Syllabus

Kursbeschreibung

Titel der Lehrveranstaltung	Introduction to Programming
Code der Lehrveranstaltung	76401
Zusätzlicher Titel der Lehrveranstaltung	
Wissenschaftlich-disziplinärer Bereich	INFO-01/A
Sprache	Englisch
Studiengang	Bachelor in Wirtschaftsinformatik
Andere Studiengänge (gem. Lehrveranstaltung)	Bachelor in Computer Science
Dozenten/Dozentinnen	Prof. Chiara Ghidini, Chiara.Ghidini@unibz.it https://www.unibz.it/en/faculties/engineering/academic-staff/person/49601
Wissensch. Mitarbeiter/Mitarbeiterin	
Semester	Erstes Semester
Studienjahr/e	1
KP	9
Vorlesungsstunden	60
Laboratoriumsstunden	30
Stunden für individuelles Studium	135
Vorgesehene Sprechzeiten	27
Inhaltsangabe	• Data types and expressions • Basic data structures and generics • Functions and parameter passing • Conditionals and loops • Arrays and collections • Classes and objects • Basic Input/Output

	• Exception handling • Recursion
Themen der Lehrveranstaltung	The objective of the course is to teach the fundamental principles of programming. We will focus especially on imperative programing as the basic way to learn: (1) the basics of programming and programming elements; (2) the basics of algorithmic thinking; and (3) The basics of writing code. As programming language, we will use a subset of the Java language, mainly restricted to its imperative part. The student will learn how programs can be constructed, and also structured in more files/objects in order to solve a problem. Students will learn how to solve computational problems with well-designed programs that implement effective solutions. The learning will be based on examples, from very simple ones to more complex. We will use the Java programming language and the integrated development environment (IDE), so the goal is to train the student capability to develop java applications in this environment. The final objective for the student is to acquire the ability to solve basic algorithmic problems in a Java-based application.
Stichwörter	Programming, Algorithms, Java, Object Oriented
Empfohlene Voraussetzungen	There are no specific prerequisites. Basic notions of mathematics and set theory will be used.
Propädeutische Lehrveranstaltungen	
Unterrichtsform	The course includes frontal lectures with exercises, lab sessions, and individual programming projects.
Anwesenheitspflicht	Not mandatory, but highly recommended.
Spezifische Bildungsziele und erwartete Lernergebnisse	Type of course: "di base" for L-31 Scientific area: "Formazione informatica di base" for L-31 The objective of the course is to teach the fundamental principles of programming. We will focus especially on imperative programing as the basic way to learn: (1) the basics of programming and programming elements; (2) the basics of algorithmic thinking; and (3) The basics of writing code. As programming language, we will use a subset of the Java language, mainly restricted to its imperative part. The student will learn how programs can be constructed, and also structured in more files/objects in order to

	solve a problem. Students will learn how to solve computational problems with well-designed programs that implement effective solutions. The learning will be based on examples, from very simple ones to more complex. We will use the Java programming language and the integrated development environment (IDE), so the goal is to train the student capability to develop java applications in this environment. The final objective for the student is to acquire the ability to solve basic algorithmic problems in a Java-based application. Knowledge and understanding: • D1.3 - Know the basic principles of programming. Applying knowledge and understanding: • D2.2 - Ability to solve algorithmic problems using programming methods. • D2.17 - Know how to manage small projects for the development of information systems and how coordinate small working groups. Communication skills • D4.5 - Ability to collaborate in interdisciplinary teams to achieve IT objectives. Learning skills • D5.1 - Learning ability to undertake further studies with a high degree of autonomy.
Spezifisches Bildungsziel und erwartete Lernergebnisse (zusätzliche Informationen)	
Art der Prüfung	The assessment consists of a a final written exam, performed using laptops provided by the Faculty of Engineering, consisting both of practical components (coding questions) and theoretical components (questions about the basic notions of computer science and programming languages). The practical questions are designed to evaluate learning outcomes related to the application of acquired knowledge, critical thinking, communication, and learning skills. Specifically, students are expected to write code capable of effectively solving a basic

	computing problem. The theoretical questions assesse knowledge and understanding, the ability to apply that knowledge, and the student's learning skills. It includes verification questions and transfer-of-knowledge questions.
Bewertungskriterien	The practical questions will be assessed based on the correctness and quality of the solution, including the relevance and effectiveness of the implemented functions. Written exam answers will be evaluated based on their correctness and clarity.
Pflichtliteratur	John Lewis and William Loftus, Java Software Solutions, Pearson, 2018. Kathy Sierra, Bert Bates, Trisha Gee, Head First Java, 3rd Edition. O'Reilly Media, Inc. Cay S. Horstmann, Brief Java: Early Objects. Wiley Subject Librarian: David Gebhardi, David.Gebhardi@unibz.it
Weiterführende Literatur	The Java Tutorials: https://docs.oracle.com/javase/tutorial/
Weitere Informationen	Software used: IntelliJ IDEA (https://www.jetbrains.com/idea/)
Ziele für nachhaltige Entwicklung (SDGs)	Industrie, Innovation und Infrastruktur, Hochwertige Bildung