

## K227 (オブジェクト指向プログラミング) 演習

2010年06月28日

知識教育研究センター 金井秀明

1. 0から100までの数字で, ちょうど23で割り切れるもの数の合計を求めるプログラム (Mon1.java) を作成せよ. ただし, 繰り返し文を使うこと.
2. 次のような正方形を表すRectというクラスがあります. (配点5点)

```
class Rect {
    private double width;
    private double height;

    Rect() {}
    Rect(double x, double y) {
        width = x;
        height = y;
    }

    double getArea() {
        return width*height;
    }
}
```

このクラスを継承した, 立方体を表Cubeというクラスを作ってください.

- Cubeクラスは, double型のdepth (奥行き) というフィールドを持つ.
- Cubeクラスのコンストラクタを作成せよ. コンストラクタの中では, 引数x, y, zをそれぞれwidth, height, depthに代入するようにすること.
- getVolume()という体積を計算して返すメソッドを定義してください.

3. Mon3.javaを実行すると、実行結果1のようになる。実行結果2のように実行するように、ClassListクラスを修正せよ。

```
=====Mon3.java=====
class ClassList {
    public static String title;
    private static String teacher;
    public ClassList (String title, String teacher) {
        title = title;
        teacher = teacher;
    }
    public void Show() {
        System.out.println(title+":"+teacher);
    }
}

class Mon3 {
    public static void main (String args[]) {
        ClassList[] classlist = new ClassList[2];
        classlist[0] = new ClassList("K227", "Kanai");
        classlist[1] = new ClassList("K225", "Kanai");

        classlist[0].Show();
        classlist[1].Show();
    }
}
```

=====実行結果1 (Result1)=====

```
null:null
null:null
```

=====実行結果2 (Result2)=====

```
K227:Kanai
K225:Kanai
```

4. 次の機能を持つPlacedRectangleを、以下のRectangleのサブクラスとして宣言してください。ただし、Rectangleの定義は変更しないこと。

- ・位置を示すint型のフィールドx, yを持つ。
- ・3つのコンストラクタを持つ
  - (1)引数なし
  - (2)引数あり：位置x, yを代入する
  - (3)引数あり：位置x, yと大きさwidth, heightを代入する
- ・位置x, yを設定するメソッドsetLocationを持つ。
- ・以下のように出力するtoStringメソッドを持つ。  
x =12, y = 34, width = 123, height = 45のときには、toStringは、[(12, 34) [123, 45]]という文字列を返す。

```
#####  
// Rectangleクラスの宣言  
class Rectangle {  
    private int width;  
    private int height;  
    Rectangle () {  
        setSize(0, 0);  
    }  
    Rectangle(int width, int height) {  
        setSize(width, height);  
    }  
    private void setSize(int width, int height) {  
        this.width = width;  
        this.height = height;  
    }  
    public String toString() {  
        return "[" + width + ", "+ height+"]";  
    }  
}
```

#####

5. 「円」クラス (Circle) のサブクラスとして、扇形クラス(Fan)を宣言せよ。扇形クラスは以下の機能を持つ。

(a)フィールド

- ・中心角を示すdouble型のフィールドthetaを持つ。単位は度とする。

(b)コンストラクタ

- ・引数なし。ただし半径(r)と中心角(theta)の初期値は0.0とする。
- ・引数あり。ただし半径(r)と中心角(theta)を指定する。

(c)メソッド

- ・面積を求めるメソッドgetArea()を定義する。

扇形の面積=円の面積\*中心角/360

```
#####  
class Circle {  
    private double r; //半径  
  
    Circle() { this.r = 0.0; }  
    Circle(double r) { this.r = r; }  
  
    double getArea() {  
        return this.r*this.r*3.14;  
    }  
}
```

```
#####
```

6. 次のLadyというクラスはageというフィールドを持つ単純なクラスです。Mon6のmainメソッドからは、ageフィールドに直接アクセスします。Ladyのageフィールドをprivateに変更して、セッター、ゲッターメソッドを通してアクセスするようにLadyクラス、Mon6のmainメソッドを修正せよ。

```
=====Mon6.java=====
class Lady {
    int age;
}

class Mon6 {
    public static void main (String args[]) {
        Lady lady = new Lady();
        lady.age = 30;
        System.out.println("age=" + lady.age);
    }
}

=====Mon6.java=====
```