**Általános információk:**

**Tárgy neve: Programozás 2**

**Tárgy kód: IVB305MLMI**

**Szemester:** 2

**Kredit: 4**

**Órák száma:** 2 előadás, 3 labor

**Követelmény: Félévközi jegy**

**Előfeltétel: Programozás 1**

**Oktató: Zidarics Zoltán**

7624, Pécs, Boszorkány u. 2. B233

E-mail: [z](mailto:perjesi@mik.pte.hu)amek@vili.pmf.hu

Tel: +36 72 503650

**Java programozás**

Programozás Java nyelven. **A tárgy az alábbi ismeretköröket nyújtja a hallgatók számára:**

1. Java nyelv használata különbüző operációs rendszereken,
2. Eclipse használata mint IDE eszköz
3. Alapos tudás objektum orientált szemléletben
4. Alap aktivitások:
   1. projekt létrehozáa
   2. teszt környezet létrehozása
   3. kód dokumentálása
   4. XML és JSON használata
   5. különböző build tool-ok használata (Maven, Ant, Gradle)
   6. verzió kezelő használata (Git, Gitlab)
   7. Java alapok (Listák, alap I/O, adatbázisok, grafikus felület, szálak, generikus programozás, Funkcionális programozás, tervezési minták)

**Általános tárgyleírás:**

Rövid program: Az elméleti és gyakorlati órák során a hallgatók alap jártasságokat szereznek az alábbi témakörökben:

* Java projektek létrehozása és menedzsmentje
* Kódok megosztása verziókezelő rendszerek segítségével
* Csoportmunkában történő munkavégzés, szoftver életciklus menedzsment.

A hlaggatók megtanulják a programozás alapjait a későbbi mérnöki feladatok megoldásának segítésére. A gyakorlati órákon szerzett ismeretkelle kiegészítik az elméleti tudásukat.

**Módszertan:**

A prezentációk fontos részét képezik az oktatásnak, megkönnyítve az egyes szoftver fejlesztési technikák elsajátítását. A gyakorlatok során kiemelt figyelmet fordítunk a Java projektek menedzsmentjére.

**Időbeosztás:**

Study period in 15 weeks: February 5 - May 18 (2018)

1. Introduction of Java language. Why Java? Object Oriented Paradigm and creating projects.
2. XML and JSON, Java build tools, Version controlling (**Homework I**)
3. Using Lists (**Homework II**), Basic I/O (**Homework III**), Database Management
4. Graphical surface, Threads in Java, **Generics in Java**
5. Functional programming, Design patterns

pótlási időszak: May 22-24 (2018)

teljesítés: félévközi jegy

**Részvétel:**

Az előadásokon a részvétel nem kötelező, de a sikeres teljesítés feltétele az összes házi feladat megléte és a féléves feladat leadása, és igy legalább 65% teljesítés.

**Értékelés**

Az egyes feladatok az alábbi súlyozással kerülnek értékelésre:

1. tesztek 50%
2. Házi feladat 20 %.
3. féléves feladat 30 %
4. megajénlott jegy: min. 55 % elérése a szorgalmi és pótlási időszakban.
5. A vizsgaidőszak kezdetéig elkészült féléves feladat, minimum 55% szükséges a féléves feladat beszámításához

**Értékelés**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Jegy | 5 | 4 | 3 | 2 | 1 |
| Elért száazalék | 89%-100% | 77%-88% | 66%-76% | 55%-65% | 0-54% |

**Szakirodalom:**

**The Java tutorials** [**https://docs.oracle.com/javase/tutorial/index.html**](https://docs.oracle.com/javase/tutorial/index.html)

**XML tutorial** [**https://www.w3schools.com/xml/default.asp**](https://www.w3schools.com/xml/default.asp)

**JSON tutorial** [**https://www.w3schools.com/js/js\_json\_intro.asp**](https://www.w3schools.com/js/js_json_intro.asp)

**Apache Ant manual** [**https://ant.apache.org/manual/**](https://ant.apache.org/manual/)

**APACHE maven manual** [**https://maven.apache.org/what-is-maven.html**](https://maven.apache.org/what-is-maven.html)

**GIT documentation** [**https://git-scm.com/doc**](https://git-scm.com/doc)

**gitlab documentation** [**https://docs.gitlab.com/**](https://docs.gitlab.com/)

**DESIGN PATTERNS https://en.wikipedia.org/wiki/Software\_design\_pattern**