



# TRABAJO GRUPAL MAXIMO 2 INTEGRANTES

## TIPOS DE JOIN

El JOIN se usa para combinar filas de dos o más tablas basadas en una condición de relación entre ellas, generalmente usando claves foráneas.

### 2. Tipos de JOIN:

1. **INNER JOIN**: Este tipo de JOIN retorna solo las filas que tienen coincidencias en ambas tablas.
2. **LEFT JOIN (o LEFT OUTER JOIN)**: Retorna todas las filas de la tabla de la izquierda y las filas coincidentes de la tabla de la derecha. Si no hay coincidencias, devuelve NULL para la tabla de la derecha.
3. **RIGHT JOIN (o RIGHT OUTER JOIN)**: Similar al LEFT JOIN, pero devuelve todas las filas de la tabla de la derecha.
4. **FULL JOIN (o FULL OUTER JOIN)**: Retorna todas las filas cuando hay una coincidencia en una de las tablas, o ambas.

### ADICIONAL:

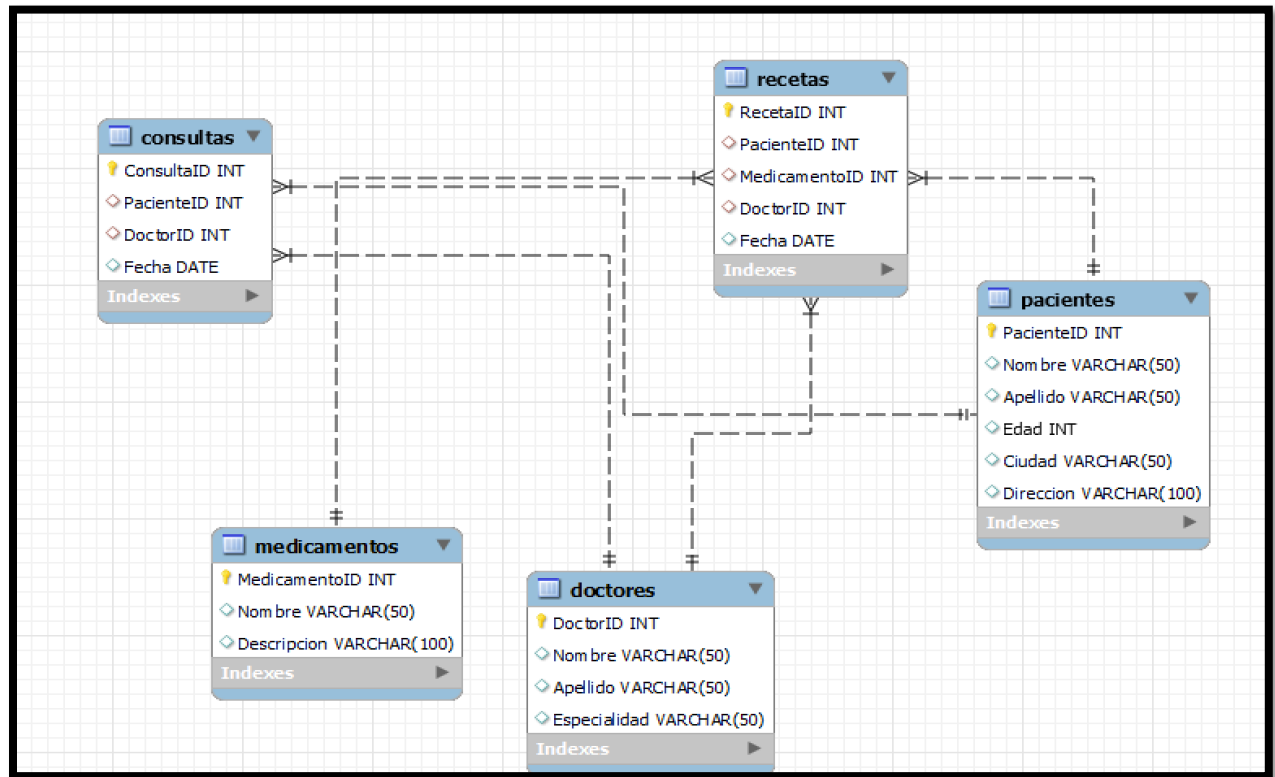
Agrega registros de prueba a las tablas correspondientes para garantizar que los resultados de cada tipo de JOIN sean representativos y completos. Los registros deben ser consistentes con los datos que se requieren para cada consulta y deben permitir visualizar claramente las diferencias en los resultados de cada JOIN.

## INSTRUCCIONES

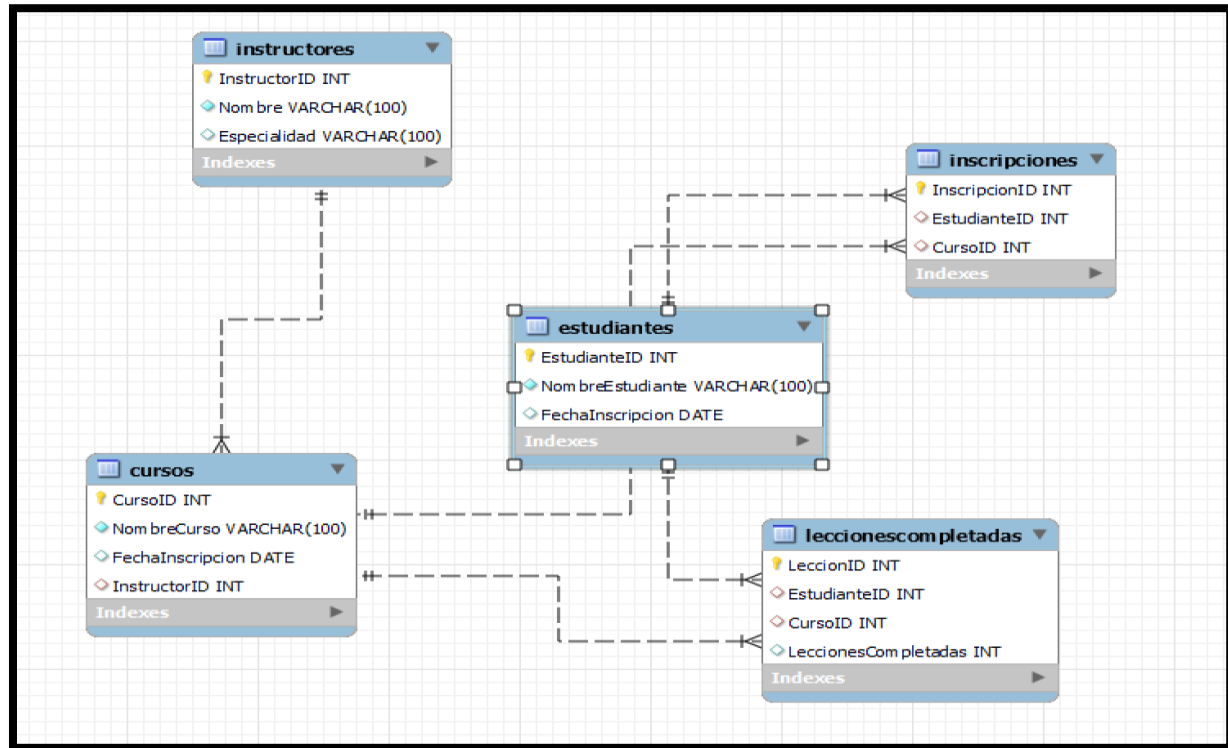
1. **Sube un archivo al git hub** que contenga todos los elementos solicitados.
2. **Incluye los scripts SQL** correspondientes a cada tipo de JOIN (INNER, LEFT, RIGHT, FULL OUTER JOIN) con **comentarios detallados** explicando cada consulta.
3. **Adjunta las capturas de pantalla** de los resultados obtenidos para cada consulta, mostrando los datos de la ejecución de las consultas.

Temática	INNER JOIN	LEFT JOIN	RIGHT JOIN	FULL OUTER JOIN
Clínica	Relacionar <b>pacientes</b> con <b>consultas</b> cuando ambos están registrados.	Mostrar todos los <b>pacientes</b> , incluyendo aquellos que aún no tienen <b>consultas</b> agendadas.	Mostrar todas las <b>consultas</b> , incluso aquellas que no están asignadas a un <b>paciente</b> .	Mostrar todos los <b>pacientes</b> y <b>consultas</b> , incluidas las que no están relacionadas entre sí.
	Relacionar <b>consultas</b> con <b>doctores</b> asignados.	Mostrar todas las <b>consultas</b> , incluyendo las que no tienen <b>doctor</b> asignado.	Mostrar todos los <b>doctores</b> , incluso aquellos que no tienen <b>consultas</b> asignadas.	Mostrar todas las <b>consultas</b> y <b>doctores</b> , incluso las no relacionadas.
	Relacionar <b>medicamentos</b> recetados con los <b>pacientes</b> que los reciben.	Mostrar todos los <b>pacientes</b> , incluyendo aquellos que no reciben <b>medicamentos</b> .	Mostrar todos los <b>medicamentos</b> , incluso los que no han sido recetados.	Mostrar todos los <b>pacientes</b> y <b>medicamentos</b> , incluso los no relacionados entre sí.
Curso	Relacionar <b>estudiantes</b> con <b>cursos</b> en los que están inscritos.	Mostrar todos los <b>estudiantes</b> , incluso aquellos que no están inscritos en ningún <b>curso</b> .	Mostrar todos los <b>cursos</b> , incluso aquellos que no tienen <b>estudiantes</b> inscritos.	Mostrar todos los <b>estudiantes</b> y <b>cursos</b> , incluso los no relacionados entre sí.
	Relacionar <b>cursos</b> con sus <b>instructores</b> .	Mostrar todos los <b>cursos</b> , incluyendo aquellos sin <b>instructor</b> asignado.	Mostrar todos los <b>instructores</b> , incluso aquellos que no tienen <b>cursos</b> a su cargo.	Mostrar todos los <b>cursos</b> y <b>instructores</b> , incluidos los no relacionados entre sí.
	Relacionar <b>estudiantes</b> con <b>lecciones completadas</b> .	Mostrar todos los <b>estudiantes</b> , incluso aquellos que no han completado ninguna <b>lección</b> .	Mostrar todas las <b>lecciones</b> , incluyendo las que no han sido completadas por ningún <b>estudiante</b> .	Mostrar todos los <b>estudiantes</b> y <b>lecciones</b> , incluso las no relacionadas entre sí.
Redes Sociales	Relacionar <b>usuarios</b> con sus <b>fotos</b> publicadas.	Mostrar todos los <b>usuarios</b> , incluyendo aquellos que no han subido ninguna <b>foto</b> .	Mostrar todas las <b>fotos</b> , incluso las que no están asociadas a ningún <b>usuario</b> .	Mostrar todos los <b>usuarios</b> y <b>fotos</b> , incluidas las que no tienen relación entre sí.
	Relacionar <b>fotos</b> con los <b>comentarios</b> recibidos.	Mostrar todas las <b>fotos</b> , incluyendo las que no tienen <b>comentarios</b> .	Mostrar todos los <b>comentarios</b> , incluso los que no están asociados a ninguna <b>foto</b> .	Mostrar todas las <b>fotos</b> y <b>comentarios</b> , incluidos los no relacionados entre sí.
	Relacionar <b>usuarios</b> con sus <b>comentarios</b> realizados.	Mostrar todos los <b>usuarios</b> , incluyendo aquellos que no han realizado ningún <b>comentario</b> .	Mostrar todos los <b>comentarios</b> , incluso los que no están asociados a ningún <b>usuario</b> .	Mostrar todos los <b>usuarios</b> y <b>comentarios</b> , incluidos los no relacionados entre sí.

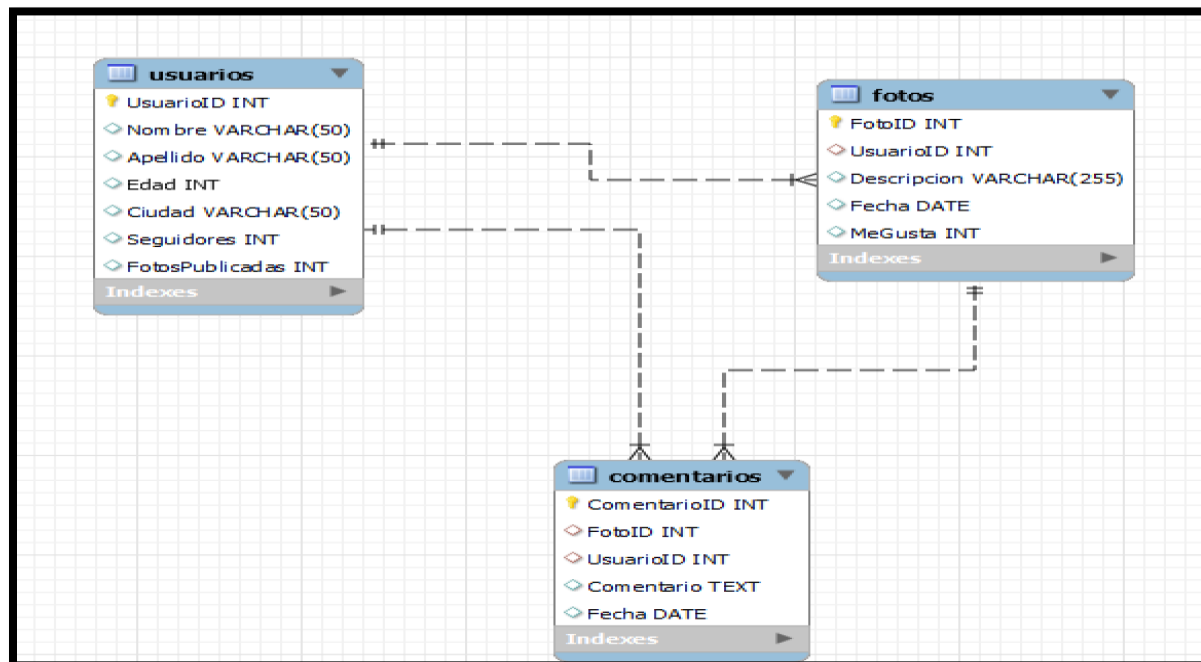
# CLÍNICA



## Cursos ONLINE



## REDES SOCIALES



Subir información del informe y Script al Git hub

# CAPTURAS CLIENTES

-- INNER JOIN: Solo pacientes con consultas

```
81 • SELECT p.Nombre, c.Fecha
82 FROM pacientes p
83 INNER JOIN consultas c ON p.PacienteID = c.PacienteID;
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
Nombre	Fecha		
Ana	2025-05-01		
Luis	2025-05-02		

-- LEFT JOIN: Todos los pacientes, incluso sin consultas

```
85 -- LEFT JOIN: Todos los pacientes, incluso sin consultas
86 • SELECT p.Nombre, c.Fecha
87 FROM pacientes p
88 LEFT JOIN consultas c ON p.PacienteID = c.PacienteID;
89
90 -- RIGHT JOIN: Todas las consultas, incluso sin paciente
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
Nombre	Fecha		
Ana	2025-05-01		
Luis	2025-05-02		
Sofia	NULL		

-- RIGHT JOIN: Todas las consultas, incluso sin paciente

```
90 -- RIGHT JOIN: Todas las consultas, incluso sin paciente
91 • SELECT p.Nombre, c.Fecha
92 FROM pacientes p
93 RIGHT JOIN consultas c ON p.PacienteID = c.PacienteID;
94
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
Nombre	Fecha		
Ana	2025-05-01		
Luis	2025-05-02		
NULL	2025-05-03		

-- FULL OUTER JOIN: Todos los pacientes y consultas (relacionados o no)

```

95 -- FULL OUTER JOIN: Todos los pacientes y consultas (relacionados o no)
96 • SELECT p.Nombre, c.Fecha
97 FROM pacientes p
98 LEFT JOIN consultas c ON p.PacienteID = c.PacienteID
99
100 UNION
101
102 SELECT p.Nombre, c.Fecha
103 FROM pacientes p
104 RIGHT JOIN consultas c ON p.PacienteID = c.PacienteID;

```

Result Grid

	Nombre	Fecha
▶	Ana	2025-05-01
	Luis	2025-05-02
	Sofia	NULL
	NULL	2025-05-03

## CONSULTAS DOCTORES

-- INNER JOIN: Solo consultas con doctor asignado

```

108 -- INNER JOIN: Solo consultas con doctor asignado
109 • SELECT c.ConsultaID, d.Nombre AS Doctor
110 FROM consultas c
111 INNER JOIN doctores d ON c.DoctorID = d.DoctorID;

```

Result Grid

	ConsultaID	Doctor
▶	1	Carlos
	3	Elena

-- LEFT JOIN: Todas las consultas, incluso sin doctor

```

113 -- LEFT JOIN: Todas las consultas, incluso sin doctor
114 • SELECT c.ConsultaID, d.Nombre AS Doctor
115 FROM consultas c
116 LEFT JOIN doctores d ON c.DoctorID = d.DoctorID;
117
118 -- RIGHT JOIN: Todos los doctores, incluso sin consultas
119 • SELECT c.ConsultaID, d.Nombre AS Doctor

```

Result Grid

	ConsultaID	Doctor
▶	2	NULL
	1	Carlos
	3	Elena

-- RIGHT JOIN: Todos los doctores, incluso sin consultas asignadas

```

118 -- RIGHT JOIN: Todos los doctores, incluso sin consultas asignadas
119 • SELECT c.ConsultaID, d.Nombre AS Doctor
120 FROM consultas c
121 RIGHT JOIN doctores d ON c.DoctorID = d.DoctorID;
122
123 -- Simulación de FULL OUTER JOIN entre consultas y doctores
124 • SELECT c.ConsultaID, d.Nombre AS Doctor

```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
ConsultaID	Doctor			
1	Carlos			
3	Elena			
NULL	Andrés			

-- Simulación de FULL OUTER JOIN entre consultas y doctores

```

123 -- Simulación de FULL OUTER JOIN entre consultas y doctores
124 • SELECT c.ConsultaID, d.Nombre AS Doctor
125 FROM consultas c
126 LEFT JOIN doctores d ON c.DoctorID = d.DoctorID
127
128 UNION
129
130 SELECT c.ConsultaID, d.Nombre AS Doctor
131 FROM consultas c
132 RIGHT JOIN doctores d ON c.DoctorID = d.DoctorID;

```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
ConsultaID	Doctor			
2	NULL			
1	Carlos			
3	Elena			
NULL	Andrés			

PACIENTES – MEDICAMENTOS (por recetas)

-- INNER JOIN: Pacientes con medicamentos recetados

```

136 • SELECT p.Nombre AS Paciente, m.Nombre AS Medicamento
137 FROM pacientes p
138 INNER JOIN recetas r ON p.PacienteID = r.PacienteID
139 INNER JOIN medicamentos m ON r.MedicamentoID = m.MedicamentoID;
140
141 -- LEFT JOIN: Todos los pacientes, incluso sin medicamentos

```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
Paciente	Medicamento			
Ana	Paracetamol			
Luis	Ibuprofeno			

-- LEFT JOIN: Todos los pacientes, incluso sin medicamentos



```

141 -- LEFT JOIN: Todos los pacientes, incluso sin medicamentos
142 • SELECT p.Nombre AS Paciente, m.Nombre AS Medicamento
143 FROM pacientes p
144 LEFT JOIN recetas r ON p.PacienteID = r.PacienteID
145 LEFT JOIN medicamentos m ON r.MedicamentoID = m.MedicamentoID;

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
Paciente	Medicamento		
Ana	Paracetamol		
Luis	Ibuprofeno		
Sofia	NULL		

-- RIGHT JOIN: Todos los doctores, incluso sin consultas asignadas

```

147 -- RIGHT JOIN: Todos los medicamentos, incluso si no han sido recetados
148 • SELECT p.Nombre AS Paciente, m.Nombre AS Medicamento
149 FROM pacientes p
150 RIGHT JOIN recetas r ON p.PacienteID = r.PacienteID
151 RIGHT JOIN medicamentos m ON r.MedicamentoID = m.MedicamentoID;
152
153 -- FULL OUTER JOIN: Todos los pacientes y medicamentos, estén relacionad
154 -- Parte 1: LEFT JOIN desde pacientes hacia recetas y medicamentos

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
Paciente	Medicamento		
Ana	Paracetamol		
Luis	Ibuprofeno		
NULL	Amoxicilina		

-- FULL OUTER JOIN: Todos los pacientes y medicamentos, estén relacionados o no

```

155 • SELECT p.Nombre AS Paciente, m.Nombre AS Medicamento
156 FROM pacientes p
157 LEFT JOIN recetas r ON p.PacienteID = r.PacienteID
158 LEFT JOIN medicamentos m ON r.MedicamentoID = m.MedicamentoID
159
160 UNION
161
162 SELECT p.Nombre AS Paciente, m.Nombre AS Medicamento
163 FROM medicamentos m
164 RIGHT JOIN recetas r ON m.MedicamentoID = r.MedicamentoID
165 RIGHT JOIN pacientes p ON p.PacienteID = r.PacienteID
166 WHERE p.PacienteID IS NULL;

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
Paciente	Medicamento		
Ana	Paracetamol		
Luis	Ibuprofeno		
Sofia	NULL		

## CAPTURAS CURSOS ONLINE

### 1. ESTUDIANTES Y CURSOS

-- INNER JOIN: estudiantes inscritos

```

85  -- INNER JOIN: estudiantes inscritos
86  • SELECT e.NombreEstudiante, c.NombreCurso
87  FROM estudiantes e
88  INNER JOIN inscripciones i ON e.EstudianteID = i.EstudianteID
89  INNER JOIN cursos c ON i.CursoID = c.CursoID;

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

	NombreEstudiante	NombreCurso
▶	Luis Pérez	Introducción a SQL
	María Gómez	Introducción a SQL
	María Gómez	Python Intermedio
	José Martínez	Python Intermedio

-- LEFT JOIN: todos los estudiantes, inscritos o no

```

91  -- LEFT JOIN: todos los estudiantes, inscritos o no
92  • SELECT e.NombreEstudiante, c.NombreCurso
93  FROM estudiantes e
94  LEFT JOIN inscripciones i ON e.EstudianteID = i.EstudianteID
95  LEFT JOIN cursos c ON i.CursoID = c.CursoID;
96

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

	NombreEstudiante	NombreCurso
▶	Luis Pérez	Introducción a SQL
	María Gómez	Introducción a SQL
	María Gómez	Python Intermedio
	José Martínez	Python Intermedio
	Lucía Herrera	NULL

-- RIGHT JOIN: todos los cursos, con o sin estudiantes

```

97  -- RIGHT JOIN: todos los cursos, con o sin estudiantes
98  • SELECT e.NombreEstudiante, c.NombreCurso
99  FROM estudiantes e
100 RIGHT JOIN inscripciones i ON e.EstudianteID = i.EstudianteID
101 RIGHT JOIN cursos c ON i.CursoID = c.CursoID;
102

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

	NombreEstudiante	NombreCurso
▶	Luis Pérez	Introducción a SQL
	María Gómez	Introducción a SQL
	María Gómez	Python Intermedio
	José Martínez	Python Intermedio
	NULL	Modelos de ML

-- FULL OUTER JOIN: todos los estudiantes y cursos (con UNION)

```

103 -- FULL OUTER JOIN: todos los estudiantes y cursos (con UNION)
104 • SELECT e.NombreEstudiante, c.NombreCurso
105 FROM estudiantes e
106 LEFT JOIN inscripciones i ON e.EstudianteID = i.EstudianteID
107 LEFT JOIN cursos c ON i.CursoID = c.CursoID
108
109 UNION
110
111 SELECT e.NombreEstudiante, c.NombreCurso
112 FROM estudiantes e
113 RIGHT JOIN inscripciones i ON e.EstudianteID = i.EstudianteID
114 RIGHT JOIN cursos c ON i.CursoID = c.CursoID;

```

NombreEstudiante	NombreCurso
Luis Pérez	Introducción a SQL
María Gómez	Python Intermedio
María Gómez	Introducción a SQL
José Martínez	Python Intermedio
Lucía Herrera	NULL
NULL	Modelos de ML

## 2. Cursos e Instructores

-- INNER JOIN: cursos con instructor

```

118 -- INNER JOIN: cursos con instructor
119 • SELECT c.NombreCurso, i.Nombre AS NombreInstructor
120 FROM cursos c
121 INNER JOIN instructores i ON c.InstructorID = i.InstructorID;
122

```

NombreCurso	NombreInstructor
Introducción a SQL	Carlos Ramírez
Python Intermedio	Ana Torres

-- LEFT JOIN: todos los cursos, incluso sin instructor

```

123 -- LEFT JOIN: todos los cursos, incluso sin instructor
124 • SELECT c.NombreCurso, i.Nombre AS NombreInstructor
125 FROM cursos c
126 LEFT JOIN instructores i ON c.InstructorID = i.InstructorID;
127
128 -- RIGHT JOIN: todos los instructores, incluso sin cursos

```

NombreCurso	NombreInstructor
Introducción a SQL	Carlos Ramírez
Python Intermedio	Ana Torres
Modelos de ML	NULL

-- RIGHT JOIN: todos los instructores, incluso sin cursos

```

128 -- RIGHT JOIN: todos los instructores, incluso sin cursos
129 • SELECT c.NombreCurso, i.Nombre AS NombreInstructor
130 FROM cursos c
131 RIGHT JOIN instructores i ON c.InstructorID = i.InstructorID;

```

Result Grid		Filter Rows:	Export:	Wrap Cell Content: <a href="#">IA</a>
NombreCurso	NombreInstructor			
Introducción a SQL	Carlos Ramírez			
Python Intermedio	Ana Torres			
Modelos de ML	Pedro Mejía			

-- FULL OUTER JOIN: todos los cursos e instructores

```

133 -- FULL OUTER JOIN: todos los cursos e instructores
134 • SELECT c.NombreCurso, i.Nombre AS NombreInstructor
135 FROM cursos c
136 LEFT JOIN instructores i ON c.InstructorID = i.InstructorID
137
138 UNION
139
140 SELECT c.NombreCurso, i.Nombre AS NombreInstructor
141 FROM cursos c
142 RIGHT JOIN instructores i ON c.InstructorID = i.InstructorID;

```

Result Grid		Filter Rows:	Export:	Wrap Cell Content: <a href="#">IA</a>
NombreCurso	NombreInstructor			
Introducción a SQL	Carlos Ramírez			
Python Intermedio	Ana Torres			
Modelos de ML	Pedro Mejía			

### 3. ESTUDIANTES Y LECCIONES COMPLETADAS

-- INNER JOIN: estudiantes que completaron lecciones

```

146 -- INNER JOIN: estudiantes que completaron lecciones
147 • SELECT e.NombreEstudiante, lc.LeccionesCompletadas, c.NombreCurso
148 FROM estudiantes e
149 INNER JOIN leccionescompletadas lc ON e.EstudianteID = lc.EstudianteID
150 INNER JOIN cursos c ON lc.CursoID = c.CursoID;
151

```

Result Grid		Filter Rows:	Export:	Wrap Cell Content: <a href="#">IA</a>
NombreEstudiante	LeccionesCompletadas	NombreCurso		
Luis Pérez	3	Introducción a SQL		
María Gómez	2	Introducción a SQL		
María Gómez	1	Python Intermedio		
José Martínez	4	Python Intermedio		

-- LEFT JOIN: todos los estudiantes

```

152 -- LEFT JOIN: todos los estudiantes
153 • SELECT e.NombreEstudiante, lc.LeccionesCompletadas, c.NombreCurso
154 FROM estudiantes e
155 LEFT JOIN leccionescompletadas lc ON e.EstudianteID = lc.EstudianteID
156 LEFT JOIN cursos c ON lc.CursoID = c.CursoID;

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
NombreEstudiante	LeccionesCompletadas	NombreCurso	
Luis Pérez	3	Introducción a SQL	
María Gómez	2	Introducción a SQL	
María Gómez	1	Python Intermedio	
José Martínez	4	Python Intermedio	
Lucía Herrera	NULL	NULL	

-- RIGHT JOIN: todas las lecciones (por curso)

```

158 -- RIGHT JOIN: todas las lecciones (por curso)
159 • SELECT e.NombreEstudiante, lc.LeccionesCompletadas, c.NombreCurso
160 FROM estudiantes e
161 RIGHT JOIN leccionescompletadas lc ON e.EstudianteID = lc.EstudianteID
162 RIGHT JOIN cursos c ON lc.CursoID = c.CursoID;
163

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
NombreEstudiante	LeccionesCompletadas	NombreCurso	
Luis Pérez	3	Introducción a SQL	
María Gómez	2	Introducción a SQL	
María Gómez	1	Python Intermedio	
José Martínez	4	Python Intermedio	
NULL	NULL	Modelos de ML	

-- FULL OUTER JOIN: todos los estudiantes y lecciones

```

164 -- FULL OUTER JOIN: todos los estudiantes y lecciones
165 • SELECT e.NombreEstudiante, lc.LeccionesCompletadas, c.NombreCurso
166 FROM estudiantes e
167 LEFT JOIN leccionescompletadas lc ON e.EstudianteID = lc.EstudianteID
168 LEFT JOIN cursos c ON lc.CursoID = c.CursoID
169
170 UNION
171
172 SELECT e.NombreEstudiante, lc.LeccionesCompletadas, c.NombreCurso
173 FROM estudiantes e
174 RIGHT JOIN leccionescompletadas lc ON e.EstudianteID = lc.EstudianteID
175 RIGHT JOIN cursos c ON lc.CursoID = c.CursoID;

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
NombreEstudiante	LeccionesCompletadas	NombreCurso	
Luis Pérez	3	Introducción a SQL	
María Gómez	1	Python Intermedio	
María Gómez	2	Introducción a SQL	
José Martínez	4	Python Intermedio	
Lucía Herrera	NULL	NULL	
NULL	NULL	Modelos de ML	

## REDES SOCIALES

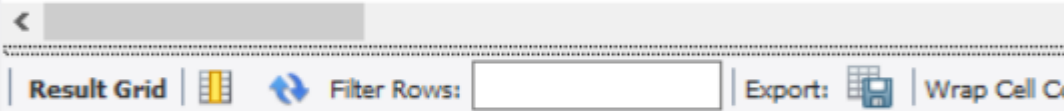
-- INNER JOIN --

-- Usuarios con sus fotos

```

57 • SELECT u.Nombre, f.Descripcion
58 FROM usuarios u
59 INNER JOIN fotos f ON u.UsuarioID = f.UsuarioID;

```

< 


	Nombre	Descripcion
▶	Juan	Atardecer en la playa
	Juan	Mi gato dormido
	Ana	Montañas

-- Fotos con comentarios

```

62 • SELECT f.Descripcion, c.Comentario
63 FROM fotos f
64 INNER JOIN comentarios c ON f.FotoID = c.FotoID;
65

```

< 


	Nombre	Descripcion
▶	Juan	Atardecer en la playa
	Juan	Mi gato dormido
	Ana	Montañas

-- Usuarios con comentarios realizados

```

67 • SELECT u.Nombre, c.Comentario
68 FROM usuarios u
69 INNER JOIN comentarios c ON u.UsuarioID = c.UsuarioID;

```

< 

	Descripcion	Comentario
▶	Atardecer en la playa	Hermosa foto!
	Montañas	Buen encuadre.
	Sin autor	Comentario anónimo.

-- LEFT JOIN -

-- Todos los usuarios y sus fotos (aunque no hayan subido ninguna)

```
79 • SELECT f.Descripcion, c.Comentario
80 FROM fotos f
81 LEFT JOIN comentarios c ON f.FotoID = c.FotoID;
```

<

Result Grid Filter Rows:  Export: Wrap Cell

	Nombre	Descripcion
▶	Juan	Mi gato dormido
	Juan	Atardecer en la playa
	Ana	Montañas
	Luis	NULL

-- Todas las fotos y sus comentarios (aunque no tengan comentarios)

```
79 • SELECT f.Descripcion, c.Comentario
80 FROM fotos f
81 LEFT JOIN comentarios c ON f.FotoID = c.FotoID;
```

<

Result Grid Filter Rows:  Export: Wrap C

	Descripcion	Comentario
▶	Atardecer en la playa	Hermosa foto!
	Mi gato dormido	NULL
	Montañas	Buen encuadre.
	Sin autor	Comentario anónimo.

-- Todos los usuarios y sus comentarios (aunque no hayan hecho comentarios)

```
84 • SELECT u.Nombre, c.Comentario
85 FROM usuarios u
86 LEFT JOIN comentarios c ON u.UsuarioID = c.UsuarioID;
87
```

<

Result Grid Filter Rows:  Export: Wrap Cell Content

	Nombre	Comentario
▶	Juan	Buen encuadre.
	Ana	Hermosa foto!
	Luis	Sin foto comentada.

-- RIGHT JOIN --

-- Todas las fotos y sus autores (aunque no tengan usuario asociado)

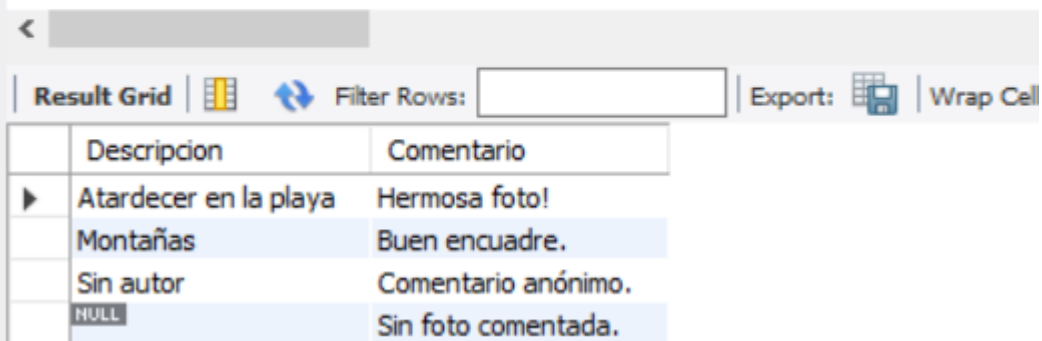
```
91 • SELECT u.Nombre, f.Descripcion
92 FROM usuarios u
93 RIGHT JOIN fotos f ON u.UsuarioID = f.UsuarioID;
```



	Nombre	Descripcion
▶	Juan	Atardecer en la playa
	Juan	Mi gato dormido
	Ana	Montañas
	NULL	Sin autor

-- Todos los comentarios y las fotos (aunque no estén asociadas)

```
96 • SELECT f.Descripcion, c.Comentario
97 FROM fotos f
98 RIGHT JOIN comentarios c ON f.FotoID = c.FotoID;
99
```



	Descripcion	Comentario
▶	Atardecer en la playa	Hermosa foto!
	Montañas	Buen encuadre.
	Sin autor	Comentario anónimo.
	NULL	Sin foto comentada.

-- Todos los comentarios y sus autores (aunque no tengan usuario asociado)






```

101 • SELECT u.Nombre, c.Comentario
102 FROM usuarios u
103 RIGHT JOIN comentarios c ON u.UsuarioID = c.UsuarioID;

```

<

Result Grid   Filter Rows:  | Export:  | Wrap Cell Content: ☐

	Nombre	Comentario
▶	Ana	Hermosa foto!
	Juan	Buen encuadre.
	NULL	Comentario anónimo.
	Luis	Sin foto comentada.

-- FULL OUTER JOIN




-- Usuarios y fotos, incluidas las no relacionadas entre sí

```

108 • SELECT u.Nombre, f.Descripcion
109 FROM usuarios u
110 LEFT JOIN fotos f ON u.UsuarioID = f.UsuarioID
111 UNION
112 SELECT u.Nombre, f.Descripcion
113 FROM usuarios u
114 RIGHT JOIN fotos f ON u.UsuarioID = f.UsuarioID;

```

<

Result Grid   Filter Rows:  | Export:  | Wrap Cell Content: ☐

	Nombre	Descripcion
▶	Juan	Mi gato dormido
	Juan	Atardecer en la playa
	Ana	Montañas
	Luis	NULL
	NULL	Sin autor

-- Fotos y comentarios, incluidos los no relacionados

```

117 • SELECT f.Descripcion, c.Comentario
118 FROM fotos f
119 LEFT JOIN comentarios c ON f.FotoID = c.FotoID
120 UNION
121 SELECT f.Descripcion, c.Comentario
122 FROM fotos f
123 RIGHT JOIN comentarios c ON f.FotoID = c.FotoID;
124

```

Result Grid | Filter Rows: | Export: | Wrap Cell

	Descripcion	Comentario
▶	Atardecer en la playa	Hermosa foto!
	Mi gato dormido	NULL
	Montañas	Buen encuadre.
	Sin autor	Comentario anónimo.
	NULL	Sin foto comentada.

-- Usuarios y comentarios, incluidos los no relacionados

```

126 • SELECT u.Nombre, c.Comentario
127 FROM usuarios u
128 LEFT JOIN comentarios c ON u.UsuarioID = c.UsuarioID
129 UNION
130 SELECT u.Nombre, c.Comentario
131 FROM usuarios u
132 RIGHT JOIN comentarios c ON u.UsuarioID = c.UsuarioID;

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	Nombre	Comentario
▶	Juan	Buen encuadre.
	Ana	Hermosa foto!
	Luis	Sin foto comentada.
	NULL	Comentario anónimo.