ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

Java alapú webfejlesztés

Java - C# eltérések

ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

Java build eszközök

Forrásfájlok => JVM által futtatható program (bájtkód, .jar)

- Ant
  - build.xml: a build folyamata, elvégzendő feladatok, függőségek
- Maven
  - pom.xml: Project Object Model: a projekt tulajdonságainak és függőségeinek leírása
- Gradle
  - build.gradle: már nem XML, hanem domain specifikus nyelv
  - többféle programozási nyelvet is támogat

ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

A Java projekt felépítése

- Solution nincs
- Projekt könyvtár tartalma (Maven project)
  - src: forrásfájlok
    - main: a projekt forrásfájljai
    - test: teszt fájlok
  - target: a build eredménye
  - pom.xml

ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

Csomagok

- package: szerepét tekintve a C# namespace megfelelője
  - a class előtt adjuk meg, a fájl elején, például:
    `package oe.lesson1;`
- Package vagy osztály importálása (≈using)

```
import java.io.*;
import java.nio.file.Files;
```

- A csomag hierarchia megfelel a mappa hierarchiának
  - pl. az oe.lesson1.Rectangle osztály az oe/lesson1/Rectangle.java fájlban található.
  - ugyanez a hierarchia a lefordított (.class) fájloknál is megtalálható

ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

Elnevezési konvenciók

- Többnyire a C#-hoz hasonló

Fontos különbségek:

- a metódusok kisbetűvel kezdődnek (camelCase)
- az interfészeket nem szoktuk `I` betűvel kezdeni

ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

Tulajdonságok

- C#-szerű tulajdonság szintaxis nincs
- Helyette getter és setter metódusok
  - `<típus> valami` változóhoz tartozó metódusok elnevezése:
    ```
    <típus> getValami()
    void setValami(<típus> value)
    ```
  - logikai típusú változó esetén:
    ```
    boolean isValami()
    void setValami(boolean value)
    ```

ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

JavaBean

Olyan osztály, amely

- rendelkezik paraméter nélküli konstruktorral
- privát mezői getter/setter metódusokon keresztül érhetők el
- implementálja a `java.io.Serializable` interfészt

ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

Osztály - forrásfájl viszony

- Általában egy .java fájlba egyetlen osztályt vagy interfészt írunk.

Szabályok:

- Egy forrásfájlban legfeljebb egy publikus osztály lehet
  - nem publikus osztályból lehet több
- A fájl neve meg kell egyezzen a benne található publikus osztály nevével

Láthatóságok

- public, protected, private értelmezése a C#-ban szokásos
- "internal"-nak megfelelő láthatóság nincs
- default (jelölés nélküli) láthatóság: **`package-private`**
  - NEM private!
  - az osztályon belül **ÉS a csomagon belül is** látható

Objektumok inicializálása és megszűnése

- Konstruktorok működése a szokásos.
- Inicializáló blokk
  - név nélküli `{ }` blokk
  - a konstruktor előtt, de az ős konstruktor hívása után fut le
  - statikus inicializáló blokk: `static { }`
- Destruktor: nincs
  - Az Object-től örökölt `finalize()` metódus felüldefiniálható

ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

Eltérő kulcsszavak

Java	C#	Magyarázat
`extends`	`: operátor`	öröklés
`implements`	`: operátor`	implementálás
`super`	`base`	ősosztályra hivatkozás
`final`	`sealed`	lezárt osztály
`final (static final)`	`const`	konstans tagváltozó
`instanceof`	`is`	típus ellenőrzés
`boolean`	`bool`	logikai típus
`try (<resource>)`	`using (<resource>)`	erőforrás felszámolása a blokk végén
-	`virtual, override`	Javában a metódusok alapértelmezetten virtuálisak

ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

További szintaktikai különbségek

- Foreach: szintén a `for` kulcsszóval fejezzük ki

```
for (int i : numbers) {    }
```

- Switch
  - nem kötelező a case blokkok végén a break
  - ilyenkor továbbugrik a vezérlés a következő case ágra (fall-through logic)

ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

A Javában nincs…

- előjel nélküli számtípus
- többdimenziós tömb (pl. `int[,]`)
  - de fűrészfogas tömb (tömbök tömbje) van
- delegate típus, event
  - de van lambda, functional interface
- operátor felüldefiniálás, indexer
- struct (érték típusú struktúra)

ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

Források

- Dr. Szénási Sándor előadás prezentációja és példái
- Java Documentation
  https://docs.oracle.com/en/java/
- Ant vs Maven vs Gradle
  https://www.baeldung.com/ant-maven-gradle