# 第7次平时作业

package work7.package1;  
  
/\* 接口类Animal\*/  
public interface Animal {  
  
 /\* 为了简化操作，只设置了一个方法\*/  
 public abstract void showType();  
  
  
}

package work7.package1;  
  
/\* 基类Pet\*/  
public class Pet {  
  
 /\* 类属性，全部设置为私有，子类只能通过公有方法访问\*/  
 private String name;  
 private int age;  
 private String sex;  
  
 /\* 全参构造\*/  
 public Pet(String name, int age, String sex) {  
 this.name = name;  
 this.age = age;  
 this.sex = sex;  
 }  
  
 /\* 公有方法抓取name属性\*/  
 public String getName() {  
 return this.name;  
 }  
  
 /\* 公有方法获得age属性\*/  
 public int getAge() {  
 return this.age;  
 }  
  
 /\* 公有方法抓取sex属性\*/  
 public String getSex() {  
 return this.sex;  
 }  
  
 /\* 打印信息\*/  
 public void printInfo(){  
 System.*out*.println("this is a pet, and its name:"+this.name+" age: "+this.age+" sex: "+this.sex);  
 }  
}

package work7.package2;  
  
import work7.package1.Animal;  
  
/\* 通过import package1，得到Animal接口的访问权限  
\* Dog类实现Animal接口 \*/  
public class Dog implements Animal {  
  
 /\* 类属性\*/  
 String name;  
 int age;  
  
 public Dog(){  
  
 }  
 /\* 构造\*/  
 public Dog(String name, int age) {  
 this.name = name;  
 this.age = age;  
 }  
  
 /\* 实现接口方法\*/  
 @Override  
 public void showType() {  
 System.*out*.println("this is a dog, and its name is "+this.name+" age: "+this.age);  
 }  
}

package work7.package2;  
  
import work7.package1.Pet;  
  
/\* import package1 ,访问Pet父类  
\* 继承Pet类\*/  
public class Cat extends Pet {  
  
 /\* 增加属性type\*/  
 private String type;  
  
 /\* 直接调用基类构造方法\*/  
 public Cat(String name, int age, String sex) {  
 super(name, age, sex);  
 }  
  
 /\* 设置特有属性type\*/  
 public void setType() {  
 this.type = "cat";  
 }  
  
 /\* 重写基类方法，打印信息\*/  
 @Override  
 public void printInfo() {  
 System.*out*.print("this is a "+this.type);  
 System.*out*.println(". and its name: "+this.getName()+" age: "+this.getAge()+" sex: "+this.getSex());  
 }  
}

package work7.package3;  
  
import work7.package1.Pet;  
import work7.package2.Cat;  
import work7.package2.Dog;  
  
/\* 测试类\*/  
public class Test {  
  
 public static void main(String[] args) {  
 /\* 实例化不同类的对象，检验方法属性是否有效\*/  
 Dog dog = new Dog("qiuqiu",2);  
 dog.showType();  
  
 Pet pet = new Pet("blank",3,"mal");  
 pet.printInfo();  
  
 Cat cat = new Cat("miaomiao",1,"femal");  
 cat.setType();  
 cat.printInfo();  
 }  
}