

# UVG - CC3057

# Bases de datos

Clase 1 – Introducción

El mundo es cada vez más  
conducido por los **datos**



La información es poder.  
(Francis Bacon)

# Ejemplos



¿Dónde se encuentra la información?

¿Bases de datos?

¿DBMS?

# Esta clase tiene por objetivos:

*Diseñar una base de datos **normalizada** que contenga la información **organizada** de manera adecuada y de **fácil acceso***

*Utilizar **SQL** como lenguaje de consulta para gestionar información almacenada en bases de datos*

*Diseñar y construir **sistemas de aplicación** de bases de datos*

*Estudiar cómo funcionan los sistemas gestores de bases de datos y los principios en que se fundamentan*

Esta clase tiene **no** por objetivos:

Cómo ser un DBA

Como *tunear* el desempeño de un SQL Server 2014



# Características de las bases de datos

Escalables

Accesibles

Persistentes

Eficientes

Seguras

Confiables

Multi usuario

# ¿Qué preguntas buscamos abordar?

¿Cómo podemos **almacenar y organizar** grandes cantidades de información?

*Construyendo herramientas y **estructuras de datos** para proveer e **indexar** eficientemente los datos*

¿Cómo podemos consultar (*query*) *eficientemente* la información?

Definiendo **lenguajes de alto nivel** que se traduzcan en operaciones sobre las estructuras de datos

¿Cómo podemos actualizar los datos de manera confiable?

Manejando accesos **concurrentes** a medida que los datos se leen y escriben

# ¿Qué aplicaciones tiene el estudio de bases de datos?

## Aplicaciones de software

*Apps móviles, cloud based applications, machine learning, big data*

Por medio de lenguajes de programación y *frameworks*

## Data analytics

Inteligencia de negocios, data science, modelamiento predictivo

# Acerca de

Ing. Héctor Hurtarte

[hahurtarte@uvg.edu.gt](mailto:hahurtarte@uvg.edu.gt)

Catedrático horario

Ingeniero de software (CCSS @ UVG 2012)

# Acerca de

A-113

Martes: 7:00 a 9:30

Viernes: 8:40 a 11:25

Actividad	Cantidad	Puntos	Total
Laboratorios	15	2	30
Exámenes	2	10	20
Proyecto #1	1	20	20
Proyecto #2	1	15	15
Proyecto #3	1	15	15
<b>TOTAL</b>			<b>100</b>

Acerca de

*An Introduction to Database Systems*, 8a Edición, C. J. Date

*Database Systems - The complete book*, 2a Edición , J. Ullman

# El modelo relacional

# Entonces, ¿qué es un DBMS?

Es un sistema de *software* diseñado para almacenar  
y manejar bases de datos



# ¿Y qué es una base de datos?

Una (potencialmente gran) colección de datos ingresados

Que se usa para **modelar** aspectos del mundo real

Que maneja **entidades y relaciones**

# Por ejemplo

Un sistema manejador de contenidos (CMS):

- Estudiantes
- Cursos
- Profesores



*Entidades*

- Quién **estudia** qué
- Quién **imparte** qué



*Relaciones*

# Modelos de datos

- Un **modelo de datos** es una colección de conceptos utilizados para describir datos
  - Modelo jerárquico
  - Modelo orientado a objetos
  - El modelo relacional es el más usado actualmente (?)
    - Se fundamenta en el concepto de una *relación* (mundamente conocido como *tabla*)
- Un **esquema** (*schema*) es una descripción de una colección particular de datos, usando el **modelo de datos** dado
  - Cada *relación* en el modelo relacional tiene un esquema que define tipos de datos

# Modelando el CMS

- *Esquema lógico*
  - Estudiantes(*e\_id: string, nom: string, pro: float*)
  - Cursos(*c\_id: string, nom: string, cred: int*)
  - Asignaciones(*sid: string, cid: string, nota: float*)

e_id	nom	pro
101	Bob	83.2
123	Mary	93.8

Estudiantes

## Relaciones

e_id	c_id	nota
123	564	85

Asignaciones

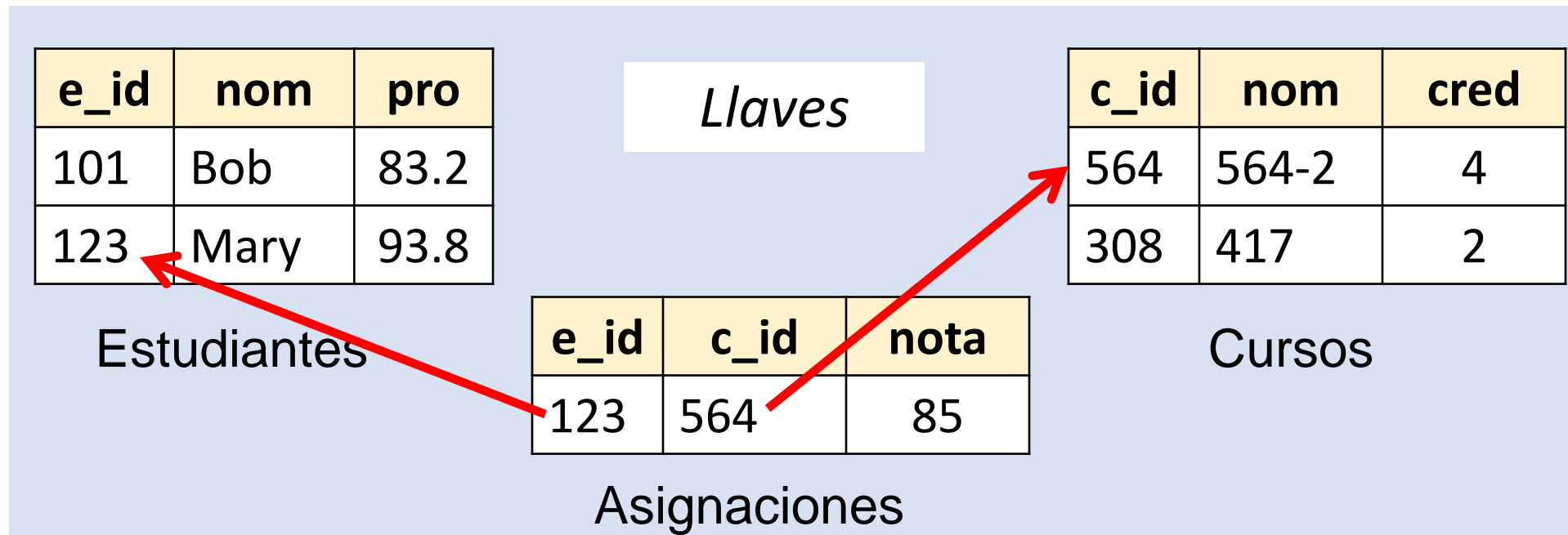
c_id	nom	cred
564	564-2	4
308	417	2

Cursos

# Modelando el CMS

- *Esquema lógico*

- Estudiantes(*e\_id: string, nom: string, pro: float*)
- Cursos(*c\_id: string, nom: string, cred: int*)
- Asignaciones(*sid: string, cid: string, nota: float*)



# Otros esquemas

- *Esquema físico*: describe la organización de datos

- *Relaciones* como archivos individuales
- Archivo único con todos los datos



*Implementador  
de DBMS*

*DBA*

- *Esquema lógico*

*Diseñador de BD*

- *Esquema externo* (vistas)

- *Informacion\_curso* (*c\_id*: *string*, *asignacion*: *integer*)
- Se deriva de otras *relaciones*



*Desarrolador de  
aplicaciones*

# Independencia de datos

Las aplicaciones no deben preocuparse de *cómo está estructurada y almacenada la data*

**Independencia lógica:** Protección de cambios en la estructura lógica de los datos

*¿Tengo que reescribir el app para crear un nuevo campo?*

**Independencia física:**

Protección de cambios en la estructura *física* de los datos

*¿En qué segmentos del disco estará guardada esta tabla?*



Talk is cheap. Show me the code.

(Linus Torvalds)



# Jupyter

Instalar Python 2.7

```
python -m pip install jupyter
```

```
python -m pip install ipython-sql
```

```
jupyter notebook
```