**UML Diagram:**

|  |
| --- |
| **Class: Car** |
| - yearModel: int  - make: String  - speed: int |
| + Car()  + Car(yearModel: int, make: String, speed: int)  + getYearModel(): int  + getMake(): String  + getSpeed(): int  + setSpeed(int speed):void  + accelerate(): void  + brake(): void  +toString():String |

**Methods Program:**

/\*\*

\* @author Jay Patel

\* @version 1.0

\*/

public class Car {

private int yearModel;

private String make;

private int speed;

/\*\*

\*

\* @param yearModel -int value,speed- int value

\*/

Car(int yearModel) {

this.yearModel = 1;

speed = 0;

make = "nomake";

}

/\*\*

\*

\* @param yearModel- int value

\* @param make - String value

\* @param speed -int value

\*/

Car(int yearModel, String make,int speed) {

this.yearModel = yearModel;

this.make = make;

setSpeed(speed);

}

/\*\*

\*

\* @return yearModel

\*/

public int getYearModel() {

return yearModel;

}

/\*\*

\*

\* @return make

\*/

public String getMake() {

return make;

}

public int getSpeed() {

return speed;

}

/\*\*

\*

\* @param speed - int

\*/

public void setSpeed(int speed){

if (speed < 0) {

throw new IllegalArgumentException("Sorry! Speed cannot be -ve");

} else {

this.speed = speed;

}

}

public void accelerate() {

speed = speed + 3;

System.out.println("Your Current Speed is: " +speed+ " mph");

}

public void brake() {

speed = speed - 3;

System.out.println("Your Current Speed is: " +speed +" mph");

}

public String toString(){

return "The Maker Of The Car is: " +make+ "\nThe Current Speed Of The Car is: " + speed

+"\nThe Year Model Of The Car is :"+ yearModel;

}

}

**Main Program:**

import java.util.Scanner;

public class TestCar {

public static void main(String[] args) {

Scanner input=new Scanner(System.in);

int yearModel=0,speed=0;

String make=null;

System.out.println("enter the year model");

yearModel=input.nextInt();

System.out.println("enter the current speed of the car");

speed=input.nextInt();

System.out.println("enter maker of the car");

make=input.next();

try {

Car myCar= new Car(yearModel,make,speed);

System.out.println(myCar.toString());

myCar.accelerate();

myCar.accelerate();

myCar.accelerate();

System.out.println();

myCar.brake();

myCar.brake();

myCar.brake();

}catch (IllegalArgumentException msg){

System.out.println(msg.getMessage());

}

}

}

**Screenshot:**

