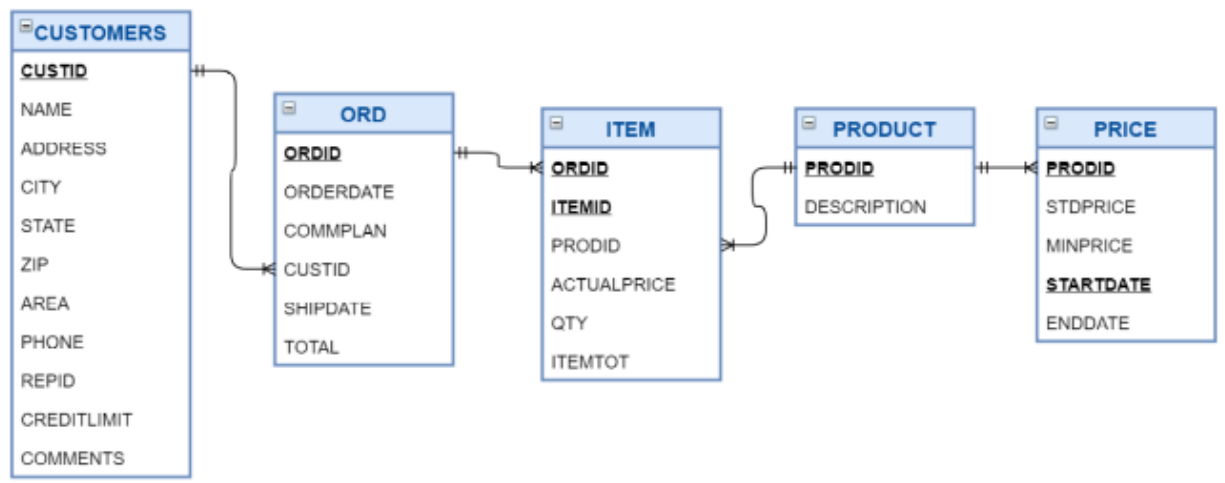




Exercici 3. JOIN's

PAS PREVI: Abans de començar la pràctica, importa la resta de taules de la base de dades companyCSV. (SALGRADE, PRODUCT, ORD, ITEM, CUSTOMER, PRICE). No calen captures. Trobaràs el fitxer al Moodle: **Taules BBDD exemple.zip**



Recorda:

SELECT taula1.columnaX, taula2.columnaY, ... (obligatori)
FROM taula1 [INNER|LEFT|RIGHT|FULL] **JOIN** taula2 (obligatori)
ON taula1.columnaZ [Operador] taula2.columnaZ (obligatori casi sempre)
WHERE per filtrar files/registres abans d'agrupar (si cal)
GROUP BY per agrupar (si cal)
HAVING per filtrar després d'agrupar (si cal)
ORDER BY per ordenar files/registres (si cal)

Utilitzant la BBDD companyCSV fes les següents consultes per línia de comandes. Adjunta captura i SQL en text:

1. Mostra per cada treballador en quina ciutat treballa. Mostra Nom treballador i ciutat.



```
MariaDB [companycsv]> SELECT emp.ename AS Treballador, dept.loc AS Ciutat  
-> FROM emp INNER JOIN dept  
-> ON emp.deptno = dept.deptno;  
+-----+-----+  
| Treballador | Ciutat |  
+-----+-----+  
| KING        | NEW YORK |  
| BLAKE       | CHICAGO  |  
| CLARK       | NEW YORK |  
| JONES       | DALLAS   |  
| MARTIN      | CHICAGO  |  
| ALLEN       | CHICAGO  |  
| TURNER      | CHICAGO  |  
| JAMES       | CHICAGO  |  
| WARD        | CHICAGO  |  
| FORD        | DALLAS   |  
| SMITH       | DALLAS   |  
| SCOTT       | DALLAS   |  
| ADAMS       | DALLAS   |  
| MILLER      | NEW YORK |  
+-----+-----+  
14 rows in set (0.001 sec)
```

SELECT emp.ename AS Treballador, dept.loc AS Ciutat

FROM emp INNER JOIN dept

ON emp.deptno = dept.deptno;

- Mostra tots els llocs de treball (job) que hi ha al departament 30. Mostra Lloc de treball (job) i ciutat del departament.

```
MariaDB [companycsv]> SELECT emp.job AS Lloc, dept.loc AS Ciutat  
-> FROM emp INNER JOIN dept  
-> ON emp.deptno = dept.deptno  
-> WHERE emp.deptno = 30;  
+-----+-----+  
| Lloc      | Ciutat |  
+-----+-----+  
| MANAGER   | CHICAGO |  
| SALESMAN  | CHICAGO |  
| SALESMAN  | CHICAGO |  
| SALESMAN  | CHICAGO |  
| CLERK     | CHICAGO |  
| SALESMAN  | CHICAGO |  
+-----+-----+  
6 rows in set (0.000 sec)
```



SELECT emp.job AS Lloc, dept.loc AS Ciutat

FROM emp INNER JOIN dept

ON emp.deptno = dept.deptno

WHERE emp.deptno = 30;

3. Vull veure una columna amb el nom de cada treballador i una altra amb el nom del seu respectiu cap. Només dels departaments 10 i 20.

```
MariaDB [companycsv]> SELECT Nom.ename AS NOM, Cap.ename AS CAP  
-> FROM emp AS Nom INNER JOIN emp AS Cap  
-> ON Nom.mgr = Cap.empno  
-> WHERE Nom.deptno IN (10,20);
```

NOM	CAP
CLARK	KING
JONES	KING
MILLER	CLARK
FORD	JONES
SCOTT	JONES
SMITH	FORD
ADAMS	SCOTT

7 rows in set, 14 warnings (0.000 sec)

SELECT Nom.ename AS NOM, Cap.ename AS CAP

FROM emp AS Nom INNER JOIN emp AS Cap

ON Nom.mgr = Cap.empno

WHERE Nom.deptno IN (10,20);

4. Què he de fer si també vull veure l'empleat KING ? Mostra el resultat. No pot sortir cap valor NULL, al seu lloc digues que "No té CAP".



DAM. M02 BASES DE DADES. UF2 - Llenguatges SQL: DML i DDL

```
MariaDB [companycsv]> SELECT Nom.ename AS NOM, IFNULL(Cap.ename, "No te cap") AS CAP
-> FROM emp AS Nom LEFT JOIN emp AS Cap
-> ON Nom.mgr = Cap.empno
-> WHERE Nom.deptno IN (10,20);
```

NOM	CAP
CLARK	KING
JONES	KING
MILLER	CLARK
FORD	JONES
SCOTT	JONES
SMITH	FORD
ADAMS	SCOTT
KING	No te cap

8 rows in set, 14 warnings (0.000 sec)

SELECT Nom.ename AS NOM, IFNULL(Cap.ename, "No te cap") AS CAP

FROM emp AS Nom LEFT JOIN emp AS Cap

ON Nom.mgr = Cap.empno

WHERE Nom.deptno IN (10,20);

5. Mostra el nivell de salari de tots els treballadors que guanyen menys de 2800. Ordenat per salari (grade) descendent. Mostra 3 columnes (Nom de treballador, Nivell de salari i salari)

```
MariaDB [companycsv]> SELECT emp.ename AS "Nom del treballador", salgrade.grade AS "Nivell de salari", emp.salary AS Salari
-> FROM emp INNER JOIN salgrade
-> ON emp.salary BETWEEN salgrade.losal AND salgrade.hisal
-> WHERE emp.salary < 2800
-> ORDER BY emp.salary DESC;
```

Nom del treballador	Nivell de salari	Salari
CLARK	4	2450
ALLEN	3	1600
TURNER	3	1500
MILLER	2	1300
WARD	2	1250
MARTIN	2	1250
ADAMS	1	1100
JAMES	1	950
SMITH	1	800

9 rows in set (0.001 sec)

SELECT emp.ename AS "Nom del treballador", salgrade.grade AS "Nivell de salari", emp.salary AS Salari

FROM emp INNER JOIN salgrade

ON emp.salary BETWEEN salgrade.losal AND salgrade.hisal



WHERE emp.salary<2800

ORDER BY emp.salary DESC;

6. Mostra els treballadors que treballen en un departament on en el seu nom (del departament) hi hagi alguna A. Mostra (Nom del treballador , num departament i nom del departament)

```
MariaDB [companycsv]> SELECT emp.ename AS Nom, emp.deptno AS "Numero departament", dept.dname AS "Nom departament"
-> FROM emp INNER JOIN dept
-> ON emp.deptno = dept.deptno
-> WHERE dept.dname LIKE '%A%';
```

Nom	Numero departament	Nom departament
KING	10	ACCOUNTING
BLAKE	30	SALES
CLARK	10	ACCOUNTING
JONES	20	RESEARCH
MARTIN	30	SALES
ALLEN	30	SALES
TURNER	30	SALES
JAMES	30	SALES
WARD	30	SALES
FORD	20	RESEARCH
SMITH	20	RESEARCH
SCOTT	20	RESEARCH
ADAMS	20	RESEARCH
MILLER	10	ACCOUNTING

14 rows in set (0.001 sec)

SELECT emp.ename AS Nom, emp.deptno AS "Numero departament", dept.dname AS "Nom departament"

FROM emp INNER JOIN dept

ON emp.deptno = dept.deptno

WHERE dept.dname LIKE '%A%';

7. Fes una consulta que mostri quants treballadors té cada departament (inclòs el departament 40). Mostra el número de departament, el nom del departament i el total d'empleats.

```
MariaDB [companycsv]> SELECT dept.deptno AS "Numero departament", dept.dname AS "Nom departament", COUNT(emp.ename) AS Total
-> FROM emp RIGHT JOIN dept
-> ON emp.deptno = dept.deptno
-> GROUP BY dept.dname, dept.deptno
-> ORDER BY dept.deptno ASC;
```

Numero departament	Nom departament	Total
10	ACCOUNTING	3
20	RESEARCH	5
30	SALES	6
40	OPERATIONS	0

4 rows in set (0.001 sec)



DAM. M02 BASES DE DADES. UF2 - Llenguatges SQL: DML i DDL

```
SELECT dept.deptno AS "Numero departament", dept.dname AS "Nom departament",  
COUNT(emp.ename) AS Total
```

```
ON emp.deptno = dept.deptno
```

```
FROM emp RIGHT JOIN dept
```

```
GROUP BY dept.dname, dept.deptno
```

```
ORDER BY dept.deptno ASC;
```

8. Respecte la consulta anterior: El número del departament de quina taula el mostres ? De la taula EMP o de la taula DEPT ? Què passa si agafes el número de departament de l'altra taula? Raona la resposta.

He posat de **dept**, i em dona el següent:

```
MariaDB [companycsv]> SELECT dept.deptno AS "Numero departament", dept.dname AS "Nom departament", COUNT(emp.ename) AS Total  
-> FROM emp RIGHT JOIN dept  
-> ON emp.deptno = dept.deptno  
-> GROUP BY dept.dname, dept.deptno  
-> ORDER BY dept.deptno ASC;
```

Numero departament	Nom departament	Total
10	ACCOUNTING	3
20	RESEARCH	5
30	SALES	6
40	OPERATIONS	0

4 rows in set (0.001 sec)

Si poso d'emp, em surt el departament 40 **NULL**:

```
MariaDB [companycsv]> SELECT emp.deptno AS "Numero departament", dept.dname AS "Nom departament", COUNT(emp.ename) AS Total  
-> FROM emp RIGHT JOIN dept  
-> ON emp.deptno = dept.deptno  
-> GROUP BY dept.dname, dept.deptno  
-> ORDER BY dept.deptno ASC;
```

Numero departament	Nom departament	Total
10	ACCOUNTING	3
20	RESEARCH	5
30	SALES	6
NULL	OPERATIONS	0

4 rows in set (0.001 sec)

Ja que he utilitzat el **RIGHT JOIN** i això vol dir que agafa totes les taules de la dreta (dept.), i no de l'esquerda, si agafo emp, donarà null, si fos **INNER JOIN** donaria el mateix.

9. Mostra el número de departament, el nom de l'empleat i el nom dels seus companys de departament PISTES: Es pot fer amb una recursiva + producte cartesià. Atenció: Un treballador no pot ser company d'ell mateix.

A partir d'ara utilitza les taules ORD, CUSTOMER, ITEM. Primer fes un DESC per entendre



quins camps hi ha i quina relació hi ha entre ells.

- 10.** Mostra el número de comanda juntament amb la data i els productes que s'han comprat en cada comanda. Només vull veure els productes que comencin pel codi 100. Mostra Data, id_comanda, id_producte

```
MariaDB [companycsv]> SELECT ord.orderdate AS Data, ord.ordid AS "ID comanda", item.prodid AS "ID producte"
-> FROM ord INNER JOIN ITEM
-> ON ord.ordid = item.ordid
-> WHERE item.prodid LIKE '100%';
```

Data	ID comanda	ID producte
07/01/07	610	100890
11/01/07	611	100861
15/01/07	612	100860
05/06/06	602	100870
15/06/06	604	100890
15/06/06	604	100861
15/06/06	604	100860
05/06/06	603	100860
07/01/07	610	100860
07/01/07	610	100870
01/02/07	614	100860
01/02/07	614	100870
15/01/07	612	100861
12/03/07	620	100860
01/02/07	613	100871
05/02/07	617	100860
05/02/07	617	100861
01/02/07	614	100871

18 rows in set (0.001 sec)

```
SELECT ord.orderdate AS Data, ord.ordid AS "ID comanda", item.prodid AS "ID producte"

FROM ord INNER JOIN ITEM

ON ord.ordid = item.ordid

WHERE item.prodid LIKE '100%';
```

- 11.** Igual que l'anterior, però ara afegeix-hi una quarta columna que sigui la descripció del producte.



DAM. M02 BASES DE DADES. UF2 - Llenguatges SQL: DML i DDL

```
MariaDB [companycsv]> SELECT ORD.ORDERDATE AS 'Data', ORD.ORDID AS 'id_comanda', ITEM.PRODID AS 'id_producte', PRODUCT.DESCRPTION AS 'Descripcio producte' FROM ORD INNER JOIN ITEM ON ORD.ORDID = ITEM.ORDID INNER JOIN PRODUCT ON ITEM.PRODID = PRODUCT.PRODID WHERE ITEM.PRODID LIKE '100%';
```

Data	id_comanda	id_producte	Descripcio producte
07/01/07	610	100890	ACE TENNIS NET
11/01/07	611	100861	ACE TENNIS RACKET II
15/01/07	612	100860	ACE TENNIS RACKET I
05/06/06	602	100870	ACE TENNIS BALLS-3 PACK
15/06/06	604	100890	ACE TENNIS NET
15/06/06	604	100861	ACE TENNIS RACKET II
15/06/06	604	100860	ACE TENNIS RACKET I
05/06/06	603	100860	ACE TENNIS RACKET I
07/01/07	610	100860	ACE TENNIS RACKET I
07/01/07	610	100870	ACE TENNIS BALLS-3 PACK
01/02/07	614	100860	ACE TENNIS RACKET I
01/02/07	614	100870	ACE TENNIS BALLS-3 PACK
15/01/07	612	100861	ACE TENNIS RACKET II
12/03/07	620	100860	ACE TENNIS RACKET I
01/02/07	613	100871	ACE TENNIS BALLS-6 PACK
05/02/07	617	100860	ACE TENNIS RACKET I
05/02/07	617	100861	ACE TENNIS RACKET II
01/02/07	614	100871	ACE TENNIS BALLS-6 PACK

18 rows in set (0.002 sec)

SELECT ORD.ORDERDATE AS 'Data', ORD.ORDID AS 'id_comanda', ITEM.PRODID AS 'id_producte', PRODUCT.DESCRPTION AS 'Descripcio producte'

' FROM ORD INNER JOIN ITEM ON ORD.ORDID = ITEM.ORDID INNER JOIN PRODUCT

ON ITEM.PRODID = PRODUCT.PRODID

WHERE ITEM.PRODID LIKE '100%';

12. Fes una consulta per veure el total d'unitats venudes per cada producte. Mostra Descripció producte i Total unitats.

```
MariaDB [companycsv]> SELECT PRODUCT.DESCRPTION AS 'Descripcio producte', SUM(ITEM.QTY) AS 'Total unitats' FROM PRODUCT INNER JOIN ITEM ON PRODUCT.PRODID = ITEM.PRODID GROUP BY PRODUCT.DESCRPTION;
```

Descripcio producte	Total unitats
ACE TENNIS BALLS-3 PACK	1023
ACE TENNIS BALLS-6 PACK	1100
ACE TENNIS NET	4
ACE TENNIS RACKET I	619
ACE TENNIS RACKET II	123
RH: "GUIDE TO TENNIS"	600
SB ENERGY BAR-6 PACK	1201
SB VITA SNACK-6 PACK	150
SP JUNIOR RACKET	150
SP TENNIS RACKET	200

10 rows in set (0.001 sec)

SELECT PRODUCT.DESCRPTION AS 'Descripcio producte', SUM(ITEM.QTY) AS 'Total unitats'

FROM PRODUCT INNER JOIN ITEM



ON PRODUCT.PRODID = ITEM.PRODID

GROUP BY PRODUCT.DESCRPTION;

- 13.** Fes una consulta per veure els productes amb el seu preu mitjà. Mostra Descripció producte i preu mitjà.

```
MariaDB [companycsv]> SELECT PRODUCT.DESCRPTION AS 'Descripcio producte', AVG(ITEM.ACTUALPRICE) AS 'Preu mitja' FROM PRODUCT INNER JOIN ITEM ON PRO  
DUCT.PRODID = ITEM.PRODID GROUP BY PRODUCT.DESCRPTION;
```

Descripcio producte	Preu mitja
ACE TENNIS BALLS-3 PACK	2.80000
ACE TENNIS BALLS-6 PACK	5.60000
ACE TENNIS NET	58.00000
ACE TENNIS RACKET I	38.57143
ACE TENNIS RACKET II	43.12500
RH: "GUIDE TO TENNIS"	3.40000
SB ENERGY BAR-6 PACK	2.33333
SB VITA SNACK-6 PACK	40.50000
SP JUNIOR RACKET	10.00000
SP TENNIS RACKET	24.00000

SELECT PRODUCT.DESCRPTION AS 'Descripcio producte', AVG(ITEM.ACTUALPRICE) AS 'Preu mitja'

FROM PRODUCT INNER JOIN ITEM

ON PRO DUCT.PRODID = ITEM.PRODID

GROUP BY PRODUCT.DESCRPTION;

- 14. EXTRA:** Igual que l'anterior, però només vull veure els productes venuts entre 2 dates determinades (tria tu les dates). Tampoc vull veure els productes que el codi acabi en 0. No cal afegir cap columna nova. Ordena de forma descendent per preu mitjà.