**Outer Joins** (Last updated 4/4/2018)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Outer Joins**  It is sometimes useful to return rows regardless of whether there was data in a joined table.  Suppose we want to show a report that includes each FR student. If the student is enrolled in courses, show the courseNr and final grade. If the student isn't enrolled, we still want to show the student. We can use an **outer join**.  Query:  select s.studentNr, e.courseNr, e.final  from Student s  left outer join Enrollment e  on s.studentNr = e.studentNr  where s.classif = "FR"; | **Example #1:** Show freshman students and their courses. Show the student even if the student isn't enrolled in a course.   |  |  | | --- | --- | | Student | Enrollment | | +-----------+---------+  | studentNr | classif |  +-----------+---------+  | 100 | FR |  | 200 | SO |  | 201 | SO |  | 300 | JR |  | 333 | FR |  | 400 | SR |  | 444 | FR |  | 500 | FR |  | 600 | FR |  | 700 | JR |  +-----------+---------+ | +-----------+----------+---------+-------+  | studentNr | courseNr | midTerm | final |  +-----------+----------+---------+-------+  | 100 | CS 1713 | 95 | 90 |  | 100 | ENG1043 | 85 | 95 |  | 100 | HIS1033 | 80 | 85 |  | 100 | MAT1224 | 0 | 0 |  | 200 | CS 1713 | 91 | 90 |  | 200 | MAT1224 | 50 | 80 |  | 201 | CS 1713 | 100 | 90 |  | 201 | ENG1013 | 50 | 50 |  | 201 | MAT1224 | 0 | 0 |  | 300 | CS 1713 | 80 | 80 |  | 444 | CS 1713 | 100 | 90 |  | 500 | CS 1713 | 100 | 80 |  | 500 | HIS1033 | 95 | 90 |  | 700 | CS 1713 | 80 | 80 |  +-----------+----------+---------+-------+ |   **Query Result**:  +-----------+----------+-------+  | studentNr | courseNr | final |  +-----------+----------+-------+  | 100 | CS 1713 | 90 |  | 100 | ENG1043 | 95 |  | 100 | HIS1033 | 85 |  | 100 | MAT1224 | 0 |  | 333 | NULL | NULL |  | 444 | CS 1713 | 90 |  | 500 | CS 1713 | 80 |  | 500 | HIS1033 | 90 |  | 600 | NULL | NULL |  +-----------+----------+-------+ |
| **Left Outer Join** All rows from the **left** table are included, unmatched rows from the right are returned as NULL values.  **Example #2, 3, 4: Data**   |  |  | | --- | --- | | One Table | Two Table | | +---+---+  | x | y |  +---+---+  | A | 1 |  | A | 2 |  | C | 1 |  | C | 2 |  | C | 3 |  | E | 1 |  | E | 2 |  +---+---+ | +---+---+  | x | z |  +---+---+  | A | 3 |  | B | 1 |  | B | 2 |  | C | 4 |  | C | 5 |  | D | 3 |  | D | 4 |  +---+---+ | | **Example #2: Left Outer Join** – all rows from left, unmatched rows from right are returned as NULL.  select o.x, o.y, t.x, t.z  from One o  left outer join Two t  on o.x = t.x;  **Result:**  +---+---+------+------+  | x | y | x | z |  +---+---+------+------+  | A | 1 | A | 3 |  | A | 2 | A | 3 |  | C | 1 | C | 4 |  | C | 1 | C | 5 |  | C | 2 | C | 4 |  | C | 2 | C | 5 |  | C | 3 | C | 4 |  | C | 3 | C | 5 |  | E | 1 | NULL | NULL |  | E | 2 | NULL | NULL |  +---+---+------+------+ |
| **Right Outer Join** All rows from the **right** table are included, unmatched rows from the left are returned as NULL values.  **Example #2, 3, 4: Data**   |  |  | | --- | --- | | One Table | Two Table | | +---+---+  | x | y |  +---+---+  | A | 1 |  | A | 2 |  | C | 1 |  | C | 2 |  | C | 3 |  | E | 1 |  | E | 2 |  +---+---+ | +---+---+  | x | z |  +---+---+  | A | 3 |  | B | 1 |  | B | 2 |  | C | 4 |  | C | 5 |  | D | 3 |  | D | 4 |  +---+---+ | | **Example #3: Right Outer Join** – all rows from right, unmatched rows from left are returned as NULL.  select o.x, o.y, t.x, t.z  from One o  right outer join Two t  on o.x = t.x;  **Result:**  +------+------+---+---+  | x | y | x | z |  +------+------+---+---+  | A | 1 | A | 3 |  | A | 2 | A | 3 |  | NULL | NULL | B | 1 |  | NULL | NULL | B | 2 |  | C | 1 | C | 4 |  | C | 2 | C | 4 |  | C | 3 | C | 4 |  | C | 1 | C | 5 |  | C | 2 | C | 5 |  | C | 3 | C | 5 |  | NULL | NULL | D | 3 |  | NULL | NULL | D | 4 |  +------+------+---+---+ |
| **Full Outer Join** All rows from **both** tables are included, NULL values fill unmatched rows.  **Example #2, 3, 4: Data**   |  |  | | --- | --- | | One Table | Two Table | | +---+---+  | x | y |  +---+---+  | A | 1 |  | A | 2 |  | C | 1 |  | C | 2 |  | C | 3 |  | E | 1 |  | E | 2 |  +---+---+ | +---+---+  | x | z |  +---+---+  | A | 3 |  | B | 1 |  | B | 2 |  | C | 4 |  | C | 5 |  | D | 3 |  | D | 4 |  +---+---+ | | **Example #4: Full Outer Join** – all rows from **both** tables are included, NULL values fill unmatched rows.  select o.x, o.y, t.x, t.z  from One o  full outer join Two t  on o.x = t.x;  Unfortunately, **MySQL doesn't support full outer join**. We can use **UNION** to get a similar result.  select o.x, o.y, t.x, t.z  from One o  left outer join Two t  on o.x = t.x  union  select o.x, o.y, t.x, t.z  from One o  right outer join Two t  on o.x = t.x;  **Result:**  +------+------+------+------+  | x | y | x | z |  +------+------+------+------+  | A | 1 | A | 3 |  | A | 2 | A | 3 |  | C | 1 | C | 4 |  | C | 1 | C | 5 |  | C | 2 | C | 4 |  | C | 2 | C | 5 |  | C | 3 | C | 4 |  | C | 3 | C | 5 |  | E | 1 | NULL | NULL |  | E | 2 | NULL | NULL |  | NULL | NULL | B | 1 |  | NULL | NULL | B | 2 |  | NULL | NULL | D | 3 |  | NULL | NULL | D | 4 |  +------+------+------+------+ |
|  |  |
|  |  |

**©2018 Larry W. Clark,** UTSA CS students may make copies for their personal use