

# RELATIONAL DATABASE

Damiano SOMENZI

SCHOOL OF ECONOMICS AND MANAGEMENT

Advanced Computer Skills

`damiano.somenzi@unibz.it`



Week 2

## ① UNIVERSITY STUDENT EXAMINATION

Query  
Report

## ② SOCIAL NETWORK

## ③ REQUIREMENTS ANALYSIS

## ④ CREATE THE DATA BASE

Design  
Create

## ① UNIVERSITY STUDENT EXAMINATION

Query

Report

## ② SOCIAL NETWORK

## ③ REQUIREMENTS ANALYSIS

## ④ CREATE THE DATA BASE

Design

Create

## QUERY: GROUPING AND RETURN VALUES

- 1 For each course subject how many examinations have been taken (if any)? For those ones how many have been passed and failed?

*Hint: use two sub-queries*

- 2 For each course subject what is the highest, the lowest, and the average mark?

## ① UNIVERSITY STUDENT EXAMINATION

Query

Report

## ② SOCIAL NETWORK

## ③ REQUIREMENTS ANALYSIS

## ④ CREATE THE DATA BASE

Design

Create

## REPORT: SUMMARY DATA

Every semester examinations are reported in a summary document

- Report examinations by subject and ordered by date
- Report should be available as RTF document

*The semester could be from 01.01.2009 to 30.06.2009*

## SOCIAL NETWORK

- “Social Networks” are a common way of interaction among people
- People can contact new and old friends and share with them common interests: opinions, web–links, pictures, video, etc.

## SOCIAL NETWORK ON A WEB SITE

We aim to design a database in order to support the web site of a social network

- People participating at the social network must be registered, therefore a user needs a personal account (user and password)
- Users can search people for a friendship
- Users write comments, send messages to friends, and publish pictures
- Users read comments from friends, receive messages and look at pictures
- In few years a user register many friendships
- In few years a user stores a lot of messages and pictures

## SOCIAL NETWORK ON A WEB SITE

A data base needs to be developed in order to register users, friendships, and to store messages and pictures

- 1 A **USER** is identified by a User-ID and private data (name, surname, email)
- 2 A **FRIENDSHIP** is identified by two User-IDs, and a registration date
- 3 A **MESSAGE** is identified by the text, author's User-ID, and the date/time
- 4 A **PICTURE** is identified by the image, User\_ID who has submitted, and the date/time

## TASKS

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

## ① UNIVERSITY STUDENT EXAMINATION

Query

Report

## ② SOCIAL NETWORK

## ③ REQUIREMENTS ANALYSIS

## ④ CREATE THE DATA BASE

Design

Create

## USER TABLE

Field	Notes
UserID	Up to 50 characters (valid email) ( <b>Primary Key</b> ) Search improvement → <b>Index</b>
Password	a valid password ends with a digit + "!" <b>required</b>
First Name	<b>required</b>
Last Name	<b>required</b>
Date of Birth	

## MESSAGE TABLE

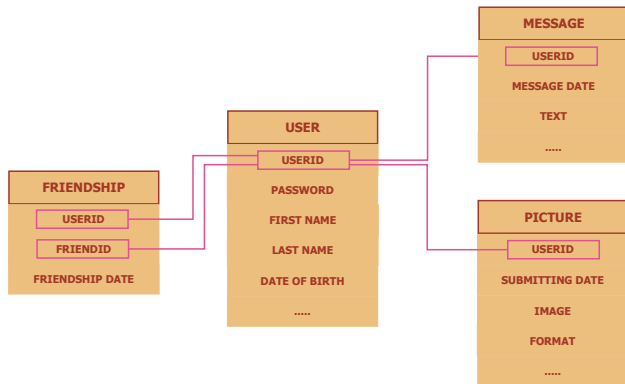
Field	Notes
UserID	
Message Date	date ( <b>required</b> ) + time ( <b>required</b> ) Search improvement → <b>Index</b>
	<i>UserID + Message Date</i> ( <b>Primary Key</b> )
Message	typically a long text <b>required</b>

## PICTURE TABLE

Field	Notes
UserID	
Submitting Date	date ( <b>required</b> ) + time ( <b>required</b> ) Search improvement → <b>Index</b>
	<i>UserID + Submitting Date (Primary Key)</i>
Picture	Object <b>required</b>
Format	{jpeg, eps, png, ....}

## FRIENDSHIP

Field	Notes
UserID	<b>required</b>
FriendID	<b>required</b>
	<i>UserID + FriendID (Primary Key)</i>
Friendship Date	date ( <b>required</b> ) + time ( <b>required</b> )



## ① UNIVERSITY STUDENT EXAMINATION

Query  
Report

## ② SOCIAL NETWORK

## ③ REQUIREMENTS ANALYSIS

## ④ CREATE THE DATA BASE

Design  
Create

## DATA DICTIONARY

<b>Table</b>	<b>Description</b>	<b>Identifier</b>
User	Records people registered at the social network	UserID
Message	Messages repository	UserID + Message Date
Picture	Pictures repository	UserID + Submitting Date
Friendship	Registers friendships	UserID + FriendID

## DATA DICTIONARY

<b>Relationship</b>	<b>Type</b>
User ↔ Message	one ↔ many
User ↔ Picture	one ↔ many
User (user) ↔ Friendship	one ↔ many
User (friend) ↔ Friendship	one ↔ many

## CONSTRAINTS

- Many fields **require** a value
- Few fields should be **indexed**
- *UserID* shall be a valid email
- *Password* must be a sequence of any character and ending with at least one digit [0 - 9] and the special character “!”
- Few fields mandatory require date plus time

## FORM

We create a friendly tool to help user to register new **friendships**