UDE
UNIVERSIDAD DE LA EMPRESA

SQL – Structured Query Language

- Licenciatura e Ingeniería en Sistemas
- 2do. año

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

UDE
UNIVERSIDAD DE LA EMPRESA

Definición

- SQL = Structured Query Language
  - Conjunto de facilidades para definir, manipular y controlar los datos en una base de datos relacional.
- En SQL estándar se utilizan ambientes (environments), cada ambiente se constituye por una única base de datos (database)

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

UDE
UNIVERSIDAD DE LA EMPRESA

Definición

- Base de Datos: conjunto de todos los datos definidos por todos los esquemas en dicho ambiente.
  - Esquema: conjunto de definiciones de tablas, un conjunto de definiciones de vistas, y un conjunto de definiciones de privilegios.

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

UDE
UNIVERSIDAD DE LA EMPRESA

Tablas

- Datos vistos x usuario como colección de tablas con nombre (named tables): Nombre único
  - C/tabla con nombre puede ser tabla base (base table) o vista (view).
- Tabla base: tabla "real" autónoma (es decir, no definida en términos de otras tablas)
- Vista: tabla "virtual“; es una tabla con nombre, derivada de otras tablas.
- Creación mediante comandos CREATE TABLE y CREATE VIEW.
  - DROP TABLE y DROP VIEW para destruir tablas y vistas respectivamente.

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Tablas

- Tabla se compone de fila de encabezados de columna, junto a cero o más renglones de valores de datos.
  - La fila de encabezado especifica una o más columnas con nombre y tipo de datos.
  - En una misma tabla no puede haber dos columnas con el mismo nombre.
  - Cada renglón de datos contiene exactamente un valor por cada columna especificada en el renglón de encabezado.
  - Todos los valores en una columna son del mismo tipo (del tipo asociado a la columna en la columna de encabezado)
  - La cantidad de renglones de datos de una tabla se denomina **cardinalidad**. El numero de columnas de una tabla se denomina **grado**
  - El valor nulo (null value) es un valor especial que representa "valor desconocido“ y va a ser un valor permitido en las columnas que así lo permitan en su definición

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

SENTENCIAS DE MANIPULACION DE DATOS

- Sentencias del lenguaje SQL, que componen el DML.
  - Select
  - Insert
  - Delete
  - Update

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select

SELECT A_i op A_l,...,f(t.A_j),...,r_i.A_k
FROM ...,r_i,...,r_j t,...
[WHERE C_a]
[GROUP BY A_s,...,t.A_r,...,r_i.A_t]
[HAVING C_b]
[ORDER BY A_n,...,t.A_m,...,r_i.A_o]

- **A_i**: nombre de atributos
- **r_i**: nombres de tablas
- **C_i**: condiciones
- **op:** operadores aritméticos
- **f:** *funciones agregadas*

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select

SELECT A_i op A_l,...,f(t.A_j),...,r_i.A_k
FROM ...,r_i,...,r_j t,...
[WHERE C_a]
[GROUP BY A_s,...,t.A_r,...,r_i.A_t]
[HAVING C_b]
[ORDER BY A_n,...,t.A_m,...,r_i.A_o]

- "**SELECT**" → para consultar a la base de datos.
- Varias cláusulas que pueden emplearse en esta sentencia.
- 2 son obligatorias: **SELECT** y **FROM**.
  - Si se utilizan las otras cláusulas, deben aparecer en el orden en que se muestran en la sintaxis.

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: EJEMPLO

- Listar nombre, apellido y teléfono de cada cliente.

  SELECT Nombre, Apellido, Telefono

  FROM Clientes

Veremos como resultado todos los clientes por su nombre, apellido y teléfono que hay en esa instancia de tabla (Clientes).

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: EJEMPLO

Para listar todas las columnas de una tabla, usar * junto al **SELECT**.

```
SELECT *
FROM Stock;
```

o también:

```
SELECT Stock.*
FROM Stock;
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: EJEMPLO

SELECT DISTINCT NrClient

FROM Ordenes;

Listar los números de clientes que tienen órdenes pedidas. Los números que se repiten no deben aparecer en el resultado.

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

UDE
UNIVERSIDAD DE LA EMPRESA

Select: Order by

- Para que los resultados aparezcan ordenados.
  - El orden (por defecto) es ASCendente.
  - Para ordenar DESCendentemente se usa la palabra DESC a continuación de la columna correspondiente.
    - Un valor NULL es tratado como menor que un valor no nulo.
  - Las columnas mencionadas en el ORDER BY deben aparecer en la cláusula **SELECT**.

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: Order by / EJ

Listar los artículos del stock, ordenados por descripción y precio unitario, en forma ascendente.

```
SELECT *
FROM Stock
ORDER BY Descrip, PrUnidad
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: Order by / EJ

Listar los artículos en stock, ordenando en forma descendente por precio unitario.

```
SELECT *
FROM Stock
ORDER BY PrUnidad DESC
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

UDE
UNIVERSIDAD DE LA EMPRESA

Select: WHERE

- Para especificar criterio de búsqueda de datos y condiciones para vincular datos de diferentes tablas (join).
- Los *criterios de búsqueda* pueden ser:
  - Exclusión de valores: **!= <>**
  - Rango de valores: **BETWEEN**
  - Subconjunto de valores: **IN**
  - Valor nulo: **IS [NOT] NULL**
  - Expresiones regulares en strings
  - **WHERE** con exclusión de valores

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: WHERE / EJ

Listar números de clientes, compañía y estado, de los clientes que no son del estado CA (California).

```
SELECT NrClient, Compania, Estado
FROM Clientes
WHERE Estado <> "CA";
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: WHERE / EJ

- **WHERE con búsqueda de un rango de valores**

Listar los números de stock, códigos de fabricantes, descripciones y precios unitarios de los artículos en stock, cuyos precios unitarios están entre $10.00 y $30.00 inclusive.

```
SELECT NrStock, CodFabr, Descrip, PrUnidad
FROM   Stock
WHERE  PrUnidad >= 10.00 AND
       PrUnidad <= 30.00;
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: WHERE / EJ

- **WHERE con búsqueda de un rango de valores**

Listar los números de stock, códigos de fabricantes, descripciones y precios unitarios de los artículos en stock, cuyos precios unitarios están entre $10.00 y $30.00 inclusive.

```
SELECT NrStock, CodFabr, Descrip, PrUnidad
FROM   Stock
WHERE  PrUnidad BETWEEN 10.00 AND 30.00;
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: WHERE / EJ

- **WHERE con subconjunto de valores**

Listar los apellidos, ciudades, estados y teléfonos, de los clientes del estado CA (California) o del estado NV (Nevada). El resultado se desea ordenado por apellidos.

SELECT Apellido, Ciudad, Estado, Telefono
FROM Clientes
WHERE Estado = "CA" **OR** Estado = "NV"
ORDER BY Apellido;

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: WHERE / EJ

O sino con IN -->

```
SELECT Apellido, Ciudad, Estado, Telefono
FROM Clientes
WHERE Estado IN ("CA", "NV")
ORDER BY Apellido;
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: WHERE / EJ

- **WHERE con valor nulo (NULL)**

Listar los apellidos, ciudades y teléfonos, de los clientes del estado CA (California) que tienen valor desconocido en su segunda dirección. El resultado se desea ordenado por apellidos.

SELECT Apellido, Ciudad, Telefono
FROM Clientes
WHERE Estado = "CA" AND Direcc2 IS NULL
ORDER BY Apellido;

Observaciones:

El valor NULL se considera "valor desconocido". El valor 0 y el valor "" **no** se consideran valores nulos y son **distintos** del valor NULL.

El valor NULL es insertado cuando al agregar un renglón a una tabla, se dejan columnas sin datos, p.ej.

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: WHERE / EJ

- **WHERE con expresiones regulares**

Una *expresión regular* es un patrón de búsqueda de strings. Las expresiones regulares se especifican mediante las cláusulas **LIKE** y **MATCHES**.

[NOT] LIKE:

% representa una secuencia de caracteres

_ representa un único caracter

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: WHERE / EJ

Listar artículos en stock cuya descripción empiece con los caracteres “manya"

```
SELECT *
FROM Stock
WHERE Descrip LIKE “manya%“;
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: WHERE / EJ

Listar datos de los fabricantes cuyos códigos tengan una letra "R" en la segunda letra del código.

```
SELECT *
FROM Fabricnt
WHERE CodFabr LIKE "_R_"
;
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Expresiones aritméticas

- Una *expresión aritmética* puede usarse en la cláusula **SELECT** o en la cláusula **WHERE**.

- Operadores aritméticos:

  * multiplicación

  / división

  + adición

  - sustracción

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Expresiones aritméticas

Listar los datos de los artículos en stock, junto con el precio unitario multiplicado por 1.05.

```
SELECT Stock.*, PrUnidad*1.05
FROM Stock;
```

Listar ciertos datos de los artículos en stock con su cantidad por su precio unitario

```
SELECT   NrOrden, Artics.NrStock, Artics.CodFabr, Cantidad,
         PrUnidad, Cantidad*PrUnidad, Precio

FROM     Artics, Stock
WHERE    Artics.NrStock = Stock.NrStock AND
         Artics.CodFabr = Stock.CodFabr
ORDER BY          NrOrden;
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Funciones agregadas

- Se usan para realizar conteos, sumas, promedios, hallar máximos y mínimos, en columnas con datos numéricos.
  - **COUNT(*) / COUNT(DISTINCT columna)**
  - **SUM(x) / SUM(DISTINCT columna)**
  - **AVG(x) / AVG(DISTINCT columna)**
  - **MAX(x) / MIN(x)**
- *x*: columna o expresión aritmética
- NOTA: Si el conjunto que está en la cláusula **SELECT** incluye alguna función agregada, no se pueden incluir otras columnas a menos que se use la cláusula **GROUP BY**.

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

UDE
UNIVERSIDAD DE LA EMPRESA

Select: Count

- Cuenta todos los renglones que satisfacen la condición de la cláusula **WHERE**.
- Si se desean contar valores diferentes de alguna columna se usa la cláusula **COUNT(DISTINCT columna)**.
- Si todos los valores de la columna son **NULL**, da 0.
- Si hay valores **NULL** y otros no lo son, no se consideran a los valores **NULL**.
- **COUNT(*)** cuenta todos los renglones, aún cuando el valor de cada columna en el renglón sea **NULL**.

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: Count

Contar todas las órdenes fechadas en abril y mayo de 1986.

```
SELECT COUNT(*)
FROM Ordenes
WHERE     FechaOrd BETWEEN
          "01/04/86" AND "31/05/86";
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: Count

Contar los números de stock diferentes que hay en la tabla de stock.

```
SELECT COUNT(*)
FROM Stock;
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

UDE
UNIVERSIDAD DE LA EMPRESA

Select: MAX / MIN

- Para hallar valores máximos y mínimos en columnas aritméticas, dentro de los renglones seleccionados con la cláusula **WHERE**.
- Si una columna contiene valores NULL y valores no NULL, MAX y MIN ignoran los valores NULL.
- Una consulta retornará el valor apropiado basado en el resto de los renglones. Si todos los valores son NULL, MIN y MAX retornarán NULL.

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: Group by

- Un *grupo* es un conjunto de renglones que tienen el mismo valor en una determinada columna. **GROUP BY** se usa para producir un único renglón de resultados, por cada grupo.

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: Group by

Agrupar los artículos de la tabla Artics. Por c/grupo listar el número de orden, la suma de los precios de los artículos del grupo y su valor promedio.

```
SELECT NrOrden, SUM(Precio),AVG(Precio)
FROM Artics
GROUP BY NrOrden;
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: Group by

NOTA: Cuando se agrupan datos, solamente las columnas incluidas en la cláusula **GROUP BY** o en funciones agregadas, pueden aparecer en el conjunto de la cláusula **SELECT**. Sin embargo, una columna en la cláusula **GROUP BY** no necesita estar en el conjunto de la cláusula **SELECT**.

```
SELECT NrOrden, SUM(PrUnidad)
FROM Artics
GROUP BY NrOrden, CodFabr;
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: Group by

NOTA: Cuando se agrupan datos, solamente las columnas incluidas en la cláusula **GROUP BY** o en funciones agregadas, pueden aparecer en el conjunto de la cláusula **SELECT**. Sin embargo, una columna en la cláusula **GROUP BY** no necesita estar en el conjunto de la cláusula **SELECT**.

```
SELECT NrOrden, SUM(PrUnidad)
FROM Artics
GROUP BY NrOrden, CodFabr;
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: HAVING

- La cláusula **HAVING** puede especificarse cuando hay una cláusula **GROUP BY**.
- Permite aplicar una condición para seleccionar los grupos que van a aparecer en el resultado.

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Select: HAVING

Listar en forma agrupada los números de órdenes que aparecen en Artics, indicando la suma de los precios de cada orden. Solamente listar las órdenes cuya suma supere los $ 150.

```
SELECT NrOrden, SUM(Precio)
FROM Artics
GROUP BY NrOrden
HAVING SUM(Precio) > 150;
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Selección vs Proyección

- **Selección**: se implementa mediante la cláusula **WHERE**, la tabla resultante tiene las mismas columnas que la tabla original.

```
SELECT *
FROM Clientes
WHERE Ciudad = "PALO ALTO“;
```

- **Proyección**: se implementa colocando en el **SELECT** el conjunto de atributos sobre el cual se desea proyectar.

```
SELECT DISTINCT Ciudad, Estado, CodPost
FROM Clientes;
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Producto cartesiano

- Se realiza colocando más de una tabla en el **FROM** y sin la sentencia **WHERE**.

  SELECT *
  FROM Fabricnt, Stock;

- NOTA: Analice detalladamente el resultado de la consulta anterior. Han aparecido datos incorrectos debido a que no todas las combinaciones posibles entre los datos de los fabricantes y los datos de stock, son combinaciones válidas.

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

UDE
UNIVERSIDAD DE LA EMPRESA

Join

- Se usa cuando dos tablas tienen una o más columnas en común y pueden vincularse para crear una nueva tabla de resultados.
  - La tabla resultado se compone de todas las columnas y renglones de las tablas que se "joinean“ (unen).
  - El vínculo entre las tablas se menciona en la cláusula WHERE.

- Un JOIN produce una tabla resultado que contiene columnas idénticas.
  - Se obtiene cuando la tabla resultado muestra a todas las columnas de las dos tablas que fueron joineadas.
  - Se usan los nombres de las tablas y alias para distinguir columnas de una tabla y otra, cuando las columnas tienen mismo nombre.

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Join: Ejemplo

Obtener los datos de los fabricantes y los artículos en stock que fabrican.

```
SELECT *
FROM Fabricnt, Stock
WHERE Fabricnt.CodFabr = Stock.CodFabr;
```

O también:

```
SELECT *
FROM Fabricnt F, Stock S
WHERE F.CodFabr = S.CodFabr;
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

UDE
UNIVERSIDAD DE LA EMPRESA

Subconsultas

- Las sentencias **SELECT** pueden anidarse en una cláusula **WHERE** para formar una subconsulta. A la sentencia **SELECT** más exterior en la consulta se la llama "SELECT **externa**". A las interiores se las llama "**anidadadas**".

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Subconsultas / EJ

Obtener los datos de clientes que viven en la misma ciudad en la que vive el cliente cuyo apellido es Higgins.

```
SELECT *
FROM Clientes
WHERE Ciudad =
        (SELECT Ciudad
        FROM Clientes
        WHERE Apellido = "HIGGINS");
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

UDE
UNIVERSIDAD DE LA EMPRESA

Subconsultas

- Una subconsulta puede dar un único valor, ningún valor, o un conjunto de valores.
- El conjunto de la cláusula **SELECT** de las subconsultas que usan las palabras **IN, ALL** o **ANY** deben contener solamente una columna o expresión.
- Si se utiliza la palabra **EXISTS** deben contener un renglón entero.
- La subconsulta no debe tener cláusula **ORDER BY**

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Subconsultas / EJ

Obtener los nombres de los fabricantes que suministran equipamiento de tenis.

```
SELECT  NomFabr
FROM Fabricnt
WHERE CodFabr IN
        (SELECT UNIQUE CodFabr
        FROM Stock
        WHERE Descrip MATCHES "*TENIS*");
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Exists

- La expresión: **EXISTS (SELECT * FROM ...)** es verdadera si y sólo si el resultado del **SELECT** tiene al menos un renglón.

Hallar los nombres de los fabricantes que suministran equipamiento de tenis.

```
SELECT NomFabr
FROM Fabricnt
WHERE EXISTS
        (SELECT *
        FROM Stock
        WHERE Fabricnt.CodFabr = Stock.CodFabr AND
        Descrip MATCHES "*TENIS*");
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Operaciones de conjuntos

- Las operaciones **UNION, INTERSECCION** y **DIFERENCIA** de Algebra Relacional pueden realizarse en SQL.
  - La operación **UNION** está implementada.
  - La **INTERSECCION** y **DIFERENCIA** no están implementadas pero pueden realizarse usando **SELECT** con la cláusula **IN**.

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

UDE
UNIVERSIDAD DE LA EMPRESA

Unión

- La unión de dos tablas produce una tabla que contiene a los renglones de ambas; se realiza usando la cláusula **UNION** entre dos **SELECT**s. Los renglones duplicados son excluidos del resultado. Para mantenerlos se usa **UNION ALL**. Las tablas deben ser compatibles, es decir:
  - Deben tener la misma cantidad de columnas.
  - Los tipos de datos deben ser compatibles para las columnas correspondientes entre sí. Las columnas de tipo CHAR deben tener el mismo ancho.

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Unión / EJ

Listar las órdenes activas y las órdenes cerradas del cliente número 101.

```
SELECT *
FROM Ordenes
WHERE NrClient = 101
UNION
SELECT *
FROM OrdenesC
WHERE NrClient = 101;
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Intersección

- La intersección de dos tablas produce una tabla con los renglones que son comunes a ambas tablas. La intersección se realiza usando **SELECT** con **IN**.

Listar números de clientes y números de orden, de aquellos clientes que tienen órdenes tales que las mismas están activas y a la vez, cerradas (una inconsistencia en los datos!).

```
SELECT NrClient, NrOrden
FROM Ordenes
WHERE NrOrden IN
      (SELECT NrOrden
      FROM OrdenesC);
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados

Diferencia

- La diferencia de dos tablas A y B (A - B) produce una tabla con los renglones de A que no están en B. La diferencia se realiza con **SELECT** y **NOT IN**.

Hacer la diferencia entre las tablas de órdenes y de órdenes cerradas.

```
SELECT NrClient, NrOrden
FROM Ordenes
WHERE NrOrden NOT IN
      (SELECT NrOrden
      FROM OrdenesC);
```

Este material es de uso exclusivo para los cursos impartidos por Universidad de la Empresa y asociados