Syllabus: Advanced Java Programming
TU Berlin Summer School 2025
Term 2

Course Instructor: Prof. Dr. Rand Kouatly
Location: TU Berlin

Key to Learning Formats

Lecture	Practical	Excursion	Assessment	Orientation or Cultural Program Session*

*The cultural program timetable will be emailed to you shortly before your course starts. For more information about the cultural program, and for examples of previous schedules, check out our website: https://www.tu.berlin/en/international/summer-school/cultural-program-1

Week 1: July 21 - 25

Date	Session Information	Details
Monday, July 21	**Welcome and Introduction** TU Berlin Summer School Team Time CEST: 10:30am -12:30pm pm	Welcome to the TU Berlin, overview of the Summer School program, distribution of the welcome pack and introduction.
	Lunch Time CEST: 12:30pm -1:00 pm	
	Campus Tour with Summer School staff Time CEST: 1:00 -1:45pm	
	Introduction to Programming **Prof. Dr. Rand Kouatly** Time CEST: 1:45 -3:45 pm Format: Lecture	Instruction in programming and Java • Installing the Java Virtual Machine, on your computer • Time frame. • Install the Java IDE. • First Program “hello world”

Tuesday, July 22	**Variables and Data Type** **Prof. Dr. Rand Kouatly** Time CEST: 9:00 -10:30 am Format: Lecture	Introduction to the course and using Variable and Data types in Java.
	Condition and Operators **Prof. Dr. Rand Kouatly** Time CEST: 11:00am -12:30pm Format: Lecture	Includes: 1- Program Control Flow. 2- IF and IF ELSE statements. 3- Nested IF statement. 4- Switch Statement
	Programming Practical 1 **Prof. Dr. Rand Kouatly** Time CEST: 1:30 -3:30 pm Format: Practical	You will work through the instructions for the programming class. The lecturer will check your answers monitor your progress, and solve the exercises
Wednesday, July 23	**Loops** **Prof. Dr. Rand Kouatly** **& Visited Lecturer** Time CEST: 9:00 -10:30am Format: Lecture	Includes: 1- What is the Loops, 2- While Loops, 3- For Loops, 4- Nested Loops, 5- Do Loops, 6- Advanced Loop Statement, 7- Infinite Loops,
	Programming Practical 2 **Prof. Dr. Rand Kouatly** **& Visited Lecturer** Time CEST: 11:00am -12:30pm Format: Practical	You will work through the instructions for the programming class. The lecturer will check your answers monitor your progress, and solve the exercises
Thursday, July 24	**Lecture 4: Array and Lists** **Prof. Dr. Rand Kouatly** **& Visited Lecturer** Time CEST: 9:00 -10:30am Format: Lecture	Includes: 1- Definition, 2- One dimensional array, 3- Two-dimensional array, 4- Multi-dimensional array, 5- Java List, 6- Array Lists, **7-** Linked Lists,
	Programming Practical 3 **Prof. Dr. Rand Kouatly** **& Visited Lecturer** Time CEST: 11:00am -12:30pm Format: Practical	You will work through the instructions for the programming class. The lecturer will check your answers monitor your progress, and solve the exercises
	Programming Practical 4 **Prof. Dr. Rand Kouatly** **& Visited Lecturer** Time CEST: 1:30 -3:30 pm Format: Practical	You will work through the instructions for the programming class. The lecturer will check your answers monitor your progress, and solve the exercises
Friday,	**NO CLASS**	

July 25	**Daytrip to Potsdam** *Schedule will be announced*

Week 2: July 28 – August 1

Date	Session Information	Details
Monday, July 28	**Functions and Modules** **Prof. Dr. Rand Kouatly** Time CEST: 9:00 -10:30am Format: Lecture	Includes: 1- Definition, 2- Static Function, 3- Modules, 4- Libraries,
	Methods and Classes **Prof. Dr. Rand Kouatly** Time CEST: 11:00am -12:30pm Format: Lecture	Includes: 1- Introduction to Object Oriented Programming, 2- Classes and Objects, 3- Methods return values, 4- "this" keywords 5- Constructor, 6- Encapsulation and data hiding,
	Programming Practical 5 **Prof. Dr. Rand Kouatly** Time CEST: 1:30 -3:30pm Format: Practical	You will work through the instructions for the class for programming. The lecturer will check your answers monitor your progress, and solve the exercises
Tuesday, July 29	**Inheritance in Java Part 1** **Prof. Dr. Rand Kouatly** Time CEST: 9:00 -10:30am Format: Lecture	Includes: 1- Polymorphism, 2- Method Overloading, 3- Inheritance, 4- Method Overriding, 5- Constructors and Inheritance, 6- Supper keyword,
	Inheritance in Java Part 2 **Prof. Dr. Rand Kouatly** Time CEST: 11:00am -12:30pm Format: Lecture	Includes: 1- Multilevel Inheritance, 2- Abstraction, 3- Interface, 4- Lambda Function, 5- Function interface,
	Programming Practical 6 **Prof. Dr. Rand Kouatly** Time CEST: 1:30 -3:30 pm Format: Practical	You will work through the instructions for the programming class. The lecturer will check your answers monitor your progress, and solve the exercises
Wednesday, July 30	**Using GitHub with Eclipse** **Prof. Dr. Rand Kouatly** **& Visited Lecturer** Time CEST: 9:00 -10:30am Format: Lecture	Includes: 1- Create Got Hub Groups 2- Create Projects 3- Export and Import Projects 4- Using Branches, 5- Update Works, 6- To Lists in Eclipse, 7- Push the Updated Works,

		8- Merge the Project,
	First Project Groups **Programming Practical 7** **Prof. Dr. Rand Kouatly** **& Visited Lecturer** Time CEST: 11:00am -12:30pm Format: Practical	Solve Project 1 in Group working using GitHub
Thursday, July 31	**Data Type** **Prof. Dr. Rand Kouatly** **& Visited Lecturer** Time CEST: 9:00 -10:30am Format: Lecture	Includes: 1- Definition, 2- User-defined datatype, 3- Generic, 4- Generic Methods, 5- Bounded Type Parameters, 6- Generic Classes, 7- Hashing, 8- Hash Set, 9- Hash Table,
	Programming Practical 8 **Prof. Dr. Rand Kouatly** **& Visited Lecturer** Time CEST: 11:00am -12:30pm Format: Practical	You will work through the instructions for the programming class. The lecturer will check your answers monitor your progress, and solve the exercises
	Visit the German Museum of Technology & Visited Lecturer Time CEST: 1:30 -3:30 pm Format: Excursion	The Museum is located on the historic site of the Anhalter freight station and the Gleisdreieck transportation hub. The "raisin bomber" on the façade has since become an internationally recognized landmark. Its 28,500 square meter premises is filled with exhibitions covering aeronautics, navigation, railways, automobility, telecommunications, computer history, the world of sugar, and much more
Friday, August 1	**NO CLASS**	
	Optional daytrip to Dresden *Schedule will be announced*	

Week 3: August 4 - 8

Date	Session Information	Details
Monday, August 4	**Graphical User Interface** **Prof. Dr. Rand Kouatly** Time CEST: 9:00 -10:30am Format: Lecture	Includes: 1- Definition, 2- Examples of GUI Applications, 3- Containers and Components, 4- What Event, 5- Event Listeners,
	Second Project Groups **Programming Practices 9**	Solve Project 2 in Group working using GitHub

	Prof. Dr. Rand Kouatly Time CEST: 11:00am -12:30 pm Format: Practical	
	Second Project Groups Programming Practices 10 **Prof. Dr. Rand Kouatly** Time CEST: 1:30 -3:30 pm Format: Practical	Solve Project 2 in Group working using GitHub
Tuesday, August 5	**Exception and Exception handling** **Prof. Dr. Rand Kouatly** Time CEST: 9:00 -10:30am Format: Lecture	Includes: 1- Definition, 2- Examples of Exception, 3- Types of Exception, 4- Exception handling
	Input and Outputs **Prof. Dr. Rand Kouatly** Time CEST: 11:00am -12:30pm Format: Lecture	Includes: 1- Definition, 2- Reading files, 3- Writing to files, 4- Using Sound, 5- Using Images
	Programming Practical 11 **Prof. Dr. Rand Kouatly** Time CEST: 1:30 -3:30 pm Format: Practical	You will work through the instructions for the programming class. The lecturer will check your answers monitor your progress, and solve the exercises
Wednesday, August 6	**Core Advanced Topics** **Prof. Dr. Rand Kouatly** Time CEST: 9:00 -10:30am Format: Lecture	Includes: 1. Collections Framework. 2. Deep dive into collection interfaces, classes, custom implementations, and performance tuning. 3. Introduction to Concurrency and Multithreading.
	Programming Practical 12 **Prof. Dr. Rand Kouatly** Time CEST: 11:00am -12:30pm Format: Practical	You will work through the instructions for the programming class. The lecturer will check your answers monitor your progress, and solve the exercises
Thursday, August 7	**Third Group Project** **Prof. Dr. Rand Kouatly** Time CEST: 9:00 -10:30 am Format: Lecture	Includes: 1. Definition, 2. Project Analysis and Description, **3.** Dividing the Groups and rules,
	Programming Practical 13 **Prof. Dr. Rand Kouatly** Time CEST: 11:00am -12:30 pm	Solve Project 3 in Group working using GitHub

	Format: Practical	
	Programming Practical 14 **Prof. Dr. Rand Kouatly** Time CEST: 1:30 -3:30 pm Format: Practical	Solve Project 3 in Group working using GitHub
Friday, August 8	**NO CLASS**	

Week 4: August 11 – 15

Date	Session Information	Details
Monday, August 11	**Database and Persistence** **Prof. Dr. Rand Kouatly** Time CEST: 9:00 -10:30am Format: Lecture	Include: 1. Introduction to Database, 2. JDBC (Java Database Connectivity): 3. Advanced database connectivity techniques. 4. Connection pooling. 5. ORM Tools: Introduction to Hibernate or JPA for object-relational mapping. 6. Transaction Management: Handling complex database operations.
	Programming Practical 15 Time CEST: 11:00am -12:30pm Format: Practical	You will work through the instructions for the programming class. The lecturer will check your answers monitor your progress, and solve the exercises
	Programming Practical 16 Time CEST: 1:30 -3:30 pm Format: Practical	You will work through the instructions for the programming class. The lecturer will check your answers monitor your progress, and solve the exercises
Tuesday, August 12	**Enterprise Application Development 1** **Prof. Dr. Rand Kouatly** Time CEST: 9:00 -10:30am Format: Lecture	Include: 1. Servlets and JSPs 2. Advanced concepts like filters and listeners. 3. Session management techniques. 4. Introduction to Spring Framework, 5. Preparing the Working Environment, 6. Creating First Application
	Enterprise Application Development 2 **Prof. Dr. Rand Kouatly** Time CEST: 11:00am -12:30pm Format: Lecture	Include: 1. Dependency Injection (DI) and Aspect-Oriented Programming (AOP). 2. Data access with Spring Data. 3. RESTful Web Services. 4. Cloud Integration: Deploying applications on AWS or Azure, 5. Project 4 Description
	Programming Practical 17 **Prof. Dr. Rand Kouatly**	You will work through the instructions for the programming class. The lecturer will check your answers monitor your progress, and solve the exercises

	Time CEST: 1:30 -3:30 pm Format: Practical	
Wednesday, August 13	**Group Project** **Prof. Dr. Rand Kouatly** Time CEST: 9:00 -10:30am Format: Practical	Solve Project 4 in Group working using GitHub
	Solve Project in Group working using GitHub Time CEST: 11:00am -12:30pm Format: Practice	Solve Project 4 in Group working using GitHub
Thursday, August 14	**Course Recap 1** **Prof. Dr. Rand Kouatly** Time CEST: 9:00 -10:30 am Format: Lecture	Review and questions answers
	Course Recap 2 **Prof. Dr. Rand Kouatly** Time CEST: 11:00am -12:30pm Format: Lecture	Review and questions answers
	Free Study Exam Preparation Time CEST: 1:30 -3:30 pm Format: Practical	Study and prepare for the exam
Friday, August 15	**Assessment** **Prof. Dr. Rand Kouatly** Time CEST: 9:00 -10:30am Format: Assessment	Students will have to do some exercises on real time,
	Certificate Ceremony Start Time CEST: 4:00pm	

Student Learning Outcomes

The following overview provides information on the key topics to be covered each week and how they relate to the learning goals in your course.

Week 1

Objective/topic:

- Students will gain a general introduction to programming and gain basic knowledge of Java and Computer architecture about the von Neumann computer structure, pseudocode and machine language, the differences between Java version, what is the Java Virtual machine, Java applications, and why Java is different from other programming languages.

- Students will learn how to install the needed software, Java Virtual Machine, and Eclipse IDE.
- Students will learn the first basic instruction and concepts in Java programming.

Learning Goal(s):

- Students will understand the context of programming, and computers.
- Students will prepare the software environments for the course.

Week 2

Objective/topic:

- Students will understand data types and how they work in Java.
- Students will learn and comprehend the basic concepts of using functions or static methods and libraries in Java, as well as applying recursion to solve complex problems.
- Students will understand the principles of object-oriented programming (OOP).
- Students will grasp the concept of inheritance, including its types, member access rules, the usage of the this and super keywords, method overloading, method overriding, abstract classes, dynamic method dispatch, and the final keyword.
- Students will apply object-oriented programming techniques to projects through collaborative group work.

Learning Goal(s):

- Understanding Fundamental Java Concepts: Gain a strong understanding of Java data types, their functionality, and how they are used in programming.
- Building Problem-Solving Skills: Learn to solve complex problems using recursion and leverage static methods and libraries for effective code reuse and modular design.
- Mastering Object-Oriented Programming (OOP): Develop a comprehensive understanding of OOP principles, such as encapsulation, inheritance, polymorphism, and abstraction.
- Advanced Understanding of Inheritance: Understand inheritance in-depth, including its types, member access rules, and the usage of important keywords (this, super, final).
- Differentiate between method overloading and method overriding.
- Explore abstract classes and dynamic method dispatch for flexible program design.
- Collaborative Application of OOP Techniques: Apply object-oriented programming concepts to real-world projects through collaborative group work, fostering teamwork and practical application.

Week 3

Objective/topic:

- Students will learn the user date type, generic data type, generic methods and classes, and uses them,
- Students will understand the concept of Arrow function in Java, and the benefit of using them in modern programming and the idea of using clean code.
- Students will be familiar with extra data structures like set, hash set and hash Tables,
- Students will be able to work with exception handling, exception types, Usage of Try, Catch, Throw, Throws and finally keywords, Built-in Exceptions, Creating own Exception classes.
- Students will be familiar with graphical user interface and AWT control, class hierarchy, and user interface components- Labels, Button, Text Components, Check Box, Check Box Group, Choice, List Box, Panels – Scroll Pane, Menu, Scroll Bar.
- Working with Frame class, Color, Fonts and layout managers.
- Work with event handling, event sources, event Listeners, Event Delegation Model (EDM), Handling Mouse and Keyboard Events, Adapter classes, Inner classe

Learning Goal(s):

- Proficiency with Advanced Data Types and Generics: Achieve a deep understanding of user-defined and generic data types in Java, enabling students to write highly flexible, reusable, and type-safe code.
- Efficiency with Modern Java Practices: Master the use of arrow functions (lambda expressions) to simplify code and enhance readability, while promoting clean code practices in software development.

- Familiarity with Key Data Structures: Gain practical knowledge of advanced Java data structures such as Set, HashSet, and Hashtable, enabling students to choose the right data structure for various programming tasks.
- Exception Handling Mastery: Develop the ability to handle exceptions effectively, ensuring robust error management and program stability through the correct usage of Java's exception handling mechanisms.
- GUI Development Competence: Acquire the skills necessary to create user-friendly graphical user interfaces (GUIs) using Java AWT components, and implement effective layouts and event-driven programming for desktop applications.
- Understanding Event-Driven Programming: Gain a solid understanding of event handling in Java, including how to manage user inputs through event sources and listeners, and implement the Event Delegation Model (EDM) for handling various events.

Week 4

Objective/topic:

- The objectives of the topics are to equip learners with essential skills and knowledge in building robust, scalable, and efficient applications. Here's a breakdown of the objectives for each topic.
- Understand the foundational concepts of databases, including their types (relational vs non-relational), structure, and the importance of data management in software applications.
- Learn how to connect Java applications to relational databases, execute SQL queries, and manage database interactions using JDBC API.
- Understand advanced techniques for database interactions like using PreparedStatement for security and efficiency.
- Learn the benefits and implementation of connection pooling to manage database connections efficiently, reducing overhead and improving application performance.
- Get familiar with Object-Relational Mapping (ORM) frameworks like Hibernate or JPA, which simplify database interactions by mapping Java objects to database tables.
- Understand how to manage database transactions to ensure the consistency and integrity of data when multiple operations are executed.
- Learn how to build dynamic web applications using Servlets (for handling business logic) and JSP (for rendering HTML content).
- Understand how to use filters and listeners to extend the functionality of servlets, such as request/response interception (filters) and tracking session/application states (listeners).
- Master various session management techniques (cookies, HTTP sessions, URL rewriting) to maintain user state across multiple requests.
- Goal: Implement secure and efficient session management in web applications.
- Spring Framework Topics
- Understand the core concepts of the Spring framework, including its modular nature, and how it simplifies application development through features like Dependency Injection (DI) and Aspect-Oriented Programming (AOP).
- Build your first Spring-based application to understand basic concepts such as Beans, Dependency Injection, and Spring Configuration.
- Learn how to create and consume RESTful web services in a Spring-based application using Spring MVC and Spring Boot.
- Understand the basics of cloud platforms like AWS or Azure and how to deploy Spring-based applications in a cloud environment.

Learning Goal(s):

- Goal: Be able to perform CRUD operations from a Java application using JDBC.
- Implement complex queries and handle database operations in a more secure and optimized manner.
- Be able to configure and use a connection pool in Java applications.
- Use ORM tools to perform database operations without writing raw SQL queries
- Implement and manage transactions, ensuring ACID properties (Atomicity, Consistency, Isolation, Durability).

- Be able to develop web applications where the presentation layer (JSP) interacts with the business layer (Servlets).
- Implement filters for logging, security, or data modification, and listeners for session or application event tracking.
- Have a working environment for Spring development, enabling the creation of robust applications.
- Use Spring Data to manage persistence and database operations in a Spring-based application.
- Develop a simple Spring application to experience the framework in action.
- Understand how to interact with databases more efficiently using Spring Data, which simplifies CRUD operations through repository patterns.
- Set up and develop Java applications using the Spring framework.
- Build and expose APIs for other applications or clients to interact with your system.
- Deploy a Spring application to a cloud platform, making it accessible and scalable.

Assessment information

Your assessments will be weighted as follows:

- 20%: Participation
- 20 % Solving exercises in practical sessions
- 20%: Solving Exercises during Group Sessions
- 40%: Final Exam

Please note that all participants must attend at least 80% of their classes in order to be eligible to pass the course. Failure to attend 80% of classes without an adequate reason will result in an automatic fail.

Grading information

All participants of the TU Berlin Summer & Winter School will receive a grade under the German grading system. The following table provides an overview of the grading system and equivalent scores for international credit transfers:

Total mark	German grade	English description
Greater than or equal to 95	1,0	Excellent
Greater than or equal to 90	1,3	Very good
Greater than or equal to 85	1,7	Good
Greater than or equal to 80	2,0	Good
Greater than or equal to 75	2,3	Good
Greater than or equal to 70	2,7	Satisfactory
Greater than or equal to 65	3,0	Satisfactory
Greater than or equal to 60	3,3	Satisfactory
Greater than or equal to 55	3,7	Sufficient
Greater than or equal to 50	4,0	Sufficient
Less than 50	5,0	Fail

Credit Points

ECTS is a point system and European standard developed by the Commission of the European Community and refers to the European Credit Transfer System. The aim of the system is to provide common procedures and guarantee academic recognition of studies abroad. The credit system is based on student

workload. All lectures, homework and group assignments count towards the workload. One point is awarded for the equivalent of 25-30 hours of workload.

Allocation of workload and ECTS

Orientation + cultural program	15 hours
Lectures	61 hours
Practical	38 hours
Self-study/ Homework	35 hours
Final exam/ Project presentation	02 hours
Total	**150 hours**
25 hours ≙ 1 Credit	**⇒ 6 Credits**