

作成者: mitsuya.85u@ray.ocn.ne.jp

0.2. アニマルクラス（継承と多態）

継承と多態 問題

練習問題 1

(1)以下の Animal クラスを継承したクラス（派生クラス） Cat クラスを作成してください。

- Cat クラスはメンバー・メソッド punch() を持ちます。
punch() を実行すると“猫パンチBomBom”という文字列を表示します。

```
class Animal{
    protected String name;
    protected int age;

    public void setName( String nm )    {
        name = nm;
    }
    public void setAge( int ag )    {
        age = ag;
    }
    public void showProfile(){
        System.out.println( "名前は、" + name + "、" + age + "歳です。" );
    }
}
```

(クラス図)

Animal

name:String

age:int

+ setName(nm:String):void

+ setAge(ag:int):void

+ showProfile():void

↑

Cat

```
-----  
-----  
+ punch():void  
-----  
-:private  
+:public  
#:protected
```

(2)前問の Catクラスを次の手順で実体化するAnimalMainクラス（神様クラス）を作成してください。

- Cat クラスのインスタンスを作成する。
- メンバー変数、name、age に名前、年齢を設定する。
- showProfile() を実行する。
- punch() を実行する。

(クラス図)

```
-----  
AnimalMain  
-----  
-----  
+ main(args:String[]):staic void  
-----  
-:private  
+:public  
#:protected
```

練習問題 2

「概念と実体」の問題で作成したDogクラスを Animal クラスを継承するよう変更してください。さらに、以下の仕様を追加してください。

- Dog クラスはメンバー・メソッドwalk()を持つ。
walk() を実行すると“犬の散歩TokoToko”という文字列を表示する。
- showProfile()メソッドは犬種を表示するようオーバーライドする。

(クラス図)

```
-----  
Animal  
-----
```

```
# name:String
# age:int
-----
+ setName(nm:String):void
+ setAge(ag:int):void
+ showProfile():void
-----
```

↑

```
-----
Dog
-----
- kind:String
-----
+ Dog();
+ Dog(k:String):
+ walk():void
+ showPrifile():void
-----
-:private
+:public
#:protected
```

(2)次のようにAnimalMainクラス（神様クラス）を変更してください。

- Cat クラスと Dog クラスのインスタンスを作成する。
- それぞれのメンバー変数、name、age に名前、年齢を設定する。
- それぞれの showProfile を実行する。
- Cat のインスタンスの punch() を実行する。
- Dog のインスタンスの walk() を実行する。

【実行結果】

```
>java AnimalMain
名前は、たま、3歳です。
猫パンチBomBom
雑種で名前は、ポチ、5歳です。
犬の散歩TokoToko
```

練習問題 3

前問の Animal クラスに以下のメンバー・メソッドtouch()を追加してください。

```
public void touch() {  
    System.out.println( "ナデナデ" );  
}
```

(1)Dog クラス、Cat クラスに touch() メソッドをオーバーライドするメソッドを追加してください。それぞれでtouch()メソッドを実行するとDogクラスのインスタンスでは“ワンワン”、Catクラスのインスタンスでは“ニャー”という文字列を表示します。

(クラス図)

Animal

name:String

age:int

+ setName(nm:String):void

+ setAge(ag:int):void

+ showProfile():void

+ touch():void

↑

↑

Dog

Cat

+ Dog():

+ Dog(kd:String):

+ walk():void

+ touch():void

+ showProfile():void

+ punch():void

+ touch():void

-:private

+:public

#:protected

(2)次のようにAnimalMainクラス（神様クラス）を変更してください。

- Cat クラスと Dog クラスのインスタンスを作成する。
- それぞれのメンバー変数、name、age に名前、年齢を設定する。
- それぞれの showProfile() を実行する。
- それぞれの touch() を実行する。

【実行結果】

```
>java AnimalMain
名前は、たま、3歳です。
猫パンチBomBom
ニャー
雑種で名前は、ポチ、5歳です。
犬の散歩TokoToko
ワンワン
```

練習問題 4

前問のAnimalMainクラスを次のように変更した（多態に変更した）AnimalMainPolyクラスを作成してください。

- Animal クラスの配列を作成する。要素数は 4。
- 配列の偶数番目（0 と 2）には Cat クラスのインスタンス、奇数番目（1 と 3）には Dog クラスのインスタンスを設定する。
- for文を使い配列の総ての要素の touch()メソッドを実行する。

```
>java AnimalMainPoly
ニャー
ワンワン
ニャー
ワンワン
```

練習問題 5

練習問題 4 では、Animalクラスのtouch()メソッドが全面的にオーバーライドされており、ナデナデが出力される機会がありません。Dogクラス、Catクラスのtouch()メソッドを変更することにより、以下のような出力結果となるよう実装してください。

```
>java AnimalMainPoly
ナデナデ
ニャー
ナデナデ
```

ワンワン
ナデナデ
ニャー
ナデナデ
ワンワン