/*A car service center accepts up to 5 cars at a time for service. Each car is serviced in the order it arrived

last in, first out (LIFO) — the latest car to enter is serviced first (stack behavior). If the service bay is

full, new cars are asked to wait. When a car is done, it is removed from the top of the stack.

Add a car to the service bay.

Remove the last car (serviced).

Display all cars currently in service.

Use menu-based service (Stack using array implementation)*/

```
package julyhometask;
import java.util.*;
class CarStack
{
	int car_id;
	String car_name;
	CarStack(int car_id,String car_name)
	{
		this.car_id=car_id;
		this.car_name=car_name;
	}
	void display()
	{
		System.out.println("Car ID: "+car_id+"\nCar name:"+car_name);
	}
}
public class july17ht
{
	static int top=-1;
	static final int size=5;
	static CarStack[]car_list=new CarStack[size];
```

```
	public static void push(CarStack c)
	{
		if(top==size-1)
		{
			System.out.println("The car bay is full cannot able to service the new car..Try it late...");

		}
		else
		{
			top=top+1;
			car_list[top]=c;
			System.out.println("The car with car id "+c.car_id+" is taken into the service bay successfully");


		}
	}
	public static void pop()
	{
		if(top==-1)
		{
			System.out.println("No cars are availabe in the bay");

		}
		else
		{
			System.out.println("The car in the bay is serviced and now successfully out of the bay");
			car_list[top--].display();
		}
	}
```

```
	public static void display()
	{
		if(top==-1)
		{
			System.out.println("There are no cars in the bay");


		}
		else
		{
			System.out.println("The cars in the bay are:");
			for(int i=top;i>=0;i--)
			{
				car_list[i].display();
			}
		}
	}

	public static void main(String[] args)
	{
		Scanner sc=new Scanner(System.in);
		int choice;
		do
		{
			System.out.println("1.Add new car\n2.Take the serviced car\n3.View the cars in the bay\n4.Exit");
			System.out.println("Choose any operation");
			choice=sc.nextInt();
			sc.nextLine();
			switch(choice)
			{
			case 1:
```

```
                    System.out.println("Enter the id of the car brand");
                    int car_id=sc.nextInt();
                    sc.nextLine();
                    System.out.println("Enter the brand name of the car");
                    String car_name=sc.next();
                    push(new CarStack(car_id,car_name));
                    break;
                case 2:
                    pop();
                    break;
                case 3:
                    display();
                    break;
                case 4:
                    System.out.println("System is exiting now---");
                    break;
                default:
                    System.out.println("Enter any valid choice");
                    break;


                }

            }while(choice!=4);
        }

}
```