Programming Data Science at Scale
Lab Session 1

1 Introduction

This is the first lab session for the Programming Data Science at Scale course 2024/25. You need to use the Scala Collection API to solve problems you might encounter when working with collections.

2 Internet Movie Database (IMDB)

This assignment will be on processing a subset of the IMDB dataset to encourage you to think about how to efficiently process structured text using Scala Collections. We have provided a small subset of the following schema in List format for you, so you can start implementing tasks using that list as the input.

- `Option[T]` means either type `T` is present, or `skipVal` (`'\N'`) otherwise
- `List[T]` means a comma-delimited list of type `T` is present, e.g. '`dog,cat,bear`', where `T := String`

INDEX	FIELD	TYPE	EXAMPLES/NOTES
			title.basics.tsv
0	`tconst`	`String`	`ttXXXXXXX` – Unique title ID
1	`titleType`	`Option[String]`	'`tvMovie`', '`short`', '`movie`', '`videoGame`'
2	`primaryTitle`	`Option[String]`	–
3	`originalTitle`	`Option[String]`	–
4	`isAdult`	`Int`	–
5	`startYear`	`Option[Int]`	`YYYY` – Release year
6	`endYear`	`Option[Int]`	`YYYY` – End year, e.g. when a play ends.
7	`runtimeMinutes`	`Option[Int]`	–
8	`genres`	`Option[List[String]]`	'`Documentary,Short,Sport`'

3 Tasks

Download `imdb-scala-src.zip` and extract it somewhere on your machine. You have to complete the missing implementations (specified by `???`) in `src/main/scala/imdb/ImdbAnalysis.scala`.

You are encouraged to look at the Scala API documentation while solving this exercise, which can be found here:

`https://www.scala-lang.org/api/2.12.15/index.html`

Consult the schema in Section 2 when designing your solutions in order to extract the correct data.

Task 1

◁ **Task**

Return a list containing all `primaryTitles` for titles in `TitleBasics`.

> return type: List[String]
> title:String

Task 2

◁ **Task**

Return a list of key-value pairs of `primaryName` and `startYear` for titles released between 2010-2020 (inclusive).

> return type: List[(String, Int)]
> title:String
> start_year:Int

Task 3

◁ **Task**

Return a list of `primaryTitle` for titles that `titleType` equals to `'movie'`.

> return type: List[String]
> title:String

Task 4

◁ **Task**

Return a list of key-value pairs of `primaryTitle` and `runtimeMinutes` for titles having the minimum and maximum `runtimeMinutes`.

> return type: List[(String, Int)]
> title:String
> duration:Int

Task 5

◁ **Task**

Return average `runtimeMinutes` for titles where `titleType` equals to `'movie'`.

> return type: Float
> average_duration:Float

Task 6

◁ **Task**

Return a list of key-value pairs of `primaryTitle` and `runtimeMinutes` for titles having the minimum and maximum `runtimeMinutes` where `genres` contain `'Drama'`.

return type: List[(String, Int)]
title:String
duration:Int