Programare Avansată pe Obiecte

# Curs

Profesor: Radu Boriga (email [radu.boriga@fmi.unibuc.ro](mailto:radu.boriga@fmi.unibuc.ro))

[Drive cu cursuri](https://drive.google.com/drive/folders/1JYUioHwE_0w9N0GF_kHlfGFbwhpqLLM_?usp=sharing)

## Bibliografie

Lista completă în cursul introductiv

* [**Effective Java**](https://drive.google.com/open?id=1LulOyx01VBu--HlmDIQSwLFrljiugjh-) (3rd edition) de J. Bloch
* [**Modern Java in Action**](https://drive.google.com/open?id=138dGHAGUGK1ubhO7C9R3A_gVBkiUMsgA)de R. G. Urma
* Java 8 in Action, R. G. Urma

Tutoriale:

* <https://docs.oracle.com/javase/tutorial/>
* <https://www.tutorialspoint.com/java/index.htm>

## Curs 1 intro

* Ce este Java
* Scurt istoric
* JVM (Java Virtual Machine)
* JRE (Java Runtime Environment)
* JDK (Java Development Kit)

[Curs 01 PAO - seria 23 (2019-2020).pdf](https://drive.google.com/file/d/1ZqDL2IYpTotx21aAPVlRk3KR-RLZ9pY0/view?usp=sharing)

## Curs 2 class&objects

* Principiile POO
  + Abstractizarea
  + Încapsularea
  + Moștenirea
  + Polimorfismul
* Sintaxa pentru clase
* Moștenire (inheritance) vs compoziție (composition)
* Metode
* Constructori
* Referința *this*
* Ciclul de viață al obiectelor (garbage collection)

[Curs\_02\_PAO\_2019\_2020.pdf](https://drive.google.com/file/d/10eTIzHro1XJdhwP1ZhJvsnOdlFzGwrzc/view?usp=sharing)

## Curs 3 extinderea claselor

* Extinderea claselor (moștenire)
* Supraîncărcare (overloading) vs redefinire (overriding)
* Clasa *Object*
* Polimorfism
* Clase abstracte

[Curs\_03\_PAO\_2019\_2020.pdf](https://drive.google.com/file/d/1C4_hSIv6DCDwS7Tt1LFUFceho2AEeFL8/view?usp=sharing)

## Curs 4 stringuri&has-a

* Agregare și compoziție
* Șiruri de caractere
* Expresii regulate
* String
* StringBuilder
* StringBuffer
* Clase imutabile

[Curs\_04\_PAO\_seria\_23\_2019\_2020.pdf](https://drive.google.com/file/d/1-BH--oSvCt3a70pPk48RDJYr_HdCEZ4S/view?usp=sharing)

## Curs 5 interfete

* Interfețe
  + Definire
  + Implementare
  + Utilizare
* Interfețe Marker
* Decorator Pattern
* Schimbări ale interfețelor în Java 8/9

[Curs\_05\_PAO\_2019\_2020.pdf](https://drive.google.com/file/d/1igUAdSppHFDQadIY1ypamc0NWrUv0hHb/view?usp=sharing)

## Curs 6 exceptii

* Excepții
  + Checked vs unchecked exceptions
  + try - catch - finally
* Fluxuri de intrare/ieșire (I/O streams)
* Clasa File

[Curs\_06\_PAO\_2019\_2020.pdf](https://drive.google.com/file/d/1biKll1We86Dx8NTNsI84VZH1T0qBygpR/view?usp=sharing)

## Curs 7 colectii list, set, map

* Colecții
* List
  + ArrayList
  + LinkedList
* Set
  + HashSet
  + LinkedHashSet
  + TreeSet
* Map
  + HashMap
  + TreeMap
* Iterator

[Curs\_07\_PAO\_seria\_23\_2018\_2019\_Colectii.pdf](https://drive.google.com/file/d/1Vebcf5Q_54twc5vPM3Ny7ow_o8jMD18O/view?usp=sharing)

## Curs 8 lambdas

* Programare imperativă vs programare funcțională
* Lambda expresii
* Interfețele funcționale Predicate, Consumer, Function, Supplier
* Referințe către metode
* Metoda forEach

[Curs\_08\_PAO\_seria\_23\_2018\_2019\_Lambdas.pdf](https://drive.google.com/file/d/1f2LbjY-Tgk1B2MW9pFo7u85OtZOa92HR/view?usp=sharing)

## Curs 9 streams

* Streams API
* Operații pe Streams: filter/sorted/map/forEach/etc.
* Operații de închidere: reduce/min/max/collect/etc.

[Curs\_09\_PAO\_seria\_23\_2018\_2019\_Streams.pdf](https://drive.google.com/file/d/1SkjZGjszfPAbrKL0QyN0HjMT5G7RAntz/view?usp=sharing)

## Curs 10 threads

* Fire de execuție (Threads)
* Multithreading
* Interfața Runnable
* Ciclul de viață al unui fir de execuție
* Serializare / Externalizare

[Curs\_10\_PAO\_seria\_23\_2019\_2020.pdf](https://drive.google.com/file/d/1GzA964FOr7JpH-p3H1mZ6fgEoh5z0ipS/view?usp=sharing)

## Curs 11 socket, jdbc

* Client-Server Pattern
* Socket
* TCP (Transmission Control Protocol)
* JDBC (Java DataBase Connectivity)
  + Statement
  + PreparedStatement
  + CallableStatement

[Curs\_11\_PAO\_Socket\_JDBC.pdf](https://drive.google.com/file/d/16zI4xEU8psfy2DztTEugqDxc7mkMSy-8/view?usp=sharing)

## Curs 12 gui

* Interfețe Grafice (GUI)
* AWT (Abstract Windowing Toolkit)
* JFC (Java Foundation Classes)
  + Swing
  + Java2D
* Model-View-Controller Pattern
* Containere
  + JFrame
  + JPanel
  + JScrollPane
  + JTabbedPane
  + JSplitPane
* Componente
  + JLabel
  + JButton
  + JToggleButton
  + JCheckBox
  + JRadioButton
  + JList
  + JTable
* Gestionar de poziționare (LayoutManager)
  + FlowLayout
  + BorderLayout
  + GridLayout
  + GridBagLayout
* Event
  + ActionEvent
  + MouseEvent
  + WindowEvent
  + ItemEvent
  + AdjustmentEvent
* Listener

[Curs\_12\_GUI.pdf](https://drive.google.com/file/d/18_sLBqfa4f5dHsCU713d8dPBsBERas1K/view?usp=sharing)

## Curs 13 servlets

* Servlet
* CGI (Common Gateway Interface)
* Cereri HTTP
* JSP (Java Server Pages)

[Curs\_13\_PAO\_Servleturi.pdf](https://drive.google.com/file/d/1qpAbBMGlMoKBai1Wc-GKCMN_iWRNuexG/view?usp=sharing)

# Laborator

Laborant: Constantin Voinea (email [constantin.voinea@endava.com](mailto:constantin.voinea@endava.com))

GitHub-ul cu codul laborantului: <https://github.com/constantin-voinea/java-fundamentals>

Trebuie să instalăm [Java JDK 1.8](https://www.oracle.com/java/technologies/javase-jdk8-downloads.html) și un editor de text sau IDE (de exemplu [IntelliJ IDEA](https://www.jetbrains.com/idea/)).