

Енумерације

Понекад променљива служи за чување вредности из ограниченог скупа. Нпр. можда продајете одећу у 4 величине: small, medium, large и extra large. Наравно, можете кодирати ове величине целим бројевима 1, 2, 3, 4 или карактерима S, M, L, X. Али такав приступ је склон грешкама. Веома лако се може десити да променљива добије погрешну вредност (попут 0 или m).

Почев од Java SE 5.0 могуће је дефинисати сопствени тип енумерације. Такав тип има коначан број именованих вредности. Нпр.

```
enum Velicina {SMALL, MEDIUM, LARGE, EXTRA_LARGE};
```

Тип енумерације заправо се понаша као класа. Ова класа има тачно 4 инстанце - није могуће конструисати нове објекте.

Према томе, никада нема потребе користити метод `equals()` за вредности типа енумерације. Оне се пореде коришћењем `==`.

Могуће је декларисати променљиву овог типа:

```
Velicina v = Velicina.MEDIUM;
```

Променљива типа `Velicina` може чувати само једу од вредности излистаних у декларацији овог типа или специјалну вредност `null`.

Можете, ако желите, додати конструкторе, методе и поља типу енумерације. Наравно, конструктори се позивају само приликом конструисања константи енумерације. Следи пример:

```
enum Velicina {  
    SMALL("S"), MEDIUM("M"), LARGE("L"), EXTRA_LARGE("XL");  
  
    private String skracenica;  
  
    private Velicina(String skracenica) {  
        this.skracenica = skracenica;  
    }  
  
    public String getSkracenica() {  
        return skracenica;  
    }  
}
```

Сви типови енумерације су поткласе класе `Enum`. Они наслеђују извештајан број метода од те класе. Најкориснији међу њима је метод `toString()`, који враћа име константе енумерације. Нпр. `Velicina.SMALL.toString()` враћа стринг „SMALL“.

Супротан метод је статички метод `valueOf()`. Нпр.

```
Velicina v = (Velicina) Enum.valueOf(Velicina.class, "SMALL");
```

поставља `v` на `Velicina.SMALL`.

Сваки тип енумерације поседује статички метод `values()` који враћа низ вредности енумерације. Нпр. позив

```
Velicina[] vrednosti = Velicina.values();
```

враћа низ са елементима

`Velicina.SMALL`, `Velicina.MEDIUM`, `Velicina.LARGE` и `Velicina.EXTRA_LARGE`.

Метод `ordinal()` враћа позицију константе енумерације у `enum` декларацији, рачунајући од 0. Нпр. `Velicina.MEDIUM.ordinal()` враћа 1.

ПРИМЕР 1 (EnumTest): Илуструје рад са типовима енумерације.

```
java.lang.Enum<E>
static Enum valueOf(Class enumClass, String name)
враћа константу енумерације дате класе са задатим именом.

String toString()
враћа име текуће константе енумерације.

int ordinal()
враћа позицију текуће константе енумерације у enum декларацији (броји од 0).

int compareTo(E other)
враћа негативан цео број ако је текућа константа енумерације испред other,
0 ако је this == other, а позитиван цео број иначе.
Поредак константи дат је enum декларацијом.
```

Енумерације се могу користити приликом вишеструког гранања помоћу `switch`:

```
Velicina v = . . . ;
switch (v) {

    case SMALL: // nema potrebe koristiti Velicina.SMALL
        . . .
        break;
    . . .
}
```

У `case` клаузама не наводи се тип енумерације, већ само име константе.