abstract class Animal

{

protected int legs;

protected Animal(){legs=0;}

protected Animal(int legs)

{

this.legs=legs;

}

abstract void eat();

void walk()

{

System.*out*.println("Most animals walk fastly!");

}

}

class Spider extends Animal

{

Spider()

{

super(8);

}

void eat()

{

System.*out*.println("Spider eats insects!");

}

}

public class Fish extends Animal implements pet

{

String name;

Fish()

{

super(0);

}

void walk()

{

System.*out*.println("Fish can't walk because it has no legs!");

}

public String getName()

{

return name;

}

public void setName(String name )

{

this.name=name;

}

public void play()

{

System.*out*.println("Fish play with ripples!");

}

public void eat()

{

System.*out*.println("Fish eat insects in water!");

}

}

public interface pet

{

String getName();

void setName(String name);

void play();

}

public class Cat extends Animal implements pet

{

String name;

Cat(String name)

{

super(4);

this.name=name;

}

Cat()

{

this(" ");

}

public String getName()

{

return name;

}

public void setName(String name)

{

this.name=name;

}

public void eat()

{

System.*out*.println("Cat eats meat!");

}

public void play()

{

System.*out*.println("Cat plays with ball");

}

public void walk()

{

System.*out*.println("Cat walks and also Runs!");

}

}

public class TestAnimal

{

public static void main(String args[])

{

Fish f=new Fish();

Cat c=new Cat("Fluffy");

Spider s=new Spider();

Animal a=new Fish();

Animal e=new Cat();

Animal sp=new Spider();

f.play();

c.play();

e.eat();

e.walk();

a.walk();

s.eat();

s.walk();

}

}