```java
package com.example;

public abstract class Animal {

    protected int legs;

    public Animal(int legs) {
        this.legs = legs;
    }

    public void walk() {
        System.out.println("This animal walks on " + legs + " legs.");
    }

    public abstract void eat();

}
```

```java
package com.example;

public class Spider extends Animal {

    public Spider() {
        super(8);
    }

    @Override
    public void eat() {
        System.out.println("The spider eats a fly.");
    }

}
```

```java
package com.example;

public class Cat extends Animal implements Pet {

    private String name;

    public Cat() {
        this("Fluffy");
    }

    public Cat(String name) {
        super(4);
        this.name = name;
    }

    @Override
    public void eat() {
        System.out.println("Cats like to eat spiders and fish.");
    }

    @Override
    public String getName() {
        return name;
    }

    @Override
    public void setName(String name) {
        this.name = name;
    }

    @Override
    public void play() {
        System.out.println(name + " likes to play with string.");
    }
```

```
}

package com.example;

public class Fish extends Animal implements Pet {

    private String name;

    public Fish() {
        super(0);
    }

    @Override
    public void eat() {
        System.out.println("Fish eat pond scum.");
    }

    @Override
    public String getName() {
        return name;
    }

    @Override
    public void setName(String name) {
        this.name = name;
    }

    @Override
    public void play() {
        System.out.println("Just keep swimming.");
    }

    @Override
    public void walk() {
        super.walk();
        System.out.println("Fish, of course, can't walk; they swim.");
    }
}

package com.example;

interface Pet {

    public String getName();

    public void setName(String name);

    public void play();

}

package com.example;

public class PetMain {

    public static void main(String[] args) {
        Animal a;
        //test a spider with a spider reference
        Spider s = new Spider();
        s.eat();
        s.walk();
        //test a spider with an animal reference
        a = new Spider();
        a.eat();
        a.walk();
```

```
        Pet p;

        Cat c = new Cat("Tom");
        c.eat();
        c.walk();
        c.play();
        a = new Cat();
        a.eat();
        a.walk();
        p = new Cat();
        p.setName("Mr. Whiskers");
        p.play();

        Fish f = new Fish();
        f.setName("Guppy");
        f.eat();
        f.walk();
        f.play();
        a = new Fish();
        a.eat();
        a.walk();


        playWithAnimal(s);
        playWithAnimal(c);
        playWithAnimal(f);
    }

    public static void playWithAnimal(Animal a) {
        if (a instanceof Pet) {
            Pet p = (Pet) a;
            p.play();
        } else {
            System.out.println("Danger! Wild Animal");
        }
    }
}
```