

<b>BLOCK NAME</b>	BASIC DATABASES
<b>BLOCK CODE</b>	CS-L2B4
<b>COURSE</b>	1
<b>LEVEL</b>	2
<b>CREDITS</b>	6
<b>CLASS HOURS</b>	60
<b>HOMEWORK</b>	90
<b>TOTAL HOURS</b>	150

**DESCRIPTION**

This block introduces the student to relational databases. We will face the challenge of creating a program that will be able to manage customer information for a business. This will include doing the corresponding data modelling, defining the Entity-Relationship model, creating databases and tables, and implementing all functionalities to access such DB using SQL.

**PRE-REQUISITES**

Programming skills are needed.  
**CS-L1B1, CS-L1B2**

**OBJECTIVES**

The goal is for students to be familiar with relational databases and their basic usage.

**SKILLS TO BE DEVELOPED**

- 1 - Data modelling.**
  - 1.1 - Understand data modelling.
  - 1.2 - Define a suitable data modelling description to solve a specific problem.
- 2 - Entity-Relationship model.**
  - 2.1 - Define a suitable ER model to solve a specific problem.
- 3 - Relational databases.**
  - 3.1 - Understand how relational databases work.
- 4 - Database management using engine tools.**
  - 4.1 - Create, edit and delete databases.
  - 4.2 - Create, edit and delete tables.
  - 4.3 - Create, edit and delete indexes at tables.
- 5 - Structured Query Language (SQL).**
  - 5.1 - Write SQL queries to create, edit and delete databases.
  - 5.2 - Write SQL queries to create, edit and delete tables.
  - 5.3 - Write SQL queries to create, edit and delete indexes at tables.
  - 5.4 - Write SQL queries to perform CRUD operations over table records.
- 6 - Database access from programs.**
  - 6.1 - Execute queries from programs.
  - 6.2 - Read query results from programs.

**SYLLABUS**

- 1 - Data modelling.
- 2 - Entity-Relationship model.
- 3 - Relational databases.
- 4 - Database management using engine tools.
- 5 - Structured Query Language (SQL).
- 6 - Database access from programs.

## METHODOLOGY

Resolution of practical activities supervised by the mentor. Compulsory attendance.

## DEDICATION AND EVALUATION

The student must pass the mandatory activities (challenges/projects) that are covered in the block.

Each challenge/project produces its own score and has been designed to cover certain block percentages.

Such score is 80% objective (the program that solves the challenge/project works without errors and producing the expected results) and 20% subjective (solution elegance, how clean the code is, documentation).

Block scores are finally calculated by prorating individual activities with respect to their block coverage percentages.