

確認〇×問題

次の各文は正しいか誤っているか答えなさい。

- (1) 新しいクラスを宣言するとき既存のクラスを利用することはできない
- (2) 新しいクラスが既存のクラスのメンバを受け継ぐことを継承という
- (3) 継承における既存のクラスをサブクラスという
- (4) サブクラスからスーパークラスのメンバをアクセスすることは一般にはできない
- (5) 継承はキーワード extends により指定する
- (6) コンストラクタはクラスのメンバではないため、継承されない
- (7) スーパークラスのコンストラクタは super();により呼び出しが可能である
- (8) スーパークラスの protected メンバはサブクラスからアクセスが可能である

課題1 次は建物全般を表す建物クラスです。この建物クラス(スーパークラス)を継承して、具体的な建物を表すクラス(サブクラス)を宣言しなさい。例えば、マンションを表すクラスは、メンバに世帯数を加えて以下のように宣言すればよいでしょう。追加するメンバは各自にお任せします。

```
// 建物クラス (スーパークラス)
class Building
{
    public int floors; // 何階建
    public int height; // 高さ

    public void showBuilding(){
        System.out.println(floors+"階建");
        System.out.println("高さ："+height+"m");
    }
}
// マンションクラス (サブクラス)
class Mansion extends Building
{
    public int families; // 世帯数

    public void showMansion(){
        showBuilding();
        System.out.println("世帯数："+families);
    }
}
// この他、具体的な建物を表すクラスをここに宣言してください。
// たとえば、デパート (Department) や大学 (University) などがあります。

class Assignment6_1
{
    public static void main(String[] args){
        // 作成したサブクラスの動作を確認 (マンションクラスの場合)
        Mansion mymansion=new Mansion();
        mymansion.floors=10;
        mymansion.height=20;
        mymansion.families=50;
        mymansion.showMansion();
    }
}
```

課題2 次は物体全般を表す物体クラスです。この物体クラスを継承し、具体的な物体を表すサブクラスを宣言しなさい。例えば、机、車、テレビを表すクラスなどがあります。サブクラスではフィールドを初期設定するためのコンストラクタも適切に宣言しなさい。さらに、メインメソッドからサブクラスのオブジェクトを作成して動作を確認しなさい。

```
// 物体クラス
class Objects
{
    public String name;      // オブジェクトの名前
    public double[] size;    // サイズ縦、横、奥行(mm)
    public double weight;    // 重さ(g)

    public Objects(){
        // フィールドの初期設定
        name="No name";
        size=new double[3];
        for(int i=0;i<size.length;i++)
            size[i]=0.0;
        weight=0.0;
    }
    public Objects(String n, double[] s, double w){
        // 与えられたデータをフィールドに設定
        name=n;
        size=s;
        weight=w;
    }
    public void show(){
        // 各フィールドを表示
        System.out.println("Name:"+name);
        System.out.println("size:H"+size[0]+" W"+size[1]+" D"+size[2]+"(mm)");
        System.out.println("Weight:"+weight+"(g)");
    }
}
```

課題3 MP3 プレーヤや携帯など電子ガジェットを管理するクラスを作りたい。そこで次のようなクラス階層を考えた。このクラスを宣言しなさい。

スーパークラス Gadget ※各メンバのアクセス制限は各自で適切に設定しなさい

メンバ : 製品名 (String)、価格 (int)

メソッド : Gadget(String, int); コンストラクタ (製品名と価格の設定)

void show(); データ表示 (製品名と価格の表示)

サブクラス Mp3Player ※各メンバのアクセス制限は各自で適切に設定しなさい

追加メンバ : 容量 MB (int)

追加メソッド : Mp3Player(String, int, int);

コンストラクタ (super() で製品名と価格、と容量の設定)

void print();

データ表示 (継承メソッドを用いて製品名と価格、と容量の表示)

サブクラス CellPhone ※各メンバのアクセス制限は各自で適切に設定しなさい

追加メンバ : 番号 (String)

追加メソッド : CellPhone(String, int, String);

コンストラクタ (super() を用いて製品名と価格、と番号の設定)

void print();

データ表示 (継承メソッドを用いて製品名と価格、と番号の表示)

課題4 メインメソッド内でクラス Mp3Player と CellPhone のオブジェクトを生成しなさい。また各オブジェクトのデータを出力しなさい。

課題5 次はひと月の家計簿を処理するクラス HouseholdAcc のメンバの一覧である。インスタンス変数とメソッドで各月の収入支出バランスを、クラス変数とメソッドで年間の収入支出のバランスを管理したい。クラス HouseholdAcc を宣言しなさい。(クラスの配列の作成：5回目の講義の範囲です)

クラス変数：

```
// 年間収入支出バランス (各月バランスの累積)
Private メンバ int fiscal_balance=0;
```

クラスメソッド：

```
// 年間収入支出バランスを出力 (黒字?赤字?)
Public メンバ void showFiscalBalance();
```

インスタンス変数：

```
Private メンバ
    int income; // 収入
    int spent; // 支出
```

インスタンスメソッド：

```
Public メンバ
    HouseholdAcc(); // コンストラクタ
                    // 収入と支出を 0 で初期化する
    void setData(int, int); // 今月の収入と支出に与えられたデータをセットして、
                           // 年間収入支出バランスに累積する
    void showBalance(); // 今月の収入と支出のバランスを出力 (黒字?赤字?)
```

課題6 クラス HouseholdAcc のオブジェクト1年間分(12か月分)を配列を用いて生成しなさい。各オブジェクトに各月のデータを入力し、各月の収入支出バランスと年間の収入支出バランスを出力しなさい。(クラスの配列の作成：5回目の講義の範囲です)

課題7 次は仕事するヒトを表わすクラス Worker である。この Worker クラスを継承し、具体的な仕事を行うヒトを表わすサブクラスを宣言しなさい。例えば、医者、ドライバ、先生などがあります。医者であれば、患者数などの変数と、表示するメソッド void showall(); を以下の例のように宣言すればよいでしょう。メインメソッドからサブクラスのオブジェクトを作成して動作を確認しなさい。

```
// 仕事をする人を表わすクラス（スーパークラス）
```

```
class Worker
{
    public String name; // 名前
    public int age;     // 年齢

    public void show(){
        System.out.println("名前:"+name);
        System.out.println("年齢:"+age);
    }
}
```

```
// 医者を表わすクラス（サブクラス）
```

```
class Doctor extends Worker
{
    public int patients; // 患者数

    public void showall(){
        System.out.println("■医者■");
        show();
        System.out.println("患者数:"+patients);
    }
}
```

```
// 医者の場合のコード例
```

```
class Assignment6_7
{
    public static void main(String[] args){

        Doctor doc=new Doctor();

        doc.name="福工大 太郎";
        doc.age=25;
        doc.patients=55;

        doc.showall();
    }
}
```