Jazyk Scala - programování I

Kód kurzu: SCALA_INTRO

Absolvent bude do hloubky obeznámen s OOP i funkcionálními technikami a jejich využitím v silně typovém jazyce. Kurz začíná základními konstrukcemi jazyka, pokračuje s OOP, poté funkctionálními transformacemi, typovým systémem jazyka a končí asynchronním programováním.

Požadované vstupní znalosti:

- Kurz předpokládá znalost jakéhokoli jiného programovacího jazyka a základů algoritmizace.

Metody výuky:

- Výklad s prezentací
- Praktické ukázky na malých útržcích kódu, tzv. "tabule" nebo scratches v InteliJ Idea
- Málá cvičení pro vyzkoušení právě vysvětleného
- Velké samostatné úkoly z tématického celku vyhodnocené individuálně pro každého studenta (s radami jak dále)

Studijní materiály:

- Prezentace probírané látky v tištěné nebe online formě.

Osnova:

Instalace potřebných nástrojů/Scaly a úvod

- Instalace Scala-cli
- Scala v InteliJ Idea (Scala plugin)
- Co je Scala, malá ukázka cílů a směrování kurzu is praktickými příklady
- Vytvoření projektu

Základní konstrukce jazyka a úvod do typů

- Měnitelné vs neměnitelné proměnné - princip immutable
- Základní typy: numerické, řetězce, pravdivostní hodnoty
- Podmínky a cykly
- For cyklus který vrací výsledek, teorie výrazů - co je výraz a co ne
- Funkce, rekurze, tail rekurze
- Interpolace řetězců
- Typ tuple, "rozbití" na proměnné, pattern matching
- Typ option jako jednoprvková kolekce, řetězení options ve for cyklu, Option(null)
- Dopis/Seq/Vector/Set/Map a operace nad nimi
- Mutable varianty kontejnerů

Operace nad kolekcemi

- foreach, map, flatMap
- rekurzivní počítání s kolekcemi, foldLeft, sum, reduce
- find, headOption, filtr, exists, contains, collect, groupBy, mkString

OOP ve Scale

- Třída a její atributy, konstruktor, companion object a apply
- case class, metoda copy, pattern matching
- Traity a vícenásobná dědičnost
- Anonymní třídy
- sealed trait a enumerace

Pokročilejší konstrukce jazyka

- Vyšší funkce, funkce jako parametr a návratová hodnota
- Kdy se kód skutečně zavolá/vyhodnotí?
- Lambdy

GOPAS Praha
Na Strži 2097/63
140 00 Praha 4 - Krč
Tel.: +420 226 201 390
info@gopas.cz

GOPAS Brno
Nové sady 996/25
602 00 Brno
Tel.: +420 530 513 590
info@gopas.cz

GOPAS Bratislava
Dr. Vladimíra Clementisa 10
Bratislava, 821 02
Tel.: +421 902 903 132
info@gopas.sk

Copyright © 2026 GOPAS, a.s.,
All rights reserved

- Obalení primitivních hodnot do typů namísto přímého použití
- implicitní funkce, konverze
- implicitní třídy, doplňování metod
- implicitní hodnoty
- Try vs try
- Řetězení potencionálně neúspěšných operací ve for cyklu, recover, orElse
- Funkce s typovým parametrem
- Ohraničení typů
- Náš vlastní reduce

Asynchronní programování

- Future
- Await.ready/result
- Execution Context (global, fixed thread pool, cached, work stealing pool)
- Future neznamená thread, Co je to thread pool
- map, flatMap, skládání ve for cyklu
- andThen, recover, transform
- lazyness

GOPAS Praha
Na Strži 2097/63
140 00 Praha 4 - Krč
Tel.: +420 226 201 390
info@gopas.cz

GOPAS Brno
Nové sady 996/25
602 00 Brno
Tel.: +420 530 513 590
info@gopas.cz

GOPAS Bratislava
Dr. Vladimíra Clementisa 10
Bratislava, 821 02
Tel.: +421 902 903 132
info@gopas.sk

Copyright © 2026 GOPAS, a.s.,
All rights reserved