# Progtech 1

1. Objektum orientált programozás nagy vonalakban, osztály, objektum:

* <https://docs.oracle.com/javase/tutorial/java/concepts/index.html>

2. A java használatához alapok, nyelvi elemek bemutatása, változók, ciklusok, elágazosok:

* <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/index.html>

3. Objektim orientált programozás java nyelven: osztályok definiálása, objektumok:

* <https://docs.oracle.com/javase/tutorial/java/javaOO/index.html>

4. Öröklődés, interfészek

* <http://docs.oracle.com/javase/tutorial/java/IandI/index.html>

5. Collections

* <http://docs.oracle.com/javase/tutorial/collections>

6. Kivételek

* <http://docs.oracle.com/javase/tutorial/essential/exceptions/index.html>

7. UML - Osztály diagram:

* <http://www.agilemodeling.com/artifacts/classDiagram.htm>

# Progtech 2

1. Swing componensek:

* <https://docs.oracle.com/javase/tutorial/uiswing/index.html>

2. Szálkezelés:

* <https://docs.oracle.com/javase/tutorial/essential/concurrency/index.html>

3. JDBC:

* <https://docs.oracle.com/javase/tutorial/jdbc/index.html>

## Clean code

* Robert C. Martin - Clean Code: A Handbook of Agile Software Craftsmanship

([https://www.dropbox.com/s/255exvgg5pfp3su/Martin\_-\_Clean\_Code\_-\_A\_Handbook\_of\_Agile\_Software\_Craftsmanship.pdf](https://www.dropbox.com/s/255exvgg5pfp3su/Martin_-_Clean_Code_-_A_Handbook_of_Agile_Software_Craftsmanship.pdf?dl=0))

Chapter 1: Clean Code

Chapter 2: Meaningful Names

Chapter 3: Functions

Chapter 4: Comments

Chapter 5: Formatting

Chapter 6: Objects and Data Structures

Chapter 7: Error Handling

Chapter 9: Unit Test

Chapter 10: Classes

Chapter 13: Concurrency