**

**}**

**public override string ToString()**

**{**

**return $"Employee Id:{this.empId} Name:{this.empName} Salary:{this.empSalary} Age:{this.age}";**

**}**

**}**

**class Driver{**

**// Prints the array**

**static void printArray<T>(T []arr)**

**{**

**int n = arr.Length;**

**for (int i=0; i<n; ++i)**

**Console.WriteLine(arr[i]);**

**Console.WriteLine();**

**}**

**static bool compareInts(int x,int y){**

**if(x<y)**

**return true;**

**else**

**return false;**

**}**

**public static void Main(){**

**Console.WriteLine("Sorting Int Array...");**

**int []arr = {64,25,12,22,11};**

**Sorter.selectionSort<int>(arr,compareInts);**

**Console.WriteLine("Sorted array:");**

**printArray<int>(arr);**

**Employee []Emps=new Employee[5];**

**Emps[0]=new Employee(7,"Joe",10000,23);**

**Emps[1]=new Employee(2,"Mark",2000,25);**

**Emps[2]=new Employee(5,"Sam",20000,32);**

**Emps[3]=new Employee(1,"Virat",5000,22);**

**Emps[4]=new Employee(10,"Karan",55000,35);**

**Console.WriteLine("Sorting Employees By Age:\n");**

**Sorter.selectionSort<Employee>(Emps,Employee.compareEmpsByAge);**

**Console.WriteLine("Sorted array:");**

**printArray<Employee>(Emps);**

**Console.WriteLine("Sorting Employees By Id:\n");**

**Sorter.selectionSort<Employee>(Emps,Employee.compareEmpsById);**

**Console.WriteLine("Sorted array:");**

**printArray<Employee>(Emps);**

**Console.WriteLine("Sorting Employees By Salary:\n");**

**Sorter.selectionSort<Employee>(Emps,Employee.compareEmpsBySalary);**

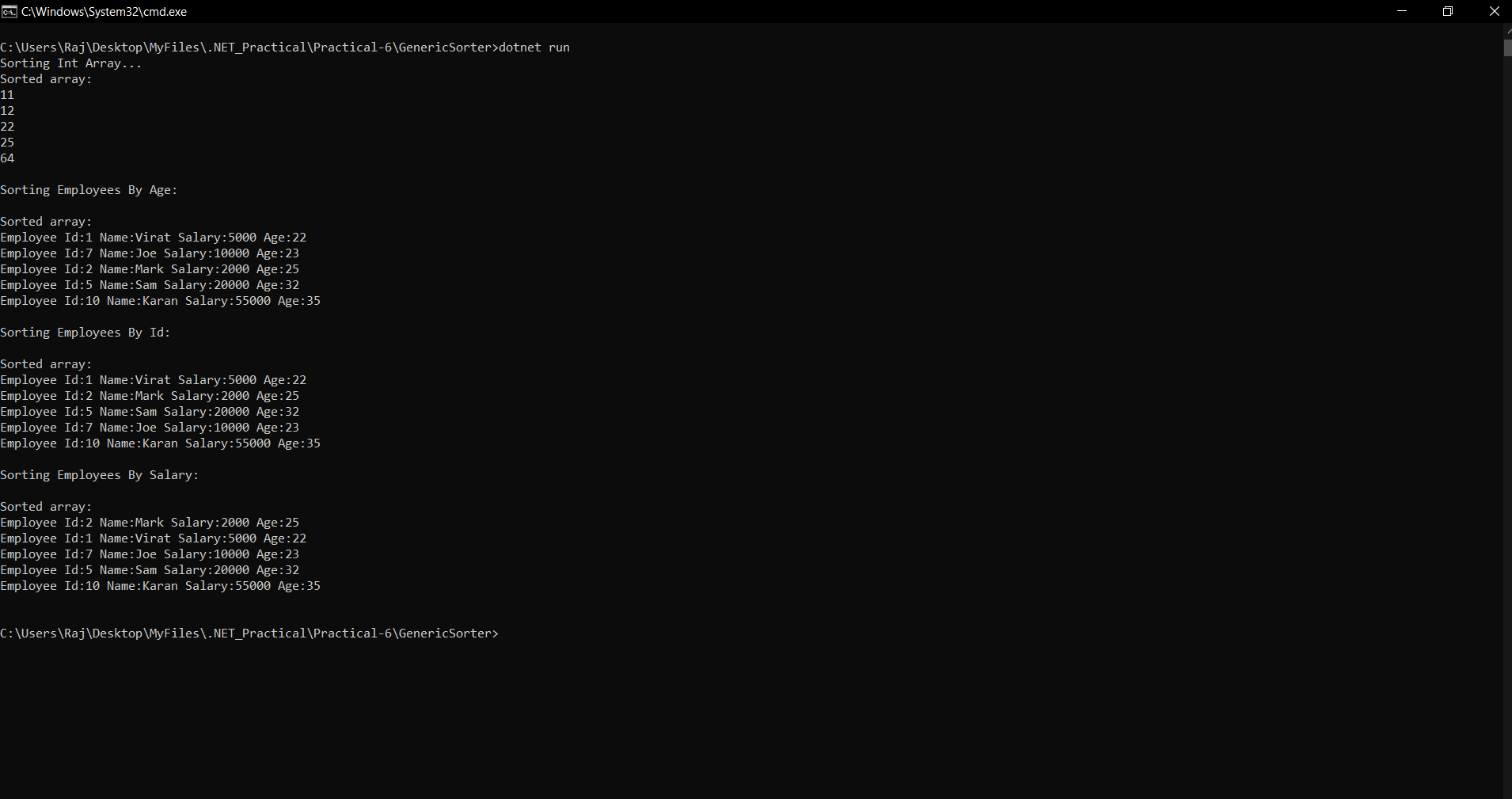
**Console.WriteLine("Sorted array:");**

**printArray<Employee>(Emps);**

**}**

**}**

**OUTPUT:**

****

**DELEGATES AND LAMBDA EXPRESSIONS**

**CODE:**

**using System;**

**class Account{**

**public double balance;**

**public delegate void Del();**

**public Del checkBalance;**

**public Account(double balance){**

**this.balance=balance;**

**this.checkBalance=()=>{**

**if(this.balance<0){**

**Console.WriteLine($"\nBalance:{this.balance} you are overdrawn");**

**}else if(this.balance<10){**

**Console.WriteLine($"\nBalance:{this.balance} your account is very low");**

**}else if(this.balance<100){**

**Console.WriteLine($"\nBalance:{this.balance} watch your spending carefully");**

**}else{**

**Console.WriteLine($"\nBalance:{this.balance} you have over $100 in your account");**

**}**

**};**

**}**

**}**

**class Driver{**

**public static void Main(){**

**Account ac=new Account(200);**

**ac.checkBalance();**

**ac.balance=50;**

**ac.checkBalance();**

**ac.balance=5;**

**ac.checkBalance();**

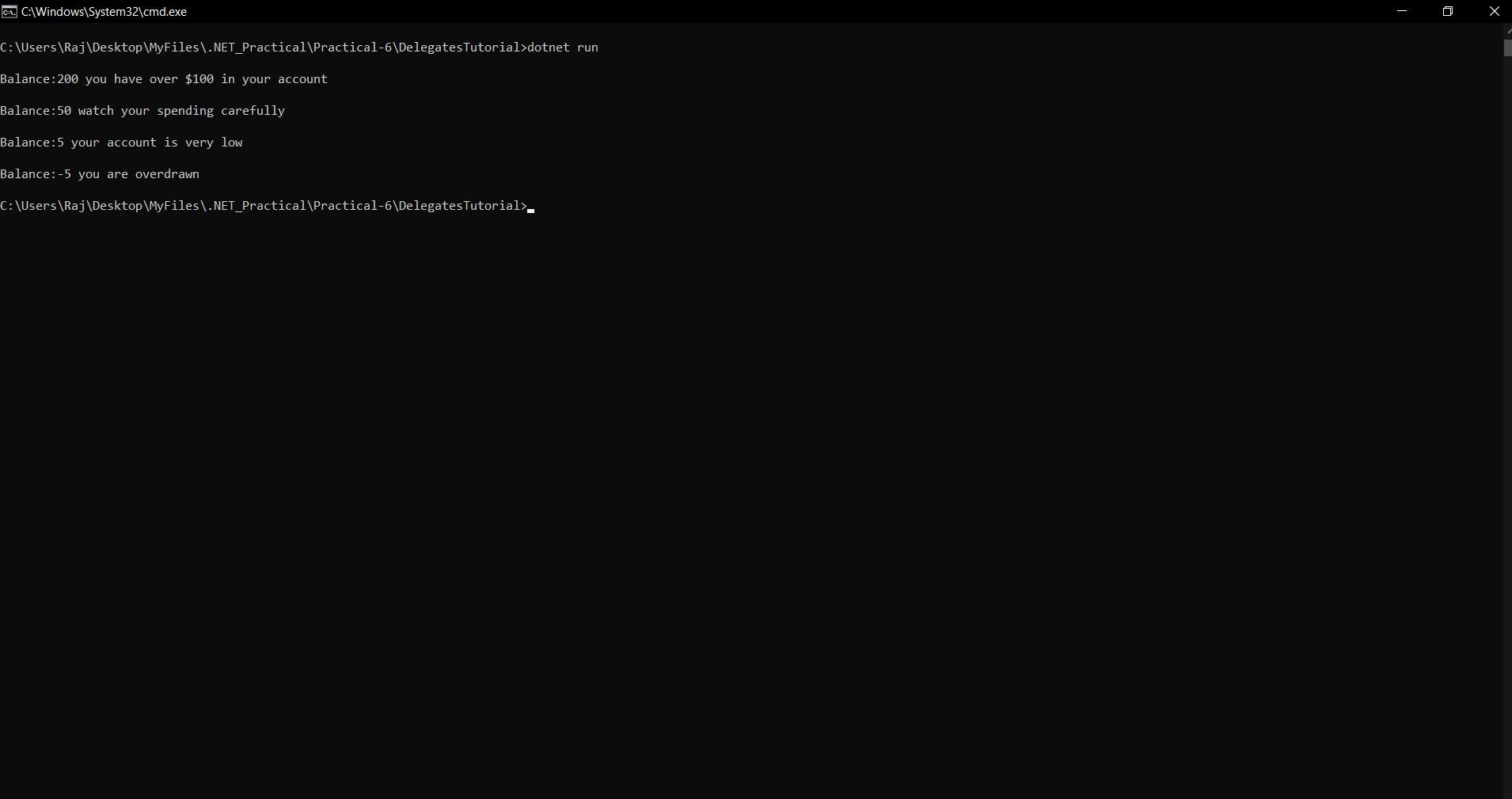
**ac.balance=-5;**

**ac.checkBalance();**

**}**

**}**

**OUTPUT:**

****

**CODE:**

**using System;**

**class Driver{**

**public static void Main(){**

**//1**

**Console.WriteLine("\n1st Code Snippet:-\n");**

**double x,y;**

**Console.Write("\nEnter Value For x:");**

**x=Convert.ToDouble(Console.ReadLine());**

**Console.Write("\nEnter Value For y:");**

**y=Convert.ToDouble(Console.ReadLine());**

**var parse = (double x, double y) => (x > y ? x : y);**

**Console.WriteLine("\nGreater Value:"+parse(x,y));**

**//2**

**Console.WriteLine("\n2nd Code Snippet:-\n");**

**Func<double, double, double> f =(x, y) => {**

**if (x > y)**

**return x;**

**return y;**

**};**

**double z = f(10, 20);**

**Console.WriteLine("\nGreater Value Between 10 And 20:"+z+"\n");**

**//3**

**// double z;**

**// Func<double, double, double> f;**

**Console.WriteLine("\n3rd Code Snippet:-\n");**

**f = (x, y) => {**

**if (x > y)**

**return x;**

**return y;**

**};**

**z = f(10, 20);**

**Console.WriteLine("\nGreater Value Between 10 And 20:"+z+"\n");**

**// z holds 20.**

**f = (x, y) => {**

**if (x < y)**

**return x;**

**return y;**

**};**

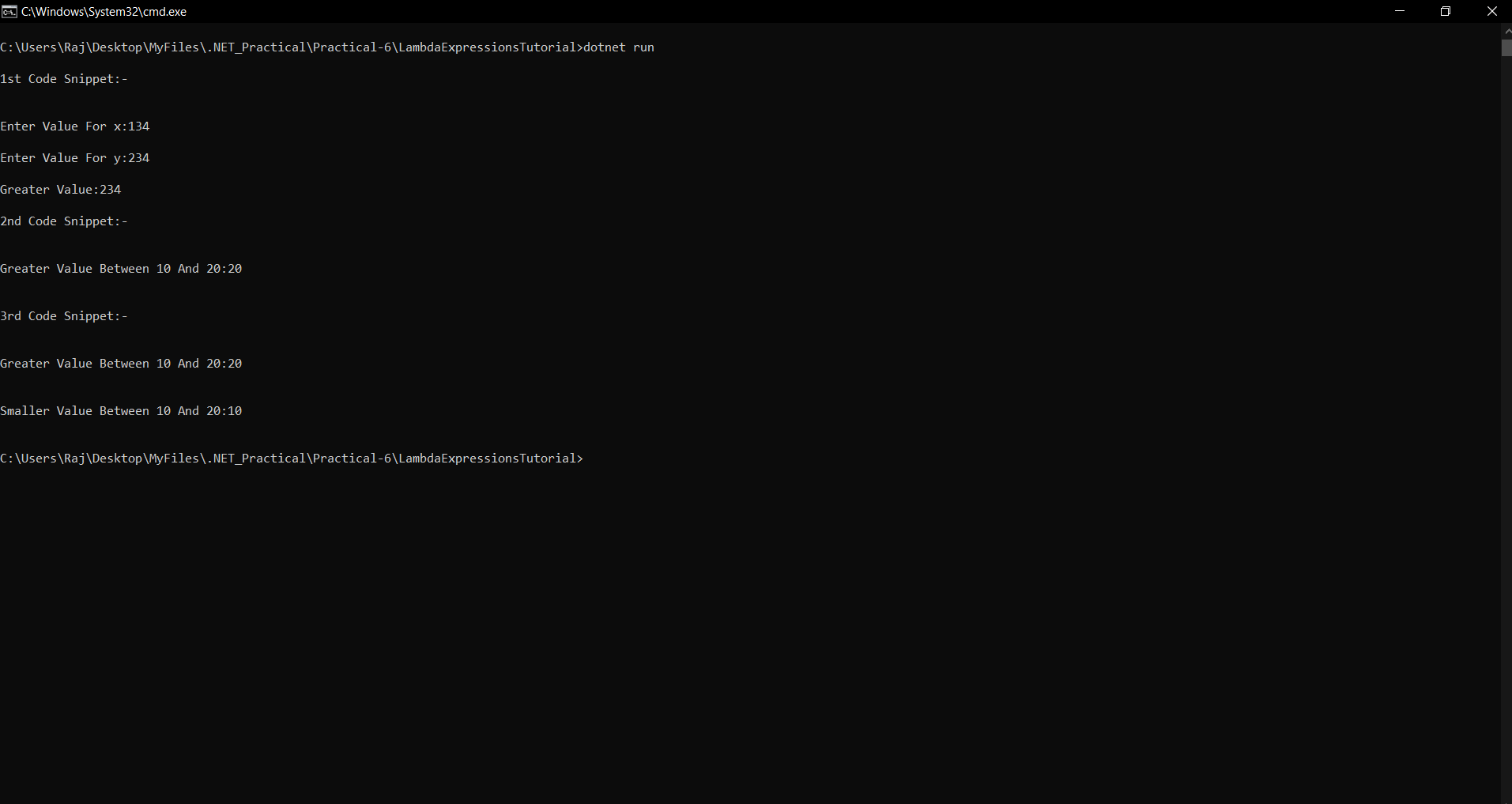
**z = f(10, 20);**

**Console.WriteLine("\nSmaller Value Between 10 And 20:"+z+"\n");**

**}**

**}**

**OUTPUT:**

****