

JavaプログラミングⅡ

4回目 クラスの機能（2）コンストラクタ、クラス変数、クラスメソッド 課題

確認○×問題

次の各文は正しいか誤っているか答えなさい。

- (1) コンストラクタはメソッドと同様に戻り値をもつ
- (2) コンストラクタはオブジェクトが生成されると最初に実行される
- (3) コンストラクタはメソッドと同様にオーバーロードができる
- (4) コンストラクタは常に public メンバとしなければならない
- (5) クラス変数はクラス単位で準備される変数である
- (6) クラスメソッド内でインスタンス変数を参照してもよい
- (7) クラスメソッドを実行する場合はクラス名を指定して行う
- (8) クラス変数とクラスメソッドの宣言は static 修飾子で行う

難易度☆☆☆

課題1 コンストラクタの機能を用いて、次に示すクラス **Triangle** のフィールドを与えられた値で初期設定できるようにしなさい。下にコードの実行結果を示します。

ソースファイル名：Assignment4_1.java （main()メソッドがあるクラス名と一致させる）

```
class Triangle
{
    private double height;
    private double base;

    

ここにコンストラクタを追加します



    public void setHeight(double h){
        height = h;
    }
    public void setBase(double b){
        base = b;
    }
    public double getArea(){
        return(height*base/2.0);
    }
}

class Assignment4_1
{
    public static void main(String[] args){
        // このように初期設定がしたい
        Triangle tr;
        tr = new Triangle(10.5, 2.5);
        System.out.println("三角形の面積は"+tr.getArea()+"です");
    }
}
```

〔実行結果〕

```
>java Assignment4_1
三角形の面積は 13.125 です
```

難易度★★★

課題2 コンストラクタの機能を用いて、次に示すクラス **ConstructorTest** のフィールドを与えられた値で初期設定できるようにしなさい。下にコードの実行結果を示します。

ソースファイル名：Assignment4_2.java （main()メソッドがあるクラス名と一致させる）

```
class ConstructorTest
{
    private int i;
    private double d;
    private String s;

    

ここにコンストラクタを追加します



    public void setValue(int n){
        i=n;
    }
    public void setValue(double n){
        d=n;
    }
    public void setValue(String n){
        s=n;
    }
    public void dispValue(){
        System.out.println("int="+i+", double="+d+", String="+s);
    }
}

class Assignment4_2
{
    public static void main(String[] args){
        // このように初期設定がしたい！！
        ConstructorTest ct;
        ct=new ConstructorTest(10,12.5," Constructor");
        ct.dispValue();
    }
}
```

〔実行結果〕

```
>java Assignment4_2
int=10, double=12.5, String=Constructor
```

難易度★★★

課題3 クラス変数とクラスメソッドを用いることにより簡単なライブラリを作成できます。次に示すクラス変数とクラスメソッドをもつクラス **MyMath** を宣言しなさい。

クラス変数：

```
private メンバ    double PI=3.141592;
```

クラスメソッド：

```
public メンバ    double transRad2Deg(double r);    // ラジアン r を度へ変換
                                                         // degree = radian * 180.0 / PI
                                                         double transDeg2Rad(double d);    // 度 d をラジアンへ変換
                                                         // radian = degree * PI / 180.0
                                                         double getAreaOfCircle(double r);    // 半径 r の円の面積を求める
                                                         // area = PI * r * r
```

難易度★★★

課題4 クラス **MyMath** の動作をチェックするために、メインメソッドからクラス **MyMath** のクラスメソッドを実行するコードを記述しなさい。

難易度★★★

課題5 次は書籍を管理するクラス **Book** のメンバの一覧である。インスタンス変数/メソッドにより個々の書籍の情報を保持し、クラス変数/メソッドにより書籍の総数を保持するようにしたい。クラス **Book** を宣言しなさい。

クラス変数：

```
private メンバ    int num_of_books;    // 書籍の総数
```

クラスメソッド：

```
public メンバ    void stat_of_books();    // 書籍の総数を出力
```

インスタンス変数：

```
private メンバ
    String name;    // 書籍名
    int price;    // 価格
```

インスタンスメソッド：

```
public メンバ
    void infoBook();    // 書籍の情報を出力
```

コンストラクタ：

```
public
    Book();    // コンストラクタ
               // 書籍名"No title"、価格は-1 で初期化
    Book(Sting, int);    // コンストラクタ
               // 与えられた書籍と価格で初期化
```

難易度★★★

課題6 クラス **Book** の動作をチェックするために、メインメソッドからクラス **Book** のオブジェクトを複数個生成し、書籍の総数を出力するコードを記述しなさい。

難易度★★★

課題7 コンストラクタの機能を用いて、次に示すクラス **Person** のフィールドを与えられた値で初期設定できるようにしなさい。下に実行結果を示します。

ソースファイル名：Assignment4_7.java （main()メソッドがあるクラス名と一致させる）

```
class Person
```

```
{
```

```
    private String name;
```

ここにコンストラクタを
オーバーロードして追加します

```
    public void setName(String s){  
        name=s;  
    }
```

```
    public String getName(){  
        return name;  
    }
```

```
}
```

```
class Assignment4_7
```

```
{
```

```
    public static void main(String[] args){  
        Person p1 = new Person();  
        Person p2 = new Person("太郎");
```

```
        System.out.println("p1="+p1.getName());  
        System.out.println("p2="+p2.getName());
```

```
    }
```

```
}
```

〔実行結果〕

```
>java Assignment4_7
```

```
p1=名無し
```

```
p2=太郎
```