

# Rozhraní

Z FAV wiki

Objektově orientované jazyky poskytují pro realizaci tříd, kromě dědičnosti, další prostředek - rozhraní. Rozhraní definuje soubor metod bez jejich implementace (podobně jako kompletně virtuální třída). Analogie je v rozepsané kuchařce, kdy víme co to bude za jídlo a z čeho budeme vařit, ale postup neznáme. Třída, která implementuje toto rozhraní (tj. implementuje - definuje - všechny jeho metody), je po té v polymorfovateľná do ukazatele definovaného tímto rozhraním, což znamená, že klient ví, že tato třída má implementované všechny metody, které potřebuje (a které jsou definované v rozhraní).

// trochu divoký vysvětlení

Deklarace rozhraní je podobná deklaraci třídy

```
interface jmeno {  
    // hlavičky metod, všechny samozřejmě virtuální a abstraktní  
}
```

V Javě se implementace rozhraní zapíše pomocí klíčového slova implements

```
Třída implements Rozhraní {  
    ...  
}
```

V jiných jazycích (C++, C#, ...) používáme dvojtečku, přičemž první třída za dvojtečkou je obvykle dědičí, ostatní jsou implementace rozhraní (i když v C++ je to jedno, tam můžeme implementovat i dědit od více tříd). Implementovat můžeme i několik rozhraní (například implements Cloneable, Runnable)

Může se stát, že 2 rozhraní mají stejné metody k implementaci. Implementovat je tedy možné

- **Implicitně** - Implementujeme metodu první nalezené metody s tímto jménem ve všech rozhraních, nebo selže, záleží na jazyku
- **Explicitně** - Implementujeme metodu tak, že řekneme kterou metodu kterého rozhraní implementujeme např:

```
public this.type Cloneable.Clone() {  
    // tělo  
}
```

Nejsilnější zbraní rozhraní je, že odděluje kód od definic (vhodné např. pro pluginové systémy) a je sdružuje třídy do jedné skupiny funkčnosti.

Citováno z „<http://www.512.cz/index.php?title=Rozhran%C3%AD>“

Kategorie: Fav-kiv-bzinf

- Stránka byla naposledy editována 20. 2. 2014 v 06:37.