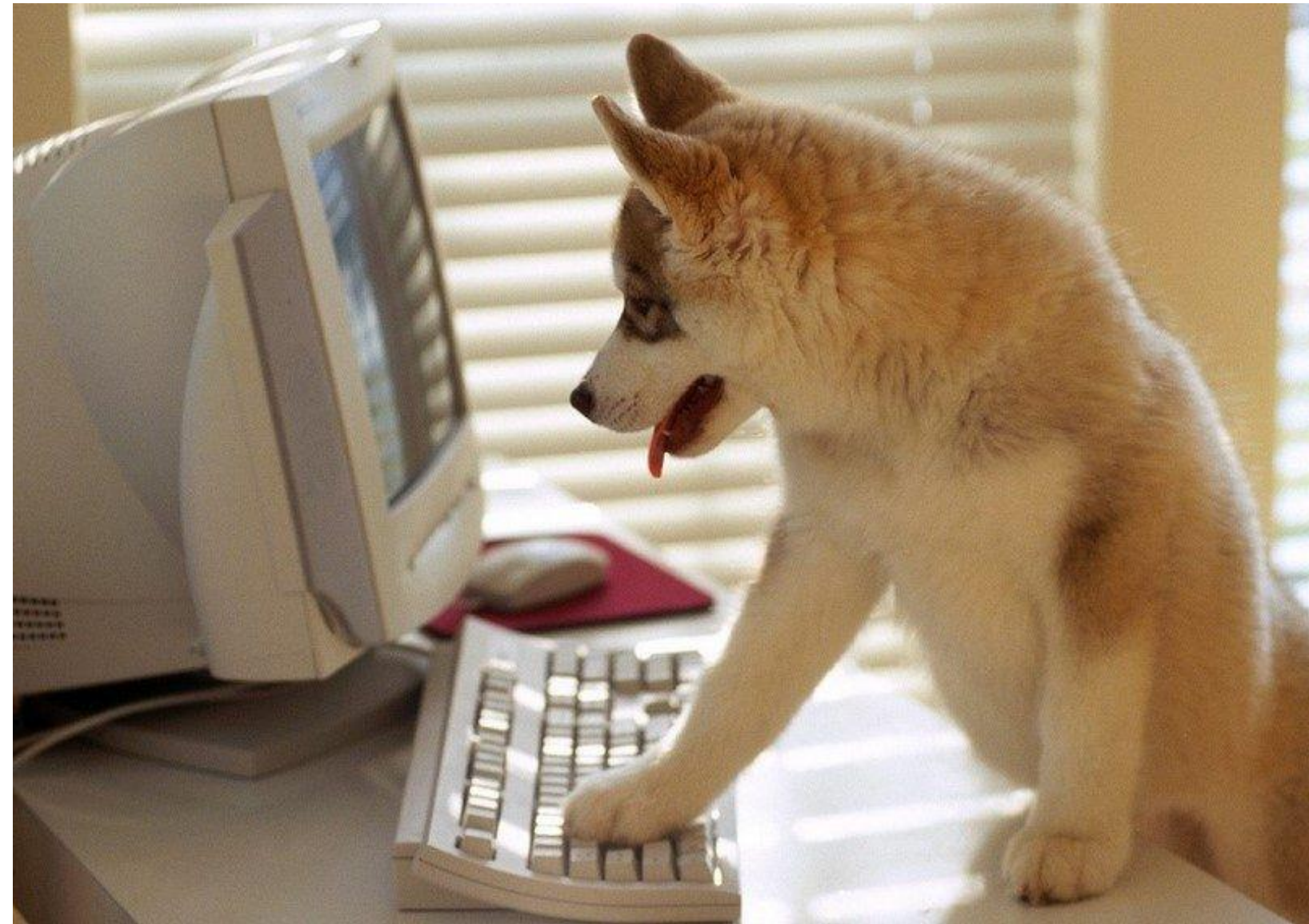


# Java Programming



# Organizational Stuff

Find a partner to work on the mini project together.  
You can also suggest a project topic till tomorrow.

# Organizational Stuff

18.03.: Structures

19.03.: Methods

20.03.: Recursion

21.03.: Arrays

22.03.: Strings

-----

25.03.: OOP1

26.03.: OOP2

27.03.: Generics

28.03.: Exceptions & **Testing**

29.03.: GUI

```
public abstract class Dog{  
    protected String name;  
    public abstract void bark();  
}
```

```
public abstract class Dog{  
    protected String name;  
    public abstract void bark();  
}
```

abstract : **Declaration without implementation**

```
public abstract class Dog{  
    protected String name;  
    public abstract void bark();  
}
```

An abstract class is a class with at least one abstract method.

```
public abstract class Dog{  
    protected String name;  
    public abstract void bark();  
}
```

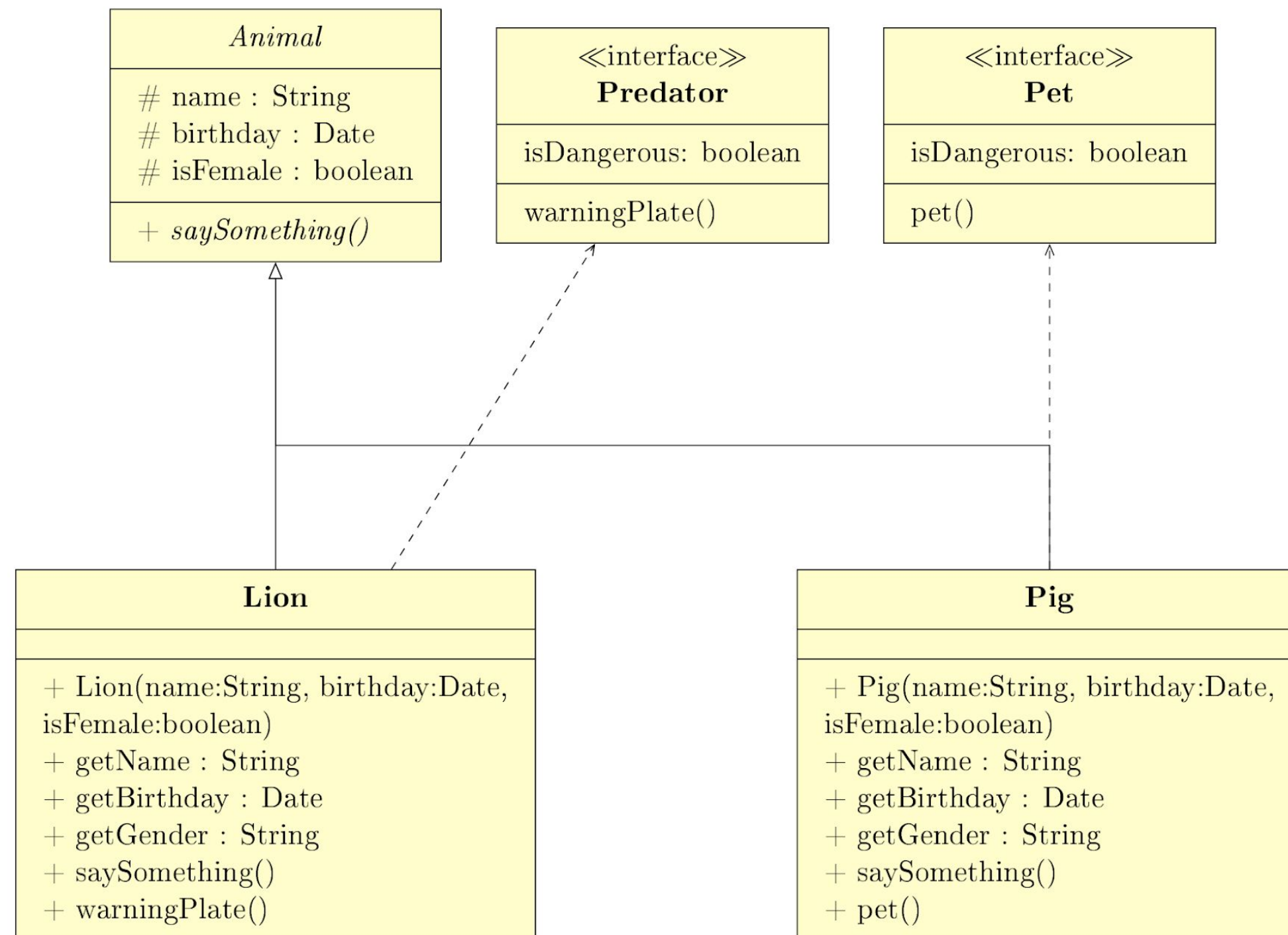
```
public class Bulldog extends Dog{  
    public void bark() {  
        System.out.println("Woof!");  
    }  
}
```

```
public interface iWalkable{  
    void walk();  
}
```

An Interface is like a service, not like a relation.  
It contains just constants and abstract methods (no keyword **abstract** needed)



```
public interface iWalkable{  
    void walk();  
}  
  
public class Bulldog extends Dog implements iWalkable{  
    public void bark(){  
        System.out.println("Woof!");  
    }  
    public void walk(){  
        System.out.println("I'm walking like a Bulldog");  
    }  
}
```



**Today's Assignment:**

<https://classroom.github.com/a/RmFqfHPR>

