



OSNOVA KURZU

# Java Akademie

*Poslední aktualizace:*  
16. 11. 2020

## Obsah

<b>1 Úvod do Java Akademie</b>	<b>3</b>
<b>2 Podmínky a cykly</b>	<b>3</b>
<b>3 Git</b>	<b>3</b>
<b>4 Základy OOP a Maven</b>	<b>3</b>
<b>5 Kolekce v Javě</b>	<b>4</b>
<b>6 Základy testování v JUnit</b>	<b>4</b>
<b>7 Algoritmizace, abstraktní datové struktury a složitost</b>	<b>4</b>
<b>8 IO, streamy, closeables</b>	<b>4</b>
<b>9 Základy funkcionálního programování, streamy</b>	<b>5</b>
<b>10 Práce s vlákny</b>	<b>5</b>
<b>11 Pokročilé testování, koncepty reflexe a anotace</b>	<b>5</b>
<b>12 Rekapitulace</b>	<b>5</b>



## 1 Úvod do Java Akademie

V první lekci ti představíme obsah celého kurzu, poznáš ostatní studenty a probereme historii a základní syntax jazyka Java. Navážeme představením základních pojmů jako proměnná, operátor, výraz a datový typ, jichž si několik ukážeme.

Představíme ti také nástroje, které budeš v průběhu kurzu používat.

## 2 Podmínky a cykly

Představíme si řídicí struktury jazyka, práci s výjimkami. Z nástrojů používaných při vývoji si na projektu jednoduché kalkulačky představíme debugger. Ten významně usnadňuje ladění chyb. Od této lekce taky začneš dostávat programátorské domácí úkoly.

## 3 Git

Ve třetí lekci se zaměříme na verzovací systém Git, který budeš odteď používat při vývoji a odevzdávání domácích úkolů. Vysvětlíme ti základní pojmy jako repozitář, commit, branch, merge, rebase a push. Založíš si účet na některé ze služeb, které poskytují hosting Git repozitářů, a zprovozníš si Git na svém počítači.

## 4 Základy OOP a Maven

V této lekci si ukážeme nejsilnější koncepty jazyka Java – vysvětlíme ti, co jsou to objekty, třídy a metody, jak funguje dědičnost a co je to kompozice. Také se podíváme na to, jak strukturovat program pomocí packages. Z oblasti vývojářských



nástrojů si představíme Maven, který ti umožní program zkompileovat nezávisle na IDE.

## 5 Kolekce v Javě

Budeme se věnovat základním datovým strukturám v Javě – kolekcím, konkrétně List, Set, Map. Představíme ti metody equals() a hashCode() a vysvětlíme, možná trochu filosofický, problém identity a rovnosti dvou objektů. Taky se poprvé potkáš s generiky. To vše v páté lekci.

## 6 Základy testování v JUnit

V šesté lekci se podíváme na oblast, kterou by měl každý programátor zvládat, ale je často opomíjená – testování. Na knihovně JUnit ti ukážeme základní koncepty, vysvětlíme, co je to mock, představíme taky testovací pyramidu a myšlenkový proces – přístup k vývoji umožňující ti využít výhody, které testy nabízejí. Od této lekce už nebude žádný domácí úkol bez testů kompletní.

## 7 Algoritmizace, abstraktní datové struktury a složitost

Sedmá lekce bude více teoretická. Podíváme se opět na známé datové struktury, ke kterým si přidáme ještě strom, ale tentokrát z hlediska teorie. Vysvětlíme ti, co je to algoritmus, představíme základní algoritmy nad našimi strukturami a řekneme si, co je časová a prostorová složitost a jak je určit.



## 8 IO, streamy, closeables

Osmá lekce se bude věnovat vstupu a výstupu (IO) v Javě. Ukážeme ti základní práci se soubory a cestami, základy práce se Stringy a vysvětlíme, co je to Closeable a konstrukci try-with-resources. Představíme ti také větší projekt, na kterém budeš pracovat do konce kurzu.

## 9 Základy funkcionálního programování, streamy

V deváté lekci se zaměříme na novinku přidanou v Javě 8 – stream API v kolekcích a porovnáme si je s klasickým přístupem pomocí cyklu. Abys je mohl/a efektivně používat, představíme ti koncepty jako lambda, method reference a functional interface, které má Java na podporu funkcionálního programování.

## 10 Práce s vlákny

Znalosti z desáté lekce ti umožní psát programy, které dělají více věcí zároveň a lépe tak dokážou využít možností moderních vícejádrových procesoru. Řekneme si, co je vlákno, představíme problém sdílení paměti mezi vlákny a co se s ním dá dělat. Necháme filosofy navečeřet.

## 11 Pokročilé testování, koncepty reflexe a anotace

Jedenáctou lekci začneme pokročilým testováním ve frameworku JUnit, ve kterém už teď umíš vytvářet základní testy. V průběhu lekce se podíváme na koncept reflexe, kde můžeš zjistit informace o třídě, případně její instanci a za běhu měnit hodnoty atributů. Také se mrkneme na anotace, co jsou doprovodné informace k elementu v kódu – třídě, metodě nebo jejímu parametru.



## 12 Rekapitulace

A jsme u konce! V poslední dvanácté lekci si společně zopakujeme nejdůležitější části kurzu. Shrňeme nabyté znalosti a projdeme projekty. Ukážeme ti vzorové řešení finálního projektu, zodpovíme všechny dotazy a případně dovysvětlíme nejasná témata.

