Hochschule Offenburg
offenburg.university

## Semester Project Overview:

# Database Schema Design and Implementation

## Phase 1/2

WS 25/26 / Course RED / Prof. Dr.-Ing. J. Bausch

11. November 2025

# General Information

## Applicable to all Groups and tasks

As part of this course, each student is required to contribute to the development of a structured database project, culminating in both a written term paper and an individual presentation (see StuPo HA+RE guidelines). The project is divided into two phases, with initial submissions due on November 28th and final submissions due on January 16th.

**General Requirements:**
The focus of this project is the creation and manipulation of database structures through SQL or Python-based code. Students are expected to submit well-documented code, with ample comments and explanations to ensure clarity. Alternatively, a traditional written report may be provided. Each student must also prepare an individual presentation, discussing one aspect of their schema or database development, to be delivered at the end of each phase. This approach is intended to balance the workload evenly among participants.

# General Information - Phase 1

## Applicable to all Groups and tasks

**Part 1**

**Deadline: December 5th, 2025, 23:59h (submission portal closes on Moodle).**
Collaboration: Students will work in groups, with grading based on group performance. Each group member will also receive an individual assessment.

**Submission Requirements for Phase 1:**

Database Structure Creation: Submit an SQL script that initializes the database structure, including tables, relations, constraints, and views. This script should begin with a USE <database_name>; statement to specify the database context. You may utilize tools such as MySQL Workbench to assist in the model development.

Sample Data Generation: Include an SQL script that populates the schema with sample data, which will be used to demonstrate and test the functionality of the database.

Functionality Testing: Provide SQL scripts that test the database's constraints and functionalities, ensuring that the schema operates as expected.

Individual Presentations: Each group member must prepare an individual presentation that covers a specific component of the schema. This will be followed by a Q&A session to assess understanding and depth of knowledge.

**Guidelines for Individual Sessions:**

Each student will receive an individual grade, with a specific emphasis on the quality of their presentation and understanding of the schema component they discuss. The presentation should provide an in-depth explanation of the design choices, challenges encountered, and any optimizations applied.

**Development Environment:**

All tables, views, and procedures must be implemented within the group schema provided on the server at reddata.m.hs-offenburg.de. If additional test schemas are needed, students may request them or utilize a local database installation.

# Task 1: Vehicle Fleet-Management-System

## Group 1 - Scope of Term Paper

**Part 1**

**Interview:**
*Interviewer:* "We need a database to track data from our fleet of delivery vehicles. Can you help us design it?"
*Interviewee:* "Sure. What kind of data do you need to track?"
*Interviewer:* "We need to track the vehicle ID, make, model, year, current location, speed, fuel or charging level, and engine/e-motor temperature. We also need to track the driver's ID, name, and license number. And finally, we need to keep a log of all the trips each vehicle has made, including the start and end times and start address and destination address, distance traveled, and fuel consumption. "
*Interviewee:* "Are there any special functions, aggregations needed?"
*Interviewer:* "Yes, we want an efficiency analysis of all drivers and need the cost per km/mile in including fuel/power and maintenance. It would be nice to have a list of speed averages km/h mi/h per driver and type"

**Minimum Requirements:**
Different Types of Vehicles (cars, trucks, motorcycles)
Vehicle specifications (make, model, year, license plate)
Driver Information (driver ID, name, license number)
Real-time location tracking (latitude, longitude, timestamp)
Vehicle diagnostics (speed, fuel level, engine temperature, tire pressure, service status [available/on tour/ maintance/out of service/etc.])
Trip Logging (start/end time, distance, fuel/power consumption)
Maintenance records (date, type of service, cost)
Fuel Efficiency Analysis (average fuel consumption per vehicle type)
Driver Performance Monitoring (driving habits, adherence to speed limits)
Alert System (notifications for low fuel, engine issues, speeding)

Use tables, relations, constraints and views, functions

# Task 2: The Online Shop

## Group 2 - Scope of Term Paper

**Part 1**

**Interview:**
*Interviewer:* "We are launching an online shop and we need a database to manage our products, customers, and orders."
*Interviewee:* "Certainly! Could you please tell me more about the types of products you'll be offering, the customer information you need to collect, and how you'd like to manage orders?"
*Interviewer:* "We'll have various product categories like electronics, clothing, and books, each with attributes like name, description, customer and purchase price, and stock quantity. For customers, we need to store their name, email, shipping address, and billing information. Orders should track the purchased items, quantities, total amount, and order status."
*Interviewee:* "Are there any special functions, aggregation needed?"
*Interviewer:* "We have a minimum stock quantity for each item, if we could have an alarm list with 20% and 10% levels this would be helpful. Furtheron and we need a basic profit analysis on a monthly basis"

**Minimum Requirements:**
Product Catalogue (product ID, name, description, price, category, image, stock quantity)
Customer Accounts (customer ID, name, email, password, shipping address, billing address)
Order Management (order ID, customer ID, order date, order status, total amount)
Order Items (order ID, product ID, quantity, price)
Shopping Cart Functionality (store items temporarily before checkout)
Payment Processing (integration with payment gateways, storing transaction details)
Shipping and Delivery Tracking (shipping provider, tracking number, delivery status)
Product Reviews and Ratings (customer reviews, star ratings)
Wish List (customers can save items for later purchase)
Discount Coupons and Promotions (manage discount codes and promotional campaigns)

Use tables, relations, constraints and views, functions

# Task 3: University-Management-System

## Group 3 - Scope of Term Paper

**Part 1**

**Interview:**
*Interviewer:* "We need a database to manage student information, courses, enrollments, and grades at our university."
*Interviewee:* "Of course! Could you elaborate on what information you need to store for each student, course, enrollment, and grade?"
*Interviewer:* "For students, we need to store their student ID, name, address, and contact information. Courses should include course ID, name, department, credits, and prerequisites. Enrollments should link students with courses and track their enrollment status. Grades should be recorded for each student enrollment in a course."
*Interviewee:* "Are there any special functions, aggregation needed?"
*Interviewer:* "Of course, We want to have a list of the top 10 students in gradings and efficiency, means credits per semesters achieved"

**Minimum Requirements:**
Student Information (student ID, name, address, email, phone number, date of birth)
Course Information (course ID, name, department, credits, description, prerequisites, lecturer)
Enrollment Management (student ID, course ID, enrollment date, enrollment status)
Grading System (student ID, course ID, grade, grade date)
Faculty Information (faculty ID, name, department, email, phone number)
Course Scheduling (course ID, faculty ID, semester, year, class timings, classroom)
Attendance Tracking (student ID, course ID, date, attendance status)
Student Transcripts (historical record of courses taken and grades received)
Graduation Requirements (track student progress towards degree completion)
Academic Advising (faculty advisors assigned to students, advising notes)

Use tables, relations, constraints and views, functions

# Task 4: Canteen Ordering System

## Group 4 - Scope of Term Paper

**Part 1**

**Interview:**
*Interviewer:* "We are developing a system for students and employees to order meals in advance from our university's canteen. We need a database to manage the menu, orders, and user accounts."
*Interviewee:* "Absolutely! Could you tell me more about the types of meals you offer, the information you need to collect for user accounts, and how you want to handle order placement and fulfillment?"
*Interviewer:* "We offer various meal types categorized as breakfast, lunch, and dinner, each with options for normal, vegetarian, and vegan diets. User accounts should store their ID, name, email, and payment information. Orders need to track user ID, chosen meals, quantities, order date, and pickup time."
*Interviewee:* "Are there any special functions, aggregation needed?"
*Interviewer:* "Of course, for automated billing we need a price per meal and a monthly report of the meals of each user where we charge students with the indicated price and employees with a 10% surcharge. We need a monthly based revenue list and a simple feedback system would be nice"

**Minimum Requirements:**
User Accounts (user ID, name, email, password, role [student/employee], payment method)
Menu Management (meal ID, name, description, category [breakfast/lunch/dinner], type [normal/vegetarian/vegan], price, image, availability)
Order Placement (order ID, user ID, order date, pickup time, order status, total amount)
Order Items (order ID, meal ID, quantity)
Ingredient Inventory (ingredient ID, name, quantity, unit)
Meal Recipes (meal ID, ingredient ID, quantity)
Allergen Information (meal ID, allergen)
Nutritional Information (meal ID, calories, fat, protein, carbs)
Order Fulfillment (track order preparation status, assign pickup counters)
Feedback and Ratings (users can rate meals and provide feedback)

Use tables, relations, constraints and views, functions

# Task 5: Airline-Management-System

## Group 5 - Scope of Term Paper

**Part 1**

**Interviewer:** "We need a database to manage our airline operations, including flights, bookings, aircraft, and crew."
**Interviewee:** "Certainly! Could you tell me more about the information you need to track for flights, customers, and the airline staff involved?"
**Interviewer:** "We need to keep track of flight schedules, available aircraft, and the crew assigned to each flight. We also need to manage bookings, storing customer information and their seat assignments on flights. Each flight should have a departure and arrival airport, as well as scheduled times."
**Interviewee:** "Are there any specific reports or alerts that would be helpful?"
**Interviewer:** "Yes, we'd like to monitor seat availability on each flight and see if any flights are delayed. Additionally, monthly summaries of flight occupancy and customer feedback would be helpful."

**Minimum Requirements:**
Aircraft Management - Track different types of aircraft with details such as type, seating capacity, and availability.
Customer Information - Store customer details, including name, contact information, and frequent flyer number.
Booking Management - Track bookings by flight, customer, seat assignment, booking date, and status (confirmed, canceled).
Flight Schedule - Store details of flights, including flight number, aircraft ID, departure/arrival airport, scheduled times, and status (on time, delayed, canceled).
Crew Management - Track pilots and crew assigned to each flight, including their qualifications and availability.
Airport Information - Store airport details, such as airport code, name, and location.
Overbooking and Occupancy Monitoring - Provide alerts when flights reach maximum occupancy, and allow for overbooking management.
Customer Feedback and Ratings - Collect and store customer feedback for each flight, including star ratings and comments.
Monthly Reports - Generate reports on flight occupancy, delays, and customer satisfaction.
Alert System - Notifications for changes in flight status and crew availability.

Use tables, relations, constraints and views, functions