

※概要欄から提供プログラムをダウンロード！ ↓

## ▶ ポイント

1. オーバーロード
2. コンストラクタ
3. アクセス修飾子、public、private
4. static 変数、static メソッド



## ▶ 問題

## 1

自動車进行管理する設計図クラスとして Car1 クラスを定義します。Car1 クラスがもつメンバ変数とメソッドは以下の表のとおりです。

メンバ変数	
int no	自動車のナンバー
int speed	自動車の速度
メソッド	
void setNo(int n)	変数 no に引数 n を設定する
void run(int s)	変数 speed に引数 s を設定する
void brake()	変数 speed に 0 を設定する
void brake(int s)	変数 speed から引数 s を減算する
void display()	変数 no、変数 speed の値を表示する

Car1.java のコメント内容にあわせてコードを追加して、プログラムを完成させてください。実行用クラスが含まれている DriveCar1.java はそのまま使うものとします。

## 実行例

```
>java DriveCar1
ナンバー2525 の速度は 30 です
ナンバー2525 の速度は 20 です
```

## Car1.java

```
1 | class Car1{
2 |     int no;
3 |     int speed;
4 |
5 |     void setNo(int n){
6 |         no = n;
7 |     }
8 |     void run(int s){
9 |         speed = s;
10 |    }
11 |    // brake メソッドを定義する
12 |
13 |    void display(){
14 |        System.out.println("ナンバー"+no+"の速度は"+speed+"です");
15 |    }
16 | }
```

## DriveCar1.java

```
1 | class DriveCar1{
2 |     public static void main(String[] args){
3 |         Car1 c1 = new Car1();
4 |         c1.setNo(2525);
5 |         c1.run(30);
6 |         c1.display();
7 |
8 |         c1.brake(10);
9 |         c1.display();
10 |    }
11 | }
```

# 2

問題1で作成した Car1 クラスにコンストラクタを追加して Car2 クラスを作成します。Car1.java をコピーして Car2.java を作成し、クラス名を Car2 に変更してください。

Car2 クラスがもつメンバ変数とメソッドは以下の表のとおりです。自動車のナンバーは一度設定したら変更しないものと考えて setNo メソッドは削除します。

メンバ変数	
int no	自動車のナンバー
int speed	自動車の速度
コンストラクタ	
Car2()	変数 no に 0 を設定する
Car2(int n)	変数 no に引数 n を設定する
メソッド	
void run(int s)	変数 speed に引数 s を設定する
void brake()	変数 speed に 0 を設定する
void brake(int s)	変数 speed から引数 s を減算する
void display()	変数 no、変数 speed の値を表示する

実行用クラスが含まれている DriveCar2.java はそのまま使うものとします。

## DriveCar2.java

```
1 | class DriveCar2{
2 |     public static void main(String[] args){
3 |         Car2 c1 = new Car2();
4 |         c1.run(30);
5 |         c1.display();
6 |
7 |         Car2 c2 = new Car2(2525);
8 |         c2.run(50);
9 |         c2.display();
10 |    }
11 | }
```

## 実行例

```
>java DriveCar2
ナンバー0 の速度は 30 です
ナンバー2525 の速度は 50 です
```

# 3

問題2で作成した Car2 クラスにアクセス修飾子を追加して Car3 クラスを作成します。Car2.java をコピーして Car3.java を作成し、クラス名を Car3 に変更してください。

Car3 クラスがもつメンバ変数とメソッドは以下の表のとおりです。変数はすべて private、コンストラクタとメソッドはすべて public にすることでカプセル化を実現します。

メンバ変数	
<b>private</b> int no	自動車のナンバー
<b>private</b> int speed	自動車の速度
コンストラクタ	
<b>public</b> Car3()	変数 no に 0 を設定する
<b>public</b> Car3(int n)	変数 no に引数 n を設定する
メソッド	
<b>public</b> void run(int s)	変数 speed に引数 s を設定する
<b>public</b> void brake()	変数 speed に 0 を設定する
<b>public</b> void brake(int s)	変数 speed から引数 s を減算する
<b>public</b> void display()	変数 no、変数 speed の値を表示する

実行用クラスが含まれている DriveCar3.java はそのまま使うものとします。

## DriveCar3.java

```
1 | class DriveCar3{
2 |     public static void main(String[] args){
3 |         Car3 c1 = new Car3();
4 |         c1.run(30);
5 |         c1.display();
6 |
7 |         Car3 c2 = new Car3(2525);
8 |         c2.speed = 50;
9 |         // c2.run(50);
10 |        c2.display();
11 |    }
12 | }
```

Car3 クラスを作成してから DriveCar3.java をコンパイルすると、DriveCar3.java の 8 行目で private のメンバ変数にアクセスしているため次のようなエラーが表示されます。

### コンパイル例

```
>javac DriveCar3.java
```

```
DriveCar3.java:8: エラー: speed は Car3 で private アクセスされます
```

```
    c2.speed = 50;
```

```
        ^
```

```
エラー1 個
```

DriveCar3.java の 8 行目をコメントアウトして 9 行目の//を削除すると、次のようにコンパイル・実行ができます。

### コンパイル・実行例

```
>javac DriveCar3.java
```

```
>java DriveCar3
```

```
ナンバー0 の速度は 30 です
```

```
ナンバー2525 の速度は 50 です
```

# 4

自動車を表すクラスとして Car4 クラスと DriveCar4 クラスを定義しました。3つのインスタンスを生成していますが、実行すると生成した台数がすべて「1 台」と表示されてしまいました。どこをどのように直せばいいのでしょうか。

(誤っている) 実行例

```
>java DriveCar4
ナンバーなしを作りました
1 台作成済です
ナンバー2525 を作りました
1 台作成済です
ナンバー8888 を作りました
1 台作成済です
```

Car4.java

```
1 | class Car4{
2 |     private int no;
3 |     private int count = 0;
4 |
5 |     public Car4(){
6 |         no = 0;
7 |         count++;
8 |         System.out.println("ナンバーなしを作りました");
9 |     }
10 |    public Car4(int n){
11 |        no = n;
12 |        count++;
13 |        System.out.println("ナンバー" + no + "を作りました");
14 |    }
15 |    public void display(){
16 |        System.out.println(count + "台作成済です");
17 |    }
18 | }
```

## DriveCar4.java

```
1 | class DriveCar4{  
2 |     public static void main(String[] args){  
3 |         Car4 c1 = new Car4();  
4 |         c1.display();  
5 |         Car4 c2 = new Car4(2525);  
6 |         c2.display();  
7 |         Car4 c3 = new Car4(8888);  
8 |         c3.display();  
9 |     }  
10 | }  
11 |
```

(正しい) 実行例

>*java DriveCar4*

ナンバーなしを作りました

1 台作成済です

ナンバー2525 を作りました

2 台作成済です

ナンバー8888 を作りました

3 台作成済です



＼フリーラーニング（無料で学べる場）をもっと広げたい！／  
チャンネル登録や拡散よろしくお願いします！



NEXT DOOR

