Valid from 2024.HS

Module description: Programming 2

Module Code	t.BA.IT.PROG2.19HS
ECTS Credits	4
Language of Instruction/Examination	German
Organizational Unit	InIT
Module Coordinator	Christof Marti
Legal Framework	The module description is part of the legal basis in addition to the general academic regulations. It is binding. During the first week of the semester a written and communicated supplement can specify the module description in more detail.
Module Characteristic	Type 3b 2 lecture lessons per semester week and class+ 4 lab bi-weekly lessons per semester and half-class
Module Description	Students learn the extended concepts of object-oriented programming using the Java programming language and the associated Java Development Kit (JDK) environment. The objective is to create high-quality, comprehensible and easy-to-maintain program code using advanced methods, techniques and tools.

Module description: Programming 2

Module Content

Extended Java concepts

- Nested / Inner & Anonymous Classes
- Functional Programming: Lambda Expressions / Functional Interfaces
- Method References, Functions as arguments and return values
- Functional Streams

Concurrency

- Concurrency foundation
- The Thread lifecycle
- Thread management: Executor-Services / Thread-Pooling
- Callables / Futures
- Thread safety, Atomic Types
- Thread cooperation: Mutual Exclusion & Condition Synchronisation
- Extended Monitor concept, Lock & Conditions
- Avoiding Deadlocks

Simple Graphical User Interfaces (GUIs)

- GUI-Principles, -Architecture, -Components and -Layout
- Developing GUI applications using JavaFX and FXML
- Model-View-Controller / Presenter, & Observer-Pattern
- Dynamics of GUI: Event handling

Input and Output of data – Java IO

- Handling the file systems (files & directories)
- Reading and writing files
- The decorator pattern
- Handling Resources, Properties & Resource-Bundles
- Data encoding and character sets

Mock-Testing

- Principles and economics of software testing.
- Isolated testing: Testdoubles, Stubbing & Mocking
- State testing vs. behavior verification
- Writing Mock tests

Laboratories

- In the hands-on laboratories, students solve exercises tailored to the topic of the lecture.

Prerequisite Knowledge

Knowledge taught in IT.PROG1

Learning Objectives (Competences)

Students...	Competencies	Taxonomies
Students understand the concepts of concurrency and can correctly control and synchronize parallel tasks. You are able to identify and avoid problems.	F, M	K2, K3, K4
Students can design and build applications with a simple graphical user interface.	F, M	K3
Students understand the principle of Isolated Testing and can implement it with the use of mock tests.	M, F	K2, K3
Students understand the extended functional language concepts of Java (lambda, streams) and can use them in a systematic and efficient way.	F, M	K2, K3
Students can encode or decode data appropriately and save and or read it from files.	F, M	K2, K3

Module description: Programming 2

Performance Assessment

End-of-module exam	Assessment	Length (min.)	Weighting	Form
written exam	Grade	120	80	acc. to module agreement

Performance assessment during the semester	Assessment	Length (min.)	Weighting	Form
Programming assignments *Graded programming assignments in the Laboratory*	Grade	0	20	acc. to module agreement

Classroom Attendance Requirement

None

Learning material

- Barnes, D. & Kölling, M. (2017). Java lernen mit BlueJ. 6. aktualisierte Edition. Pearson Studium. ISBN 978-3-86894-911-7.

Comments