



Junior Programmer Course

- JAVA -

course description

Nowadays the digitalization is more than just a phrase, it has serious effects on all areas of our lives and drives the major changes in the labour market as well. This digital transformation generates the need of an unbelievable amount of IT professionals, that has strong influence on the overall productivity of companies, since the needed competence became a bottleneck in satisfying the business needs.

our mission

Our mission is to support our partners in this challenging environment and provide the proper competence on an alternative way that can help to eliminate this gap, by finding potential candidates on the labour market, help them change their specialization and grow as an IT professional.

MATRITEL 
#1labs

Course thematics

- J A V A -

mathematical basics

The basics of mathematical background helps to settle the mindset that is needed for programming, and make easier to understand what happens in the background.

- technical details -

The digital computers used today are binary. In programming to be more human-readable other numeral systems are also used. Working with binary numeral system also introduces new logical operations which are used throughout the programming world. This section provides an introduction to the mathematical basics needed to understand this concepts.

computer

We'd like to build a solid baseline of the main concepts about computers and networking. This knowledge is elementary to understand since most of these are needed for daily work.

- technical details -

The understanding of the data flow in a computer is explained. To understand how data is stored in the computer, how the CPU reads/writes/processes it, what memory is used for, how the data is stored and what the basics of networks are where this data is transferred.

programming

The first part of the basics that has to be rock solid in everyone's head to be able to build up the knowledge base. It's programming itself, this is the point where we start to learn how to code, what units, objects, elements are needed to create a working code. We'll cover not just the pure technical details, but give an overview about programming languages, and development methodologies.

- technical details -

Common abstract concepts that are used in most of programming languages like variables, data types, data structures. The process from creating the first text file to running program is the main theme of this part. How source code is written, where the program starts, how systems can be structured, how exceptions are handled.

JAVA

As we have the basics to be able to start develop an application, we need a framework and a language with it's rules and opportunities as the tool what is used to create our working software. This topic is about learning the language that will be used for development.

- technical details -

JAVA is one of the most prevalent programming languages used today. A lot of low-level functionality is handled by the virtual machine which makes it easier for beginners to get a grasp on programming, without having to learn all at once. Groovy helps even further since it handles a lot of syntactical constraints put in place by Java, helping the first steps.

Course thematics

- J A V A -

planning

Careful planning is really important, to make re-usable and effective software with the proper interfaces, so in this topic we'll concentrate on the pre-development activities, and show how to turn the plans into code.

- technical details -

Complexity of a software system can grow exponentially the bigger it or the more complicated the problem it solves gets. Design tools/concepts help to handle this complexity by using proven patterns in code and high-level abstraction methods to take problem and solution definition to a human scale.

development

This topic is about the framework, that will be used for the development, we'll show different options, and go through on the tooling and the steps of the whole development process.

- technical details -

The programming development process over the years got a lot easier thanks to proven processes and tools. Integrated development environments, automated build tools, versioning systems are there to make a particular part of the development process easier. Knowing and applying these is a basic skill programmers should have.

testing

Quality first is an important mindset nowadays, and testing and test automation became a vital part of the development. Here we show the basics theories, tools and methodologies to give a starter pack to our programmers.

- technical details -

Testing is at least as important as development. Different levels of software testing are demonstrated. Test driven development gives an example of yet another development process.

collaboration

Nowadays collaboration is the same important as the coding itself since mobility and virtual connection are the keys in many different cases. Distributed, multi-site, teams are really usual and most of the companies are using a wide range of tools to support them. This topic helps to get know the most popular tools that are used in professional environments.

- technical details -

Software development is a team effort and needs to be trackable by other entities in the process. Creating the source code together requires communication, collaboration and when needed technical conflict resolution. Team chat tools, issue tracking solutions and version control system are an integral part of and used throughout the course.

INTERNSHIP

As part of our course, there is a possibility to complete the last 2 months of the training as part of an internship at one of our partner companies. This way students are able to gain experience in a real working environment already in the very beginning.