

# RELATIONAL DATABASE

Damiano SOMENZI

SCHOOL OF ECONOMICS AND MANAGEMENT

Advanced Computer Skills

`damiano.somenzi@unibz.it`



Week 1

## ① UNIVERSITY STUDENT EXAMINATION

## ② REQUIREMENTS ANALYSIS

## ③ CREATE THE DATA BASE

Design

Create

Query

## STUDENT EXAMINATION RECORD

- A University Study Programme comprises semester subjects, the examination set for each of these subjects must be passed
- Students enroll at the examination session for one or more subjects
- Enrolled students who take the examination pass or fail
- Students who pass the examination receive a mark
- The member of the examination commission fill and sign an official examination record
- In the “examination record” shall be registered student, subject, and examination data

## EXAMINATION DATA

A data base needs to be developed in order to speed-up the registrations, easily retrieve examination data and extract helpful information as well

- 1 A **Student** is identified by the studentID and private data (name, surname, etc.)
- 2 A student has a university **Career**: an enrollment date, a study programme, and a programme year (i.e. 2055 (StudentID), 10/10/2009, Bachelor in Design, 3)
- 3 A **Subject** is identified by the subject name, programme year, attendance semester, credits, etc.
- 4 Subject name is unique
- 5 **Study Programme** as a set of subjects
- 6 **Student, Subject and Examination** data are registered on the **Examination Record**

## TASKS

- 1 **DESIGN** the architecture
- 2 **CREATE** the Data Base
- 3 **QUERY** the Data Base

## ① UNIVERSITY STUDENT EXAMINATION

## ② REQUIREMENTS ANALYSIS

## ③ CREATE THE DATA BASE

Design

Create

Query

## STUDENT TABLE

Field	Notes
Student-ID	Sequence of 4 digits ( <b>Primary Key</b> ) Search improvement → <b>Index</b>
First Name	<b>required</b>
Last Name	<b>required</b>
Address	It should be split into three fields [Street, Zip Code, City/Town] <b>required</b>
Phone	<b>required</b>
Date of Birth	<b>required</b>

## CAREER TABLE

Field	Notes
Programme Name	{Bachelor in Economics, etc. } <b>required</b>
Enrollment Date	<b>required</b>
Programme Year	{1, 2, 3 } <b>required</b>
Student-ID	<b>required</b>

## COURSE SUBJECT TABLE

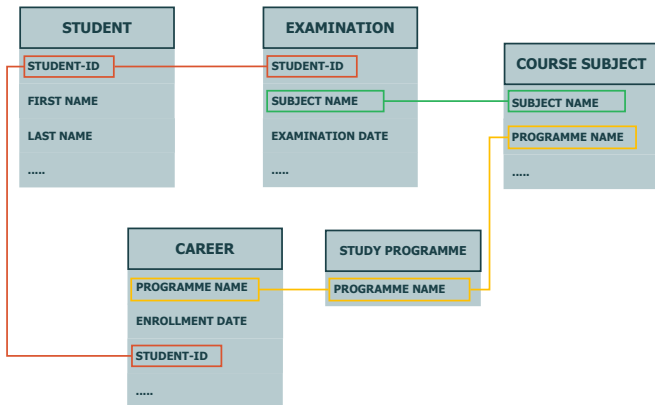
Field	Notes
Subject Name	up to 50 characters ( <b>Primary Key</b> )
Programme Name	{Bachelor in Economics, etc. } <b>required</b>
Teaching Year	{1, 2, 3 } <b>required</b>
Attendance Semester	{1 <sup>st</sup> , 2 <sup>nd</sup> } <b>required</b>
Credits	{5, 10 } <b>required</b>
Language	{English, German, Italian} <b>required</b>
Type	{Compulsory, Optional} <b>required</b>

## EXAMINATION TABLE [JUNCTION TABLE]

Field	Notes
Student-ID	Identifies a student in <b>Student</b> table ( <b>Foreign Key</b> of the relationship)
Subject Name	Identifies a course in <b>Course Subject</b> table ( <b>Foreign Key</b> of the relationship)
Examination Date	Less than or equal to <i>Today</i> <b>Required</b> and <b>Indexed</b>
	<i>Student-ID + Subject Name + Date</i> uniquely identify an examination ( <b>Primary Key</b> )
Type	{oral, written, written + oral} <b>required</b>
Taken	{yes, no}, default <b>no</b>
Passed	{yes, no}, default <b>no</b>
Mark	{18 .. 30}
Laude	{yes, no}, default <b>no</b>

## STUDY PROGRAMME

Field	Notes
Programme Name	List of values <b>Primary Key</b>



## ① UNIVERSITY STUDENT EXAMINATION

## ② REQUIREMENTS ANALYSIS

## ③ CREATE THE DATA BASE

Design

**Create**

Query

## DATA DICTIONARY

<b>Table</b>	<b>Description</b>	<b>Identifier</b>
Student	Students enrolled at the university	Student-ID
Career	Students university career	-
Course Subject	University teachings belonging to a Study Program	Subject Name
Examination	Students examination record	Student-ID + Subject Name + Examination Date
Study Programme	University study programmes	Programme Name

## DATA DICTIONARY

<b>Relationship</b>	<b>Type</b>
Student ↔ Examination	<i>one to many</i>
Course Subject ↔ Examination	<i>one to many</i>
Career ↔ Student	<i>many to one</i>
Study Programme ↔ Career	<i>one to many</i>
Study Programme ↔ Course Subject	<i>one to many</i>

## CONSTRAINTS

- Many fields **require** a value
- Few fields should be **indexed**
- *Student-ID* shall be a 4 digits code
- *Subject Name* should be a sequence up to 50 characters
- *Examination Date* shall be coherent with the registration date
- *Mark* is a number between 18 and 30
- If *Taken* is “no” then *Passed* and *Mark* should be **NULL**
- If *Taken* is “yes” and *Passed* is “no” then *Mark* should be **NULL**

## FORM

We create a friendly tool to help user to insert a new **Examination Record**

## ① UNIVERSITY STUDENT EXAMINATION

## ② REQUIREMENTS ANALYSIS

## ③ CREATE THE DATA BASE

Design

Create

Query

## QUERY: FILTER DATA

- A List for a given student (identified by the Student-ID) passed examinations during his university career
- B List students enrolled at the *Bachelor in Economics and Management* in the current academic year. Return data ordered by “programme year” and “last name”

## QUERY: GROUPING AND RETURN VALUES

- C For each course subject compute the number of passed examinations during the academic year 2008/2009
- D For each student compute the sum of credits obtained during his university career
- E For each course subject compute the number of examinations marked with 30 or 30 cum laude