

3. Funciones de agregación

Las funciones de agregación operan sobre conjuntos de filas de una tabla con el objeto de dar un resultado en un **SELECT**.

Ejemplos:

- Obtener **cuántas** asignaturas existen en la universidad:
- Obtener el **mayor número** de alumnos matriculados en una asignatura
- Obtener la **media** de alumnos matriculados en una asignatura:
- Obtener la **suma total** de los alumnos de las asignaturas:

Funciones de agregación – tipos

AVG([DISTINCT ALL] <i>n</i>)	Valor promedio de <i>n</i> , ignorando nulos.
COUNT(* [DISTINCT ALL] <i>expr</i>)	Número de filas, en donde la <i>expresión</i> evalúa a un valor diferente de nulo. Si se coloca * cuenta todas las filas incluyendo duplicados y filas con nulos.
MAX([DISTINCT ALL] <i>expr</i>)	Máximo valor de la expresión ignorando nulos.
MIN([DISTINCT ALL] <i>expr</i>)	Mínimo valor de la expresión ignorando nulos.
SUM([DISTINCT ALL] <i>n</i>)	Suma de valores de <i>n</i> ignorando nulos

- Los tipos de datos para ***expr*** son **CHAR**, **VARCHAR2**, **NUMBER** y **DATE**.
- Los tipos de datos para ***n*** son numéricos.

Sintaxis

```
SELECT columna, funcion_de_agregacion(columna)
FROM tabla
[WHERE condicion]
[ORDER BY columna];
```

Ejemplo:

" Obtener **cuántas** asignaturas existen en la universidad: "

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

"Obtener **cuántas** asignaturas hay que tengan 9 créditos y sean troncales"

```
SELECT COUNT(*)
FROM ASIGNATURAS
WHERE CREDITOS=9 AND CLASE='T';
```

4. Consultas Agrupadas

La cláusula **GROUP BY** seguida de una lista de atributos permite agrupar las tuplas en grupos que tengan los mismos valores en todos los atributos de esa lista

Ejemplo:

“Obtener el número de profesores de cada área”

```
SELECT COUNT(*) ,AREA
FROM PROFESORES
GROUP BY AREA;
```

Si queremos que aparezca con un alias de columna: nº de profesores
(hay que poner doble comilla “ ya que hay espacios en blanco)

```
SELECT COUNT(*) "nº de profesores",AREA
FROM PROFESORES
GROUP BY AREA;
```

Si queremos que aparezca en el SELECT el literal profesores de

```
SELECT COUNT(*) || ' profesores de '|| AREA
FROM PROFESORES
GROUP BY AREA;
```

La cláusula **HAVING** permite establecer una condición sobre los grupos de manera que sólo se seleccionan aquellos grupos que la cumplen

“listar los profesores y el nº de clases que imparten, pero sólo de aquellos profesores que imparten más de 10 clases”

```
SELECT PROFESOR,COUNT(*)
FROM DOCENCIA
GROUP BY PROFESOR
HAVING COUNT(*)>10;
```