

Annotációk

A nagy sikerre való tekintettel az enum típus bemutatása után következzenek az annotációk, amelyek a generics mellett Java 5 egyik legfontosabb újdonságai. A developer.com körüljárta a témát egészen részletesen [M. M. Islam Christy](#) cikkében.

Az annotációk lényege, hogy a kódot könnyebben karbantarthatóvá, illetve hibatrbbé tehetjük. Annotációt tudunk tenni osztályokra, konstruktorra, metódusokra illetve akár változókra, mezkre. Annotációkat használ a JPA (Java Persistence API) is, illetve sok egyéb programkönyvtár: így működik a Swing Application Framework databinding része. Az annotáció tehát jelez valamit a fordító vagy az egyéb keretrendszerek számára, felfoghatjuk a gépnek szóló kommentként is.

Az annotációk többféle célt szolgálnak. Leggyakrabban a **marker** jelleg annotációkat használjuk, amelyek jeleznek valamit, ezek közül is a Java által lefoglalt annotációkat:

- Override, amely lényege, hogy a megjelölt metódus mindenképpen felül kell írja az osztály azonos metódusát, különben fordításkor hibát kapunk
- Deprecated, amely jelzi, hogy a mögötte lévő metódus elavult, idvel kikerül a használt osztályból
- Suppresswarnings, amely hatására az elz kett fordításkori hibaüzenet elnyelődik

Saját annotációkat is tudunk írni, amelyeknek különféle feladatokat adhatunk, a legegyszerbb annotáció az alábbi:

SimpleAnnotation.java

```
public @interface SimpleAnnotation
{
}
```

De ennek szinte semmi értelme nincs. Az annotációk egyfajta osztályként viselkednek, tehát lehet nekünk metódusuk, amely valamit csinálhat:

SimpleAnnotation.java

```
public @interface SimpleAnnotation
{
    String givenValue();
}
```

Ezt fel tudjuk használni egy metódus annotálására, amely során semmi sem fog történni, de képesek leszünk elérni az átadott értéket, ha szükségünk van rá:

Java forrás

```
@SimpleAnnotation (givenValue="Hello World!")
public void doThis()
{
    System.out.println("Do this!");
}
```

Ha több paramétert adunk meg, akkor azokat vesszvel tudjuk elválasztani az annotáció neve után megadva. Ha egy saját annotációt nem az összes típushoz használnánk, akkor korlátozni tudjuk annak használhatósági körét a @Target annotációval, amely az annotáció annotációja:

- @Target(ElementType.TYPE) - bármilyen eleme az osztálynak
- @Target(ElementType.FIELD) - mez vagy változó
- @Target(ElementType.METHOD) - metódus
- @Target(ElementType.PARAMETER) - metódus paramétere
- @Target(ElementType.CONSTRUCTOR) - konstruktor
- @Target(ElementType.LOCAL_VARIABLE) - helyi változó
- @Target(ElementType.ANNOTATION_TYPE) - annotáció annotációja