

Deadly Diamond of Death



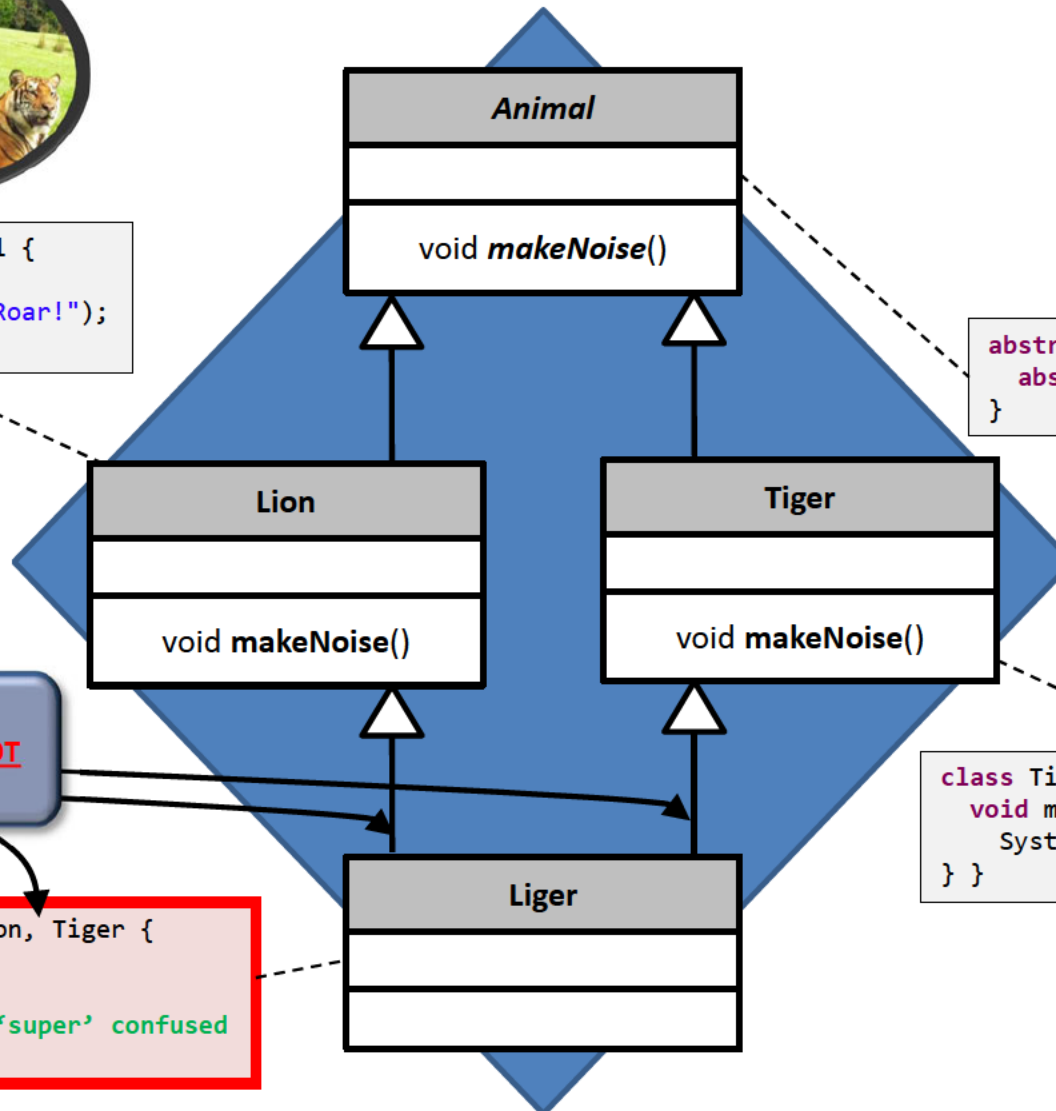
```
class Lion extends Animal {  
    void makeNoise() {  
        System.out.println("Roar!");  
    }  
}
```

```
abstract class Animal {  
    abstract void makeNoise();  
}
```

multiple
inheritance is **NOT**
allowed in Java

```
class Liger extends Lion, Tiger {  
    void makeNoise() {  
        super.makeNoise();  
        // Liger would be 'super' confused  
    }  
}
```

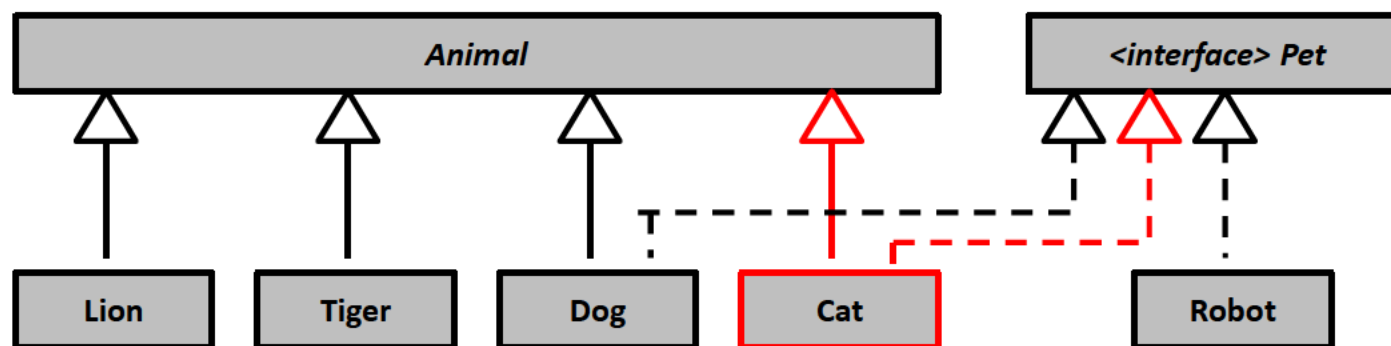
```
class Tiger extends Animal {  
    void makeNoise() {  
        System.out.println("Grrh!");  
    }  
}
```



slide idea originally by N. Wu

Class Association to Multiple Concepts

- in multiple inheritance there is **no simple way** to resolve calls to **super** (one could use separate namespaces etc)
- in response, Java **forbids all** multiple inheritance: a class can only inherit from **one** parent
- 'Liger' may still be implemented via **two has-a** relationships
- yet, we still would like a language construct that associates a single class with **several** parent concepts:



slide idea originally by N. Wu