

Pontificia Universidad Católica de Chile  
Escuela de Ingeniería  
Departamento de Ciencia de la Computación



# IIC2115 - Programación como Herramienta para la Ingeniería

Bases de datos relacionales

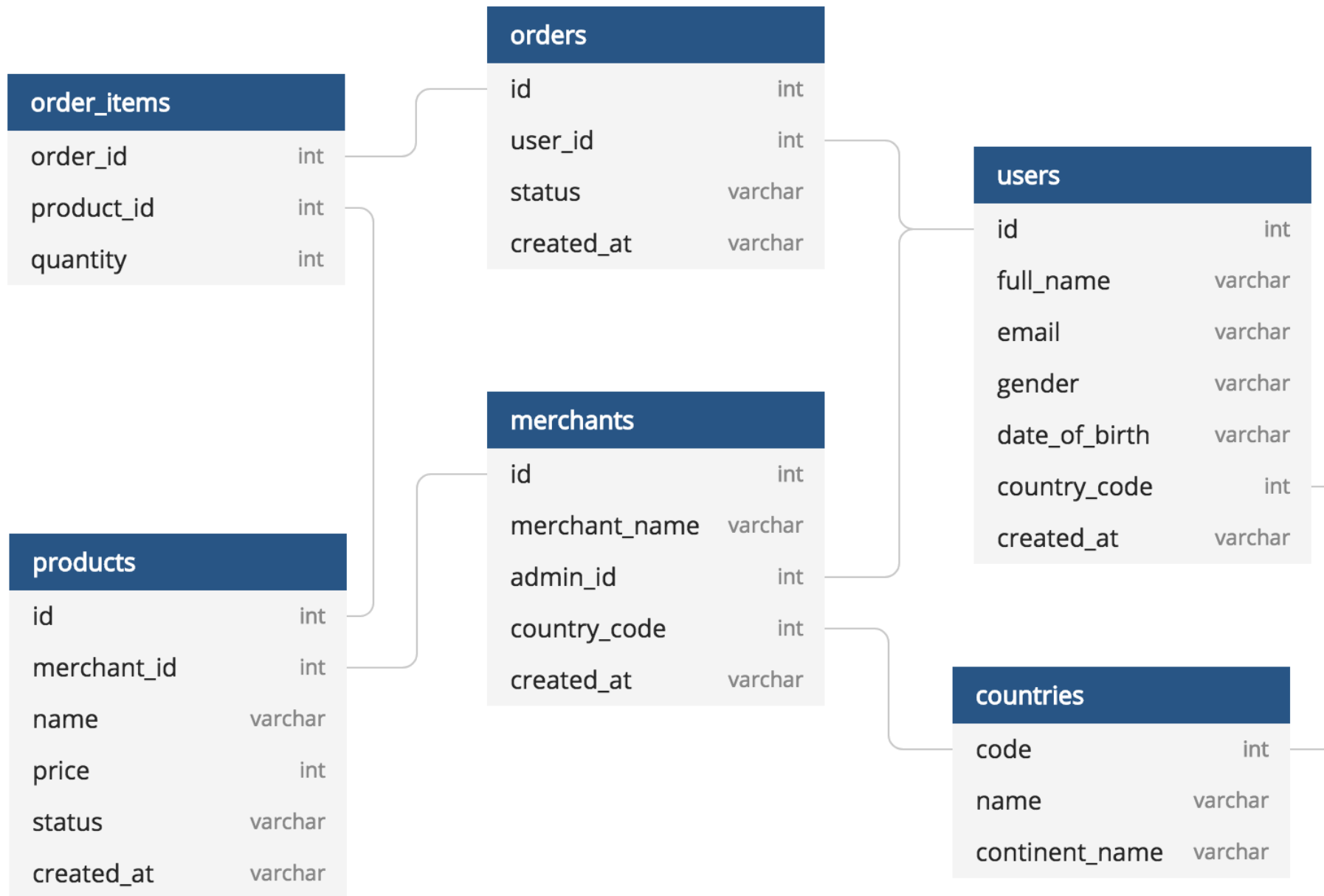
Profesor: Hans Löbel

## ¿Qué es una base de datos?

1. Corresponde un conjunto de datos de un mismo contexto y almacenados bajo cierta lógica/estructura e indexados para su posterior uso eficiente.
2. En el caso de una base relacional, es una colección de una o más relaciones, donde cada relación es una tabla con filas y columnas.

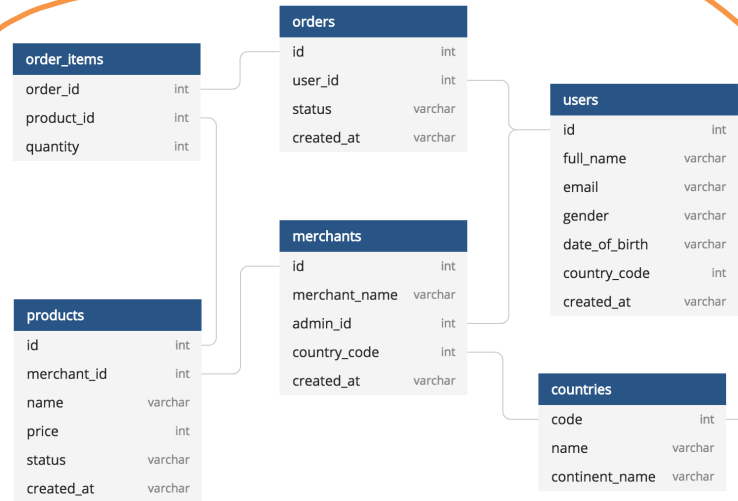


**RELATIONAL DATABASE**





Base de datos  
(Database)



Esquema (Schema)

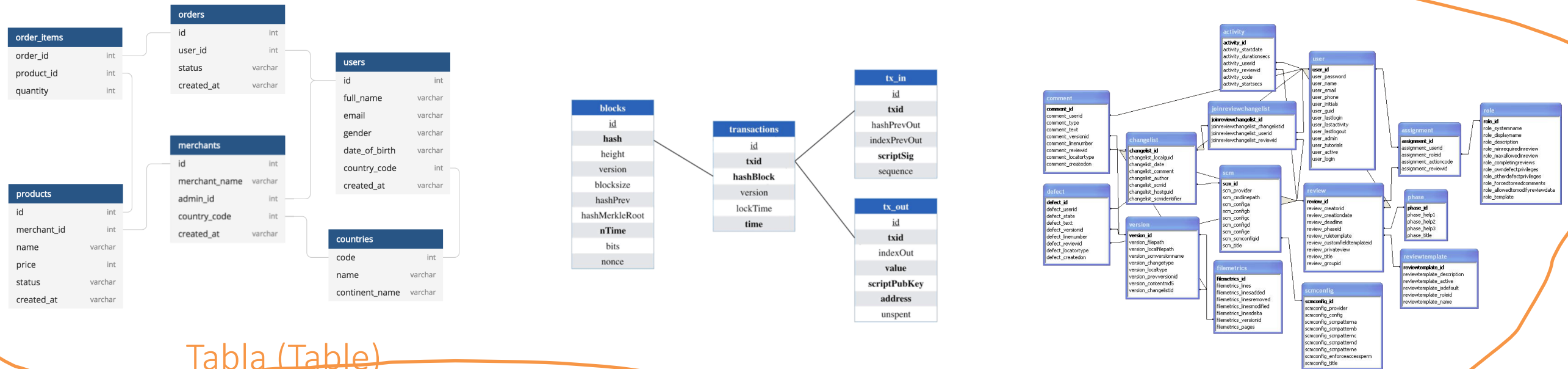


Tabla (Table)

# Tabla

Columna: Guarda un tipo específico de datos

CHAR(N)

VARCHAR(N)

INTEGER

REAL

...

Fila: Corresponde  
a un registro o  
instancia

Emp No	Name	Age	Department	Salary
001	Alex S	26	Store	5000
002	Golith K	32	Marketing	5600
003	Rabin R	31	Marketing	5600
004	Jons	26	Security	5100

Empleados (Emp No: STRING, Name: STRING, Age: INTEGER, Department: STRING, Salary: REAL )

Llaves primarias y secundarias  
(primary keys y foreign keys)

Emp No	Name	Age	Department_id	Salary
001	Alex S	26	1	\$000
002	Golith K	32	2	\$600
003	Rabin R	31	2	\$600
004	Jons	26	3	\$100

PK

FK

Id	Department
1	Store
2	Marketing
3	Security

PK

No hay FK

# Structured Query Language (SQL)

- Language de definición de datos (DDL)
  - Creación
  - Inserción
  - Eliminación
  - Modificación de definiciones de tablas.

\*Las restricciones de integridad se pueden definir en tablas, ya sea cuando se crea la tabla o posteriormente.

- Lenguaje de manipulación de datos (DML)
  - Consultas



## Creación

`CREATE TABLE [IF NOT EXISTS] table_name (column_1 data_type, column_2 data_type, ...)`

Emp No	Name	Age	Department	Salary
001	Alex S	26	Store	5000
002	Golith K	32	Marketing	5600
003	Rabin R	31	Marketing	5600
004	Jons	26	Security	5100

`CREATE TABLE Empleados (Emp_No CHAR(3), Name VARCHAR(20), Age INTEGER, Department VARCHAR(10), Salary REAL)`



## Inserción

**INSERT INTO** table\_name (column1,column2 ,..) **VALUES**( value1, value2 ,...)

Emp No	Name	Age	Department	Salary
001	Alex S	26	Store	5000
002	Golith K	32	Marketing	5600
003	Rabin R	31	Marketing	5600

**INSERT INTO** Empleados (Emp\_No, Name, Age, Department, Salary) **VALUES** ('004', 'Jons', 26, 'Security', 5100)

## Modificación

**UPDATE** table\_name **SET** column\_1 = new\_value\_1, column\_2 = new\_value\_2  
**WHERE** search\_condition

Emp No	Name	Age	Department	Salary
001	Alex S	26	Store	5000
002	Golith K	32	Marketing	5600
003	Rabin R	31	Marketing	5600
004	Jons	26	Marketing	5100

**UPDATE** Empleados E **SET** E.Department = 'Marketing' **WHERE** E.Emp\_No = '004'

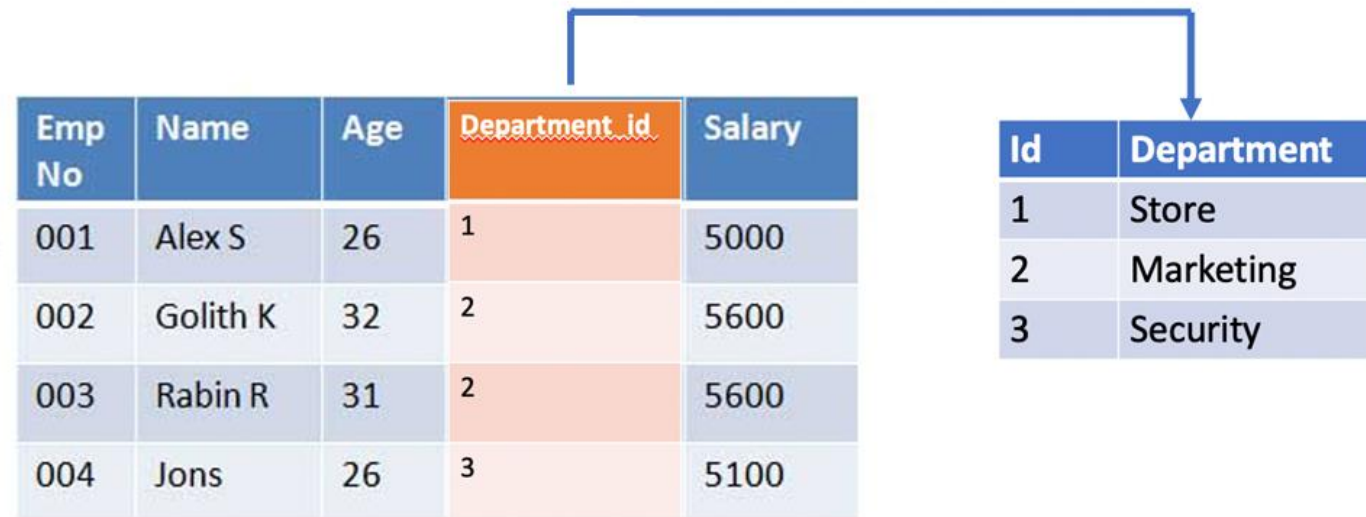
## Eliminación

**DELETE FROM** table\_name **WHERE** search\_condition;

Emp No	Name	Age	Department	Salary
001	Alex S	26	Store	5000
002	Golith K	32	Marketing	5600
003	Rabin R	31	Marketing	5600

**DELETE FROM** Empleados E **WHERE** E.Emp\_No = '004'

## Creación de tablas con *Primary Key* y *Foreign Key*



```
CREATE TABLE Departments (Id INTEGER, Department VARCHAR(20), PRIMARY KEY(Id))
```

```
CREATE TABLE Empleados (Emp_No CHAR(3), Name VARCHAR(20), Age INTEGER,  
Department_id INTEGER, Salary REAL, PRIMARY KEY(Emp_No), FOREIGN KEY (Department_id) REFERENCES  
Departments.Id)
```

## Uso en Python: DDL

```
import sqlite3
```

```
connection = sqlite3.connect('ejemplo.db')  
cursor = connection.cursor()
```

```
sqlStatement = 'CREATE TABLE Empleados (Emp_No CHAR(3), Name VARCHAR(20), Age INTEGER, Department  
VARCHAR(10), Salary REAL)'  
cursor.execute(sqlStatement)
```

```
Sq12 = 'INSERT INTO Empleados (Emp_No, Name, Age, Department, Salary) VALUES ('004', 'Jons', 26, 'Security', 5100)'
```

```
cursor.execute(Sq12)
```

```
connection.commit()  
connection.close()
```

## Manejo de errores

Al desarrollar este capítulo, se encontrarán dos tipos de errores:

- Errores de Python (de los que ya están familiarizados)
- Errores de la sintaxis de la base de datos (SQL)

CONSEJO: Pueden testear sus consultas directamente en la base de datos (p.ej., con SQLiteStudio) y luego utilizarla en Python

Pontificia Universidad Católica de Chile  
Escuela de Ingeniería  
Departamento de Ciencia de la Computación



# IIC2115 - Programación como Herramienta para la Ingeniería

Bases de datos relacionales

Profesor: Hans Löbel