

SQL II

Base de Datos 2022

Join

CONSULTAS EN SQL - JOINS

- Las operaciones de JOIN permiten combinar dos tablas y retornan otra tabla.
- Son productos cartesianos que requieren que las filas en ambas tablas satisfagan ciertas condiciones.
- Existen 4 tipos basicos de JOIN:
 - INNER JOIN
 - LEFT OUTER JOIN
 - RIGHT OUTER JOIN
 - FULL JOIN
- Existen otros tipos de joins:
 - [MySQL Joins](#)



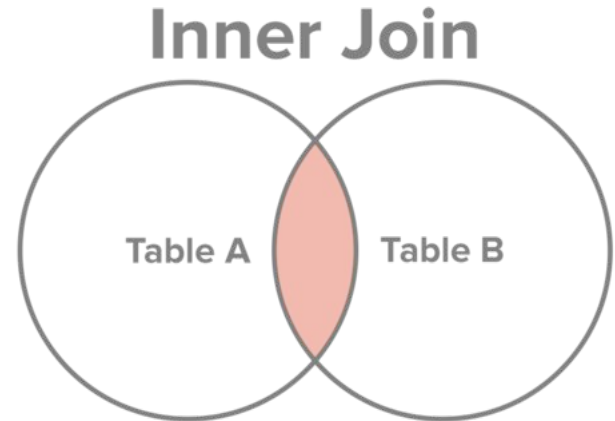
CONSULTAS EN SQL - INNER JOIN

```
SELECT ...  
FROM A  
INNER JOIN B ON join_condition  
...
```

- **join_condition** es un predicado sobre las columnas de A y B
 - $A.id = B.id$
 - $A.x \geq B.y$
- **ON join_condition** se puede reemplazar con **USING(columns)**

Interpretacion:

- Selecciona todas las filas de las tablas A y B donde la condición del join se satisface.



CONSULTAS EN SQL - INNER JOIN

course

course_id	title	dept_name	credits
BIO-301	Genetics	Biology	4
CS-190	Game Design	Comp. Sci.	4
CS-315	Robotics	Comp. Sci.	3

prereq

course_id	prereq_id
BIO-301	BIO-101
CS-190	CS-101
CS-347	CS-101

SELECT course.*, prereq.*

FROM course

INNER JOIN prereq ON course.course_id = prereq.course_id

course_id	title	dept_name	credits	prere_id	course_id
BIO-301	Genetics	Biology	4	BIO-101	BIO-301
CS-190	Game Design	Comp. Sci.	4	CS-101	CS-190

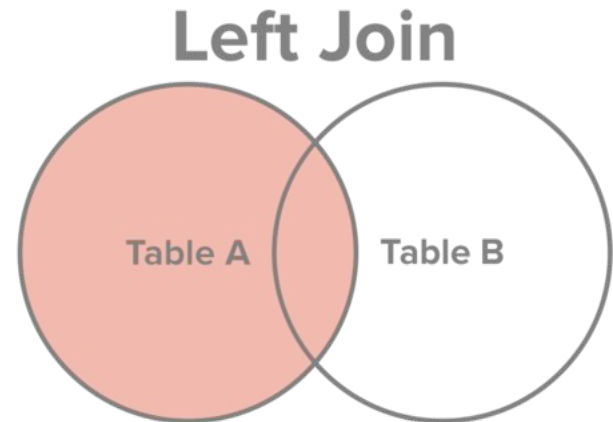
CONSULTAS EN SQL - LEFT JOIN

```
SELECT ...  
FROM A  
LEFT JOIN B ON join_condition  
...
```

- El término **OUTER** es opcional y no tiene efecto.

Interpretacion:

- Selecciona todas las filas de las tablas A y aquellas de la tabla B donde la condición del join se satisface.



CONSULTAS EN SQL - LEFT JOIN

course

<i>course_id</i>	<i>title</i>	<i>dept_name</i>	<i>credits</i>
BIO-301	Genetics	Biology	4
CS-190	Game Design	Comp. Sci.	4
CS-315	Robotics	Comp. Sci.	3

prereq

<i>course_id</i>	<i>prereq_id</i>
BIO-301	BIO-101
CS-190	CS-101
CS-347	CS-101

```
SELECT course.*, prereq.prereq_id  
FROM course  
LEFT JOIN prereq ON course.course_id = prereq.course_id
```

<i>course_id</i>	<i>title</i>	<i>dept_name</i>	<i>credits</i>	<i>prere_id</i>
BIO-301	Genetics	Biology	4	BIO-101
CS-190	Game Design	Comp. Sci.	4	CS-101
CS-315	Robotics	Comp. Sci.	3	<i>null</i>

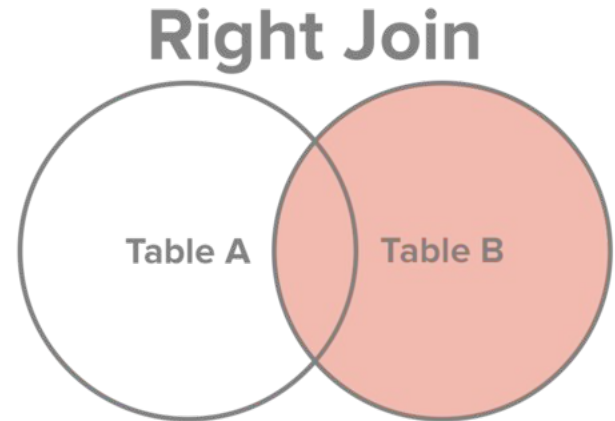
CONSULTAS EN SQL - RIGHT JOIN

```
SELECT ...  
FROM A  
RIGHT JOIN B ON join_condition  
...
```

Interpretacion:

- Selecciona todas las filas de las tablas B y aquellas de la tabla A donde la condición del join se satisface.

- MySQL suele optimizar estos joins transformándolos en LEFT JOINS
 - [Outer Join Optimization](#)
 - [Outer Join Simplification](#)



CONSULTAS EN SQL - RIGHT JOIN

course

<i>course_id</i>	<i>title</i>	<i>dept_name</i>	<i>credits</i>
BIO-301	Genetics	Biology	4
CS-190	Game Design	Comp. Sci.	4
CS-315	Robotics	Comp. Sci.	3

prereq

<i>course_id</i>	<i>prereq_id</i>
BIO-301	BIO-101
CS-190	CS-101
CS-347	CS-101

```
SELECT p.course_id, c.title, c.dept_name,  
       c.credits, p.prereq_id  
FROM course AS c  
RIGHT JOIN prereq AS p ON c.course_id = p.course_id
```

<i>course_id</i>	<i>title</i>	<i>dept_name</i>	<i>credits</i>	<i>prere_id</i>
BIO-301	Genetics	Biology	4	BIO-101
CS-190	Game Design	Comp. Sci.	4	CS-101
CS-347	<i>null</i>	<i>null</i>	<i>null</i>	CS-101

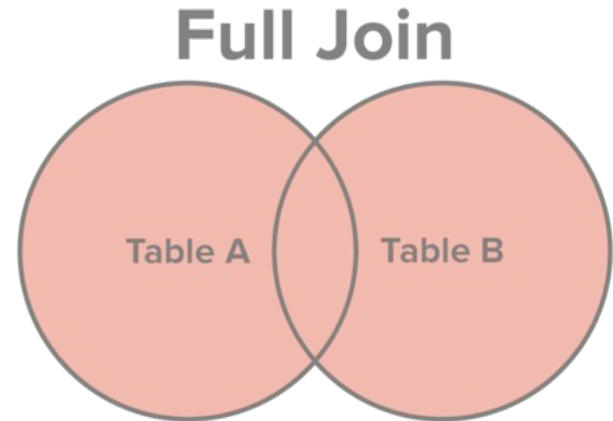
CONSULTAS EN SQL - FULL JOIN

```
SELECT ...  
FROM A  
FULL JOIN B ON join_condition  
...
```

- No es EL producto cartesiano.

Interpretación:

- Selecciona todas las filas de las tablas A y B independientemente de si la condición del join se satisface.



CONSULTAS EN SQL - FULL JOIN

course

course_id	title	dept_name	credits
BIO-301	Genetics	Biology	4
CS-190	Game Design	Comp. Sci.	4
CS-315	Robotics	Comp. Sci.	3

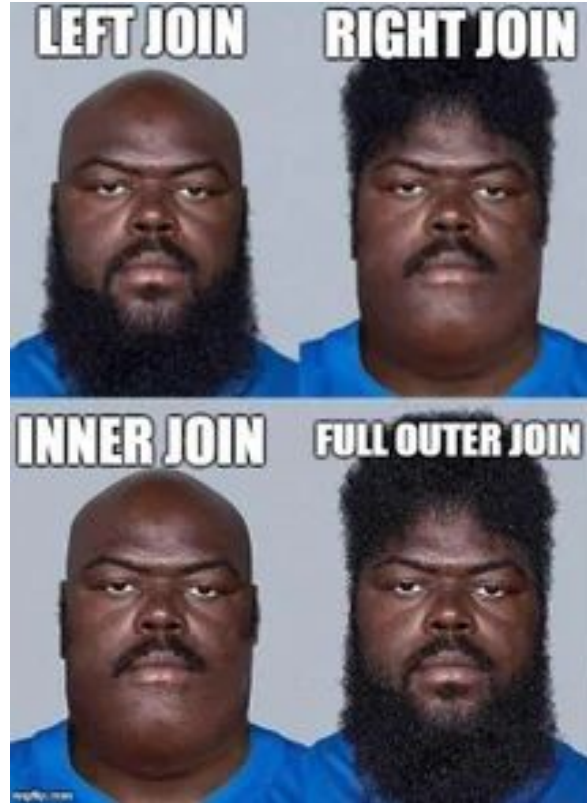
prereq

course_id	prereq_id
BIO-301	BIO-101
CS-190	CS-101
CS-347	CS-101

SELECT *
FROM course
FULL JOIN prereq USING(course_id)

course_id	title	dept_name	credits	prere_id
BIO-301	Genetics	Biology	4	BIO-101
CS-190	Game Design	Comp. Sci.	4	CS-101
CS-315	Robotics	Comp. Sci.	3	null
CS-347	null	null	null	CS-101

CONSULTAS EN SQL - JOINS CHEATSHEET



Operaciones de Conjuntos

CONSULTAS EN SQL - OPERACIONES DE CONJUNTO

```
SELECT ...  
UNION [ALL]  
SELECT ...
```

```
SELECT ...  
INTERSECT [ALL]  
SELECT ...
```

```
SELECT ...  
EXCEPT [ALL]  
SELECT ...
```

SQL provee las siguientes operaciones de conjunto: **union**, **intersect** y **except**.

Todas:

- Operan sobre tablas.
- Eliminan automáticamente los duplicados.
- Para retener duplicados hay que usar **ALL**

CONSULTAS EN SQL - UNION

```
(  
  SELECT course_id  
  FROM section  
  WHERE semester = 'Fall' AND year= 2009  
)  
  
UNION  
  
(  
  SELECT course_id  
  FROM section  
  WHERE semester = 'Spring' AND year= 2010  
)
```

course_id

CS-101
CS-347
PHY-101

course_id

CS-101
CS-315
CS-319
CS-319
FIN-201
HIS-351
MU-199

course_id

CS-101
CS-315
CS-319
CS-347
FIN-201
HIS-351
MU-199
PHY-101

CONSULTAS EN SQL - INTERSECT

```
(  
  SELECT course_id  
  FROM section  
  WHERE semester = 'Fall' AND year= 2009  
)
```

INTERSECT

```
(  
  SELECT course_id  
  FROM section  
  WHERE semester = 'Spring' AND year= 2010  
)
```

course_id

CS-101
CS-347
PHY-101

course_id

CS-101
CS-315
CS-319
CS-319
FIN-201
HIS-351
MU-199

course_id

CS-101

CONSULTAS EN SQL - EXCEPT

```
(  
  SELECT course_id  
  FROM section  
  WHERE semester = 'Fall' AND year= 2009  
)
```

EXCEPT

```
(  
  SELECT course_id  
  FROM section  
  WHERE semester = 'Spring' AND year= 2010  
)
```

course_id

CS-101
CS-347
PHY-101

course_id

CS-347
PHY-101

course_id

CS-101
CS-315
CS-319
CS-319
FIN-201
HIS-351
MU-199