

ИНТЕРФЕЙСИ И НАСЛЕДЯВАНЕ

Ненко Табаков

Пламен Танов

Технологическо училище “Електронни системи”

Технически университет – София

23 септември 2008



ИНТЕРФЕЙСИ И НАСЛЕДЯВАНЕ

Забележка: Тази лекция е адаптация на лекция от курса:

• 6.092 Java Preparation for 6.170, Януари 2006

- Lucy Mendel
- Corey McCaffrey
- Rob Toscano
- Justin Mazolla Paluska
- Scott Osler
- Ray He

Интернет адрес:

<http://ocw.mit.edu/OcwWeb/Electrical-Engineering-and-Computer-Science/6-092January--IAP--2006/CourseHome/index.htm>

Лиценз: Creative Commons – BY – NC – SA

СЪДЪРЖАНИЕ

- Интерфейси и наследяване
 - Абстрактни класове
 - Интерфейси
 - Наследяване

АБСТРАКТНИ КЛАСОВЕ

- Използва се, когато част от кода на два класа съвпада
- Дефинират се с ключовата дума *abstract*
- Не могат да се създават обекти от абстрактен клас

```
abstract class Person {  
    private String name = "";  
  
    public String getName() {  
        return name;  
    }  
    public void setName(String n) {  
        name = n;  
    }  
    abstract public String sayGreeting();  
}
```

АБСТРАКТНИ КЛАСОВЕ

- Използва се, когато част от кода на два класа съвпада
- Наследяват се с ключовата дума *extends*

```
// файл EnglishPerson.java
class EnglishPerson extends Person {
    public String sayGreeting() {
        return "Hello";
    }
}

// файл BulgarianPerson.java
class BulgarianPerson extends Person {
    public String sayGreeting() {
        return "Здравей";
    }
}
```

ИНТЕРФЕЙСИ

- Интерфейсът дава списък на достъпните методи
- Дефинират се с ключовата дума *interface*
- Реализират се с ключовата дума *implements*

```
public interface Dragable { // файл Dragable.java
    public void drag();
}
```

```
public class Icon implements Dragable { // файл Icon.java
    public void drag() { ... }
}
```

```
public class Chair implements Dragable { // файл Chair.java
    public void drag() { ... }
}
```

РЕАЛИЗАЦИЯ НА МНОЖЕСТВО ИНТЕРФЕЙСИ

- Всеки един клас може да реализира произволен брой (множество) интерфейси

```
interface Drawable {  
    public void draw();  
}  
  
interface Clickable {  
    public void click();  
}  
  
interface Draggable {  
    public void drag();  
}
```

```
class Icon implements Drawable,  
Clickable, Draggable {  
  
    public void draw() {  
        SOP("drawing...");  
    }  
    public void click() {  
        SOP("clicking...");  
    }  
    public void drag() {  
        SOP("dragging...");  
    }  
}
```

ЕДИНИЧНО НАСЛЕДЯВАНЕ

- Всеки един клас може да наследява само един клас
- Ако той не е указан непосредствено, то родител е *Object*
- Възможно е предефиниране на методи

```
class Parent { // по подразбиране родител е Object
    public String doSomething() {
        return "Hello (Parent)";
    }
}
class Child extends Parent {
    public String doSomething() {
        return "Hello (Child)";
    }
}
```