

課題 8-1, 8-2 のプログラム例

```

class Base {
    protected int num;
    public Base() {
        num = 0;
    }
    public Base(int n) {
        num = n;
    }
    public void setNum(int n) {
        num = n;
    }
    public int getNum() {
        return num;
    }
    public void showNum() {
        System.out.printf("num: %d\n", num);
    }
}

```

//偶数を保持するクラス

```

class Even extends Base {
    //代入を受け付けた回数と却下された回数
    private int accept, reject;
    //コンストラクタ
    public Even() {
        super(0);
        accept = 0;
        reject = 0;
    }
    public Even(int n) {
        super(n);
        accept = 0;
        reject = 0;
    }
    //偶数のみを代入する num のセッタ
    public void setNum(int n) {
        //偶数かどうかを調べて値を代入する
        if(n % 2 == 0) {
            num = n;
            accept++;
        } else {
            reject++;
        }
    }
    //showNum をオーバーライドする
    public void showNum() {
        super.showNum();
        System.out.println("accept: "
                           + accept
                           + ", reject: "
                           + reject);
    }
}

```

```

class Pd10Base1 {
    public static void main(String[] args) {

```

```

//インスタンスを作成する
Base b1 = new Base();
Base b2 = new Even(10);
//showNum の動作確認
System.out.println("--- b1.show() ---");
b1.showNum();
System.out.println("--- b2.show() ---");
b2.showNum();
//setNum の動作確認
b1.setNum(15);
b2.setNum(15);
b2.setNum(8);
b2.setNum(3);
//setNum の結果を出力する
System.out.println("--- b1.show() ---");
b1.showNum();
System.out.println("--- b2.show() ---");
b2.showNum();

```

```

}
}

class Pd10Base2 {
    public static void main(String[] args) {
        Base[] bases = new Base[2];
        bases[0] = new Base(100);
        bases[1] = new Even(200);
        for(int i=0; i<bases.length; i++) {
            System.out.printf(
                "--- bases[%d].show() ---\n", i);
            bases[i].setNum(i+10);
            bases[i].showNum();
        }
    }
}

```

課題 8-3, 8-4 のプログラム例

//辺の長さを表すクラス

```

class Length {
    protected int height; //高さ
    protected int width; //幅
    //コンストラクタ
    public Length(int h, int w) {
        height = h;
        width = w;
    }
    //フィールドの出力
    public void show() {
        System.out.printf("height: %d, width: %d\n",
                           height, width);
    }
    //面積を出力する
    public void area() {
        System.out.println("スーパークラスは面積
        を求めない");
    }
}

```

```

//立方体を表すクラス
class Cube extends Length {
    private int depth; //奥行
    public Cube(int h, int w, int d) {
        super(h, w);
        depth = d;
    }
    //高さと幅に加えて奥行きを表示する
    public void show() {
        super.show();
        System.out.printf("depth: %d\n", depth);
    }
    public void area() {
        System.out.printf(
            "直方体の表面積: %d\n",
            (height*width + width*depth
            + height*depth) * 2);
    }
}

//四角形を表すクラス
class Rectangle extends Length {
    public Rectangle(int h, int w) {
        super(h, w);
    }
    //四角形の面積を求める
    public void area() {
        System.out.printf("四角形の面積: %d\n",
            height*width);
    }
}

//三角形を表すクラス
class Triangle extends Length {
    public Triangle(int h, int w) {
        super(h, w);
    }
    //三角形の面積を求める
    public void area() {
        System.out.printf("三角形の面積: %d\n",
            height*width/2);
    }
}

class Pd10Length1 {
    public static void main(String[] args) {
        Cube c1 = new Cube(3, 5, 7);
        System.out.println("--- c1.show() ---");
        c1.show();
        c1.area();
    }
}

class Pd10Length2 {
    public static void main(String[] args) {
        Length[] ls;
        ls = new Length[4];
        ls[0] = new Rectangle(15, 50);
        ls[1] = new Triangle(40, 20);
        ls[2] = new Cube(10, 30, 5);
        ls[3] = new Length(8, 6);

        for(int i=0; i<ls.length; i++) {
            System.out.printf(
                "--- bases[%d].show() ---\n", i);
            ls[i].show();
            ls[i].area();
        }
    }
}

```