**Question 1 Answer:**

class Car{

private int year;

private String make;

private double speed;

Car(int year,String make,double beg\_speed){

this.year=year;

this.make=make;

this.speed=beg\_speed;

}

public int getYear(){

return this.year;

}

public String getMake(){

return this.make;

}

public double getSpeed(){

return this.speed;

}

public void Accelerate(){

this.speed++;

}

}

class Racetrack{

public static void main(String... args){

Car c = new Car(2000,"Creta",120.0);

System.out.println("Car Make Year: "+c.getYear());

System.out.println("Car Make: "+c.getMake());

System.out.println("Car Speed Before: "+c.getSpeed());

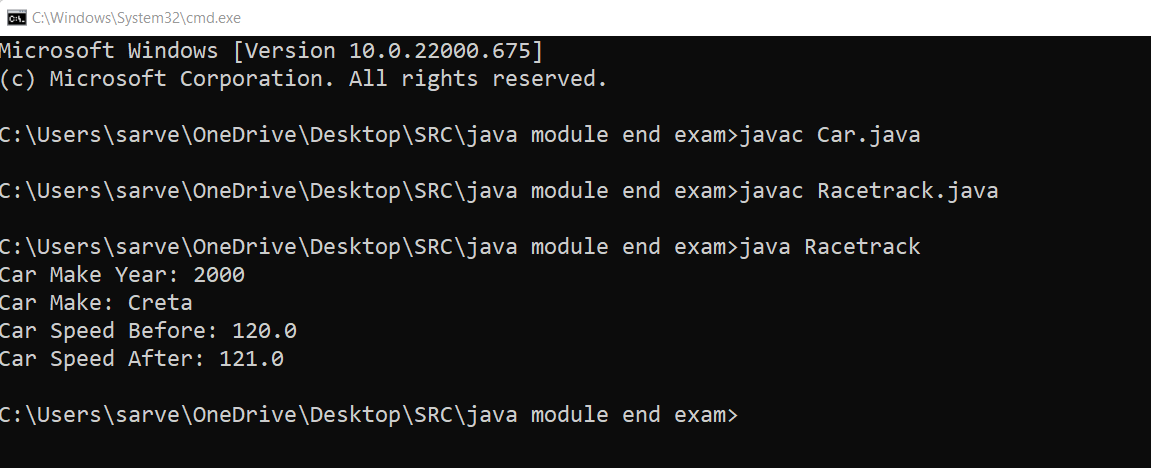
c.Accelerate();

System.out.println("Car Speed After: "+c.getSpeed());

}

}

**OUTPUT:**



**Question 2:-**

import java.util.\*;

class Item{

Integer itemId;

String itemName;

Item(int itemId,String itemName ){

this.itemName=itemName;

this.itemId=itemId;

}

Item(){}

void setitemId(int itemId){

this.itemId=itemId;

}

void setitemName(String itemName){

this.itemName=itemName;

}

public String toString(){

return this.itemId+" "+this.itemName;

}

@Override

public boolean equals(Object o){

if(o instanceof Item){

Item i = (Item) o;

if(this.itemId.equals(i.itemId) &&

this.itemName.equals(i.itemName)

)

{

return true;

}

}

return false;

}

@Override

public int hashCode(){

int prime = 13;

int val = 1;

val = val\*prime + this.itemId.hashCode();

val = val\*prime + this.itemName.hashCode();

return val;

}

}

class namesort implements Comparator<Item>{

public int compare(Item I1,Item I2){

return I1.itemName.compareTo(I2.itemName);

}

}

class idsort implements Comparator<Item>{

public int compare(Item I1,Item I2){

return I1.itemId-(I2.itemId);

}

}

class Inventory{

static Item I=new Item();

static ArrayList<Item> list=new ArrayList<> ();

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

int choice;

Item I1=new Item(1,"Ajay");

Item I2=new Item(4,"Vijay");

Item I3=new Item(3,"Jay");

Item I4=new Item(9,"Ramesh");

Item I5=new Item(8,"Suresh");

list.add(I1);

list.add(I2);

list.add(I3);

list.add(I4);

list.add(I5);

do{

System.out.println("Enter your choice ==");

System.out.println("1) Add Item.\n2) Display complete inventory in sorted order of item names as well as itemId.\n3) Remove Item.\n4) Exit");

choice=sc.nextInt();

switch(choice){

case 1:

System.out.println("Enter your details as follows");

System.out.println("Enter Item you want add");

//int n=sc.nextInt();

for(int i=1;i<=1;i++){

System.out.println("Enter id of item ");

int d=sc.nextInt();

I.setitemId(d);

System.out.println("Enter name of item ");

sc.nextLine();

String ss=sc.nextLine();

I.setitemName(ss);

if(!list.contains(I)){

list.add(I);

}

}

System.out.println("Added items as follows");

System.out.println(list);

break;

case 2:

System.out.println("before sorting");

System.out.println(list);

System.out.println("Sorting by id");

idsort n2=new idsort();

Collections.sort(list,n2);

System.out.println(list);

System.out.println("Sorting by name");

namesort n1=new namesort();

Collections.sort(list,n1);

System.out.println(list);

break;

case 3:

System.out.println("List as follows");

System.out.println(list);

System.out.println("Enter index od item which you want to remove index start from 0");

int re=sc.nextInt();

list.remove(re);

System.out.println("List after removal");

System.out.println(list);

break;

case 4:

System.out.println("Thank you");

break;

}

}while(choice!=4);

}

}

**OUTPUT:**

