



**INSTITUTO POLITÉCNICO  
NACIONAL**



**ESCUELA SUPERIOR DE CÓMPUTO**

**BASE DE DATOS**

**PRÁCTICA 6**

**PROFESOR: HERNÁNDEZ CONTRERAS EULER**

**2CM10**

**PEREZ RAYA ALEJANDRO ADOLFO**

## Índice

1. Marco Teórico.....	3
2. Instrucciones.....	5
3. Desarrollo .....	7
4. Conclusión .....	15
5. Bibliografía .....	15

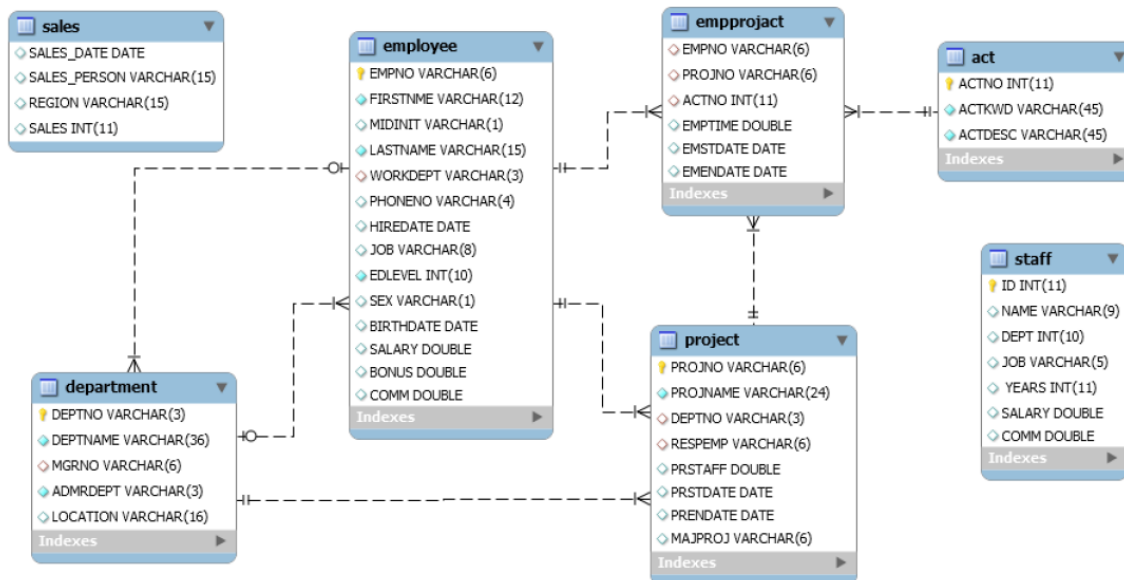
## 1. Marco Teórico

Comandos	Descripción
<b>SELECT * FROM nom_Relación;</b>	Sintaxis general para consultar todos los datos de la relación.
<b>SELECT campo AS newNameCampo FROM nom_Relación;</b>  <b>SELECT campo FROM nom_Relación;</b>	Comando para consultar solo el campo indicado, la palabra reservada AS renombra el nombre del campo solo para la salida a pantalla, el uso de AS es opcional.
<b>SELECT campo1, campo2 FROM nom_Relación;</b>	Si se desea consultar más de un campo de la relación, se deben de poner separados por una coma.
<b>SELECT campo FROM nom_Relación WHERE campo="Chihuahua";</b>	<p>La palabra clave WHERE seguida por una expresión que indica la condición o condiciones que deben satisfacer los registros para ser seleccionados. Todas las consultas pueden usar la palabra clave WHERE y si se desea más de una condición basta con poner la palabra AND entre cada una de las condiciones.</p> <p>OPCION: Puede ocupar las siguientes condiciones.</p> <ul style="list-style-type: none"><li>• Campo="Valor"</li><li>• Campo&gt;Valor</li><li>• Campo&lt;Valor</li><li>• Campo like "Valor%"</li></ul> <p>El % indica que no importa lo que este adelante después del valor que buscamos.</p>
<b>SELECT a.campo, b.campo2 FROM nom_Relación a, nomRelación2 b;</b>	Para consultar diferentes valores de diferentes relaciones desde una sola línea de comando, usamos alias (letras en rojo) al principio del campo a consultar ponemos alias.campo y después del nombre de la relación ponemos el nombre del alias que ocupamos en el campo (sin el punto).
<b>SELECT COUNT(*) FROM nom_Relación;</b>	Este comando cuenta el total de tuplas existentes en la relación.

<b>SELECT * FROM nom_Relación ORDER BY campo DESC;</b>	La palabra reservada ORDER BY ordena la salida con respecto al campo que se le indique y la palabra DESC o ASC se refieren a orden descendente o ascendente respectivamente.
<b>SELECT DISTINCT campo FROM nom_Relación</b>	La palabra reservada DISTINCT especifica que los registros duplicados en el conjunto de resultados no deben de ser mostrados.
<b>SELECT campo FROM nom_Relación GROUP BY campo;</b>	La palabra reservada GROUP BY realiza la tarea del ORDER BY y de DISTINCT a la vez, quita los valores repetidos y los ordena.
<b>INSERT INTO nom_Relación (campo1,campo2,...,campoN) VALUES (val1,val2,...,valN);</b>	Este comando nos permite instanciar los campos del registro, si se van a llenar todos los campos se puede omitir el parte de (campo1,campo2,...,campon).
<b>SELECT * FROM nomRelación WHERE campo BETWEEN valor1 and valor2 ;</b>	El comando BETWEEN nos ayuda en las condiciones para buscar los datos donde su campo este entre los valores indicados.
<b>SELECT * FROM nomRelación WHERE campo IN (valor1,valor2,...,valorn);</b>	EL comando IN es una condición del WHERE el cual indica que se mostrara los datos donde el valor sea igual a valor1 o valor2 o ... o valor n.

## 2. Instrucciones

Resuelve las siguientes consultas usando el siguiente modelo relacional.

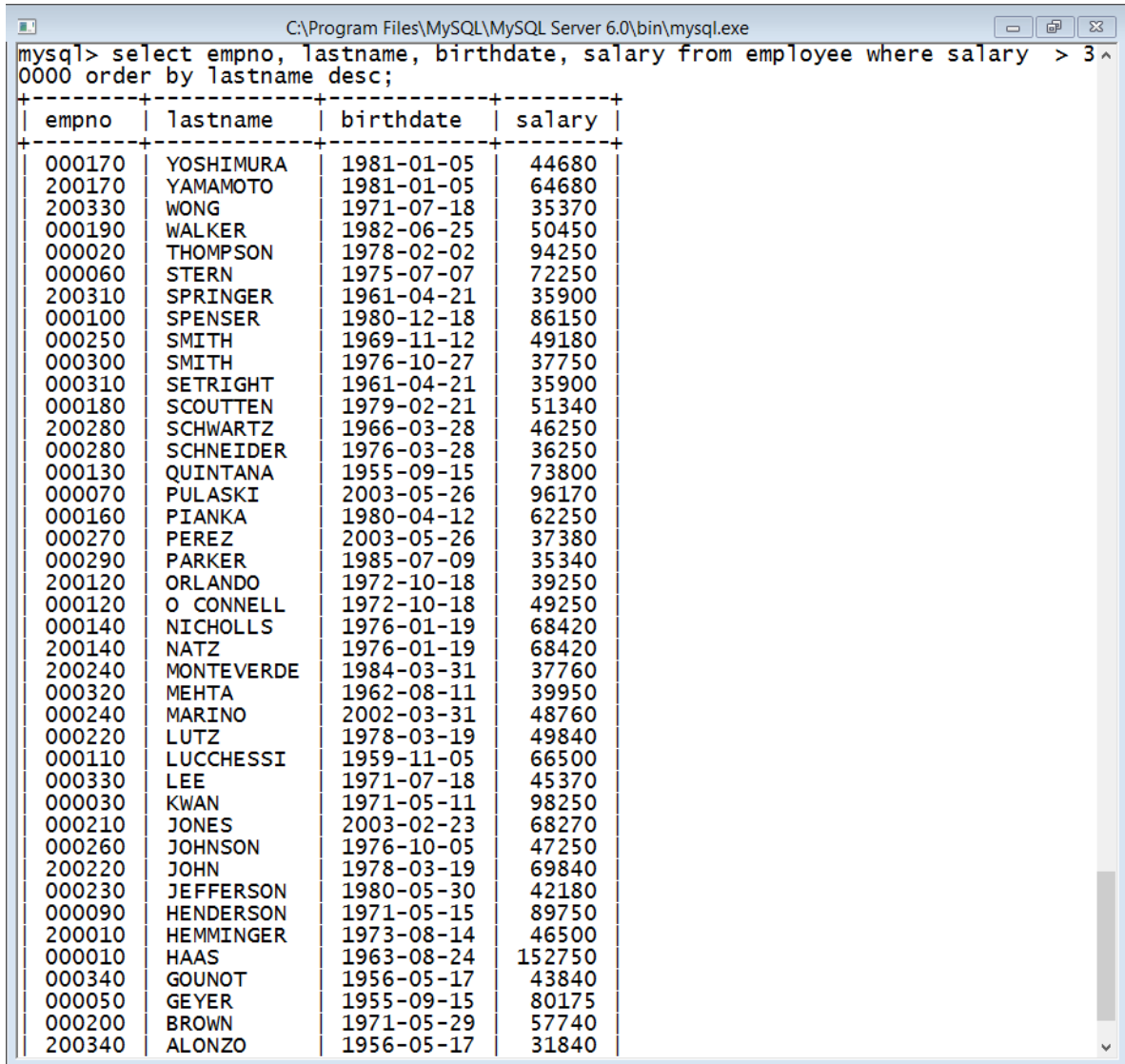


1. Mostrar el número de empleado, apellido, fecha de nacimiento y el salario de aquellos empleados que ganan más de USD \$30000 al año. Ordena los resultados de forma descendente.
2. Muestra el nombre, apellido, y el número de depto de todos los empleados. El resultado debería estar ordenando descendentemente por el número de departamento, para cada depto ordena descendentemente por el apellido.
3. Muestra los distintos niveles de educación ordenados descendentemente. Elimina tuplas duplicadas.
4. Muestra el número de empleado y el proyecto al cual se encuentra asignado, muestra aquellos empleados con un número menor o igual a 100. Elimina tuplas duplicadas, ordenando las tuplas resultantes por el número de empleados.
5. Muestra el apellido, el salario y el bono de todos los empleados varones.
6. Muestra el apellido, salario y comisión para todos los empleados con un salario menor que 20000 y la fecha de contratación después de 1979.
7. Muestra el apellido, salario, bono y comisión de todos los empleados con un salario mayor que 22000, un bono de 400, o que los empleados con un bono de 500 y una comisión menor a 1900. El resultado debe ser ordenado por el apellido.

8. Usando la relación empproject, para todos los empleados que tienen un número de proyecto que inician con AD y que tengan las actividades 10,80 o 180 asociadas con ellos. Muestra lo siguiente: Numero de proyecto, número de actividades, fecha de inicio y fecha de término para cada actividad. Ordena la lista por el número de actividad para cada número de proyecto.
9. Muestra el número del responsable y el número de departamento de todos los departamentos a los cuales un administrador ha sido asignado. La lista deberá ser ordenada por el número del responsable.
10. Muestra el número de empleado, el apellido, salario y bono de aquellos empleados que tienen un bono entre 800 y 1000. Ordena el reporte por el número de empleado con un bono menor, teniendo como primera tupla aquel que tiene el menor bono.
11. Muestra el número de empleado, apellido, salario y el número de departamento de aquellos empleados que están asignados en el departamento A00 al C01, ordenando los resultados alfabéticamente por el apellido y el número de empleado.
12. Muestra la información de los proyectos que tienen SUPPORT como parte del nombre del proyecto. Ordena el resultado por el número de proyecto.
13. Lista todos los departamentos que tienen un 1 como carácter intermedio en el número de depto., ordenando el resultado por el número de departamento.

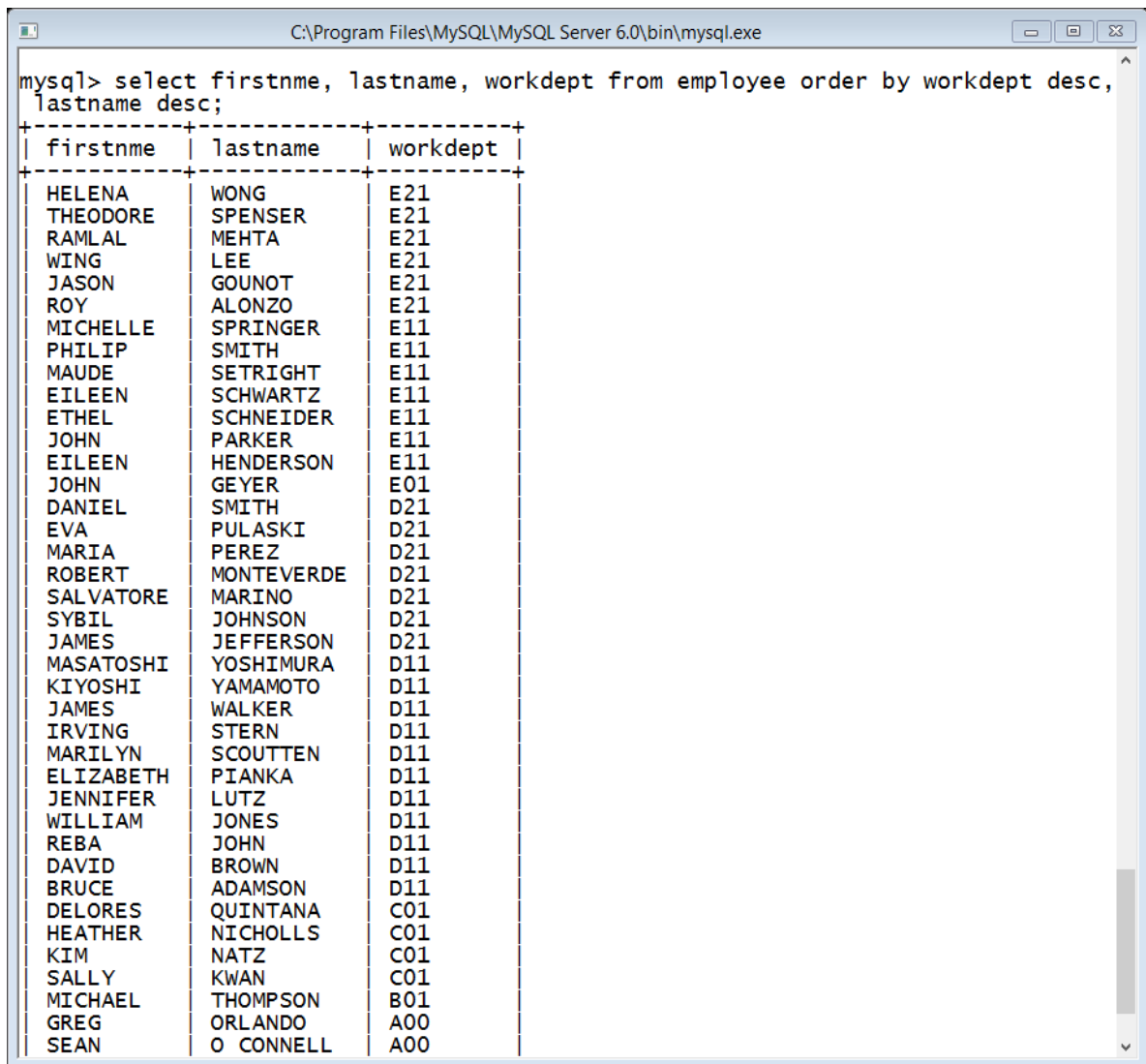
### 3. Desarrollo

1. *select empno, lastname, birthdate, salary from employee where salary>30000 order by lastname desc;*



empno	lastname	birthdate	salary
000170	YOSHIMURA	1981-01-05	44680
200170	YAMAMOTO	1981-01-05	64680
200330	WONG	1971-07-18	35370
000190	WALKER	1982-06-25	50450
000020	THOMPSON	1978-02-02	94250
000060	STERN	1975-07-07	72250
200310	SPRINGER	1961-04-21	35900
000100	SPENSER	1980-12-18	86150
000250	SMITH	1969-11-12	49180
000300	SMITH	1976-10-27	37750
000310	SETRIGHT	1961-04-21	35900
000180	SCOUTTEN	1979-02-21	51340
200280	SCHWARTZ	1966-03-28	46250
000280	SCHNEIDER	1976-03-28	36250
000130	QUINTANA	1955-09-15	73800
000070	PULASKI	2003-05-26	96170
000160	PIANKA	1980-04-12	62250
000270	PEREZ	2003-05-26	37380
000290	PARKER	1985-07-09	35340
200120	ORLANDO	1972-10-18	39250
000120	O'CONNELL	1972-10-18	49250
000140	NICHOLLS	1976-01-19	68420
200140	NATZ	1976-01-19	68420
200240	MONTEVERDE	1984-03-31	37760
000320	MEHTA	1962-08-11	39950
000240	MARINO	2002-03-31	48760
000220	LUTZ	1978-03-19	49840
000110	LUCCHESI	1959-11-05	66500
000330	LEE	1971-07-18	45370
000030	KWAN	1971-05-11	98250
000210	JONES	2003-02-23	68270
000260	JOHNSON	1976-10-05	47250
200220	JOHN	1978-03-19	69840
000230	JEFFERSON	1980-05-30	42180
000090	HENDERSON	1971-05-15	89750
200010	HEMMINGER	1973-08-14	46500
000010	HAAS	1963-08-24	152750
000340	GOUNOT	1956-05-17	43840
000050	GEYER	1955-09-15	80175
000200	BROWN	1971-05-29	57740
200340	ALONZO	1956-05-17	31840

2. *select firstnme, lastname, workdept from employee order by workdept desc, lastname desc;*

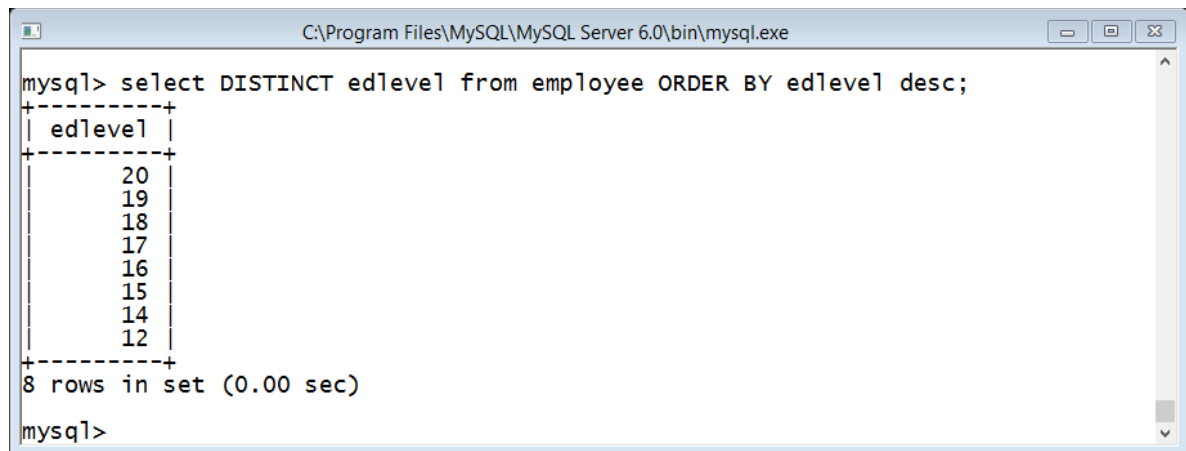


```
mysql> select firstnme, lastname, workdept from employee order by workdept desc,
lastname desc;
```

firstnme	lastname	workdept
HELENA	WONG	E21
THEODORE	SPENSER	E21
RAMLAL	MEHTA	E21
WING	LEE	E21
JASON	GOUNOT	E21
ROY	ALONZO	E21
MICHELLE	SPRINGER	E11
PHILIP	SMITH	E11
MAUDE	SETRIGHT	E11
EILEEN	SCHWARTZ	E11
ETHEL	SCHNEIDER	E11
JOHN	PARKER	E11
EILEEN	HENDERSON	E11
JOHN	GEYER	E01
DANIEL	SMITH	D21
EVA	PULASKI	D21
MARIA	PEREZ	D21
ROBERT	MONTEVERDE	D21
SALVATORE	MARINO	D21
SYBIL	JOHNSON	D21
JAMES	JEFFERSON	D21
MASATOSHI	YOSHIMURA	D11
KIYOSHI	YAMAMOTO	D11
JAMES	WALKER	D11
IRVING	STERN	D11
MARILYN	SCOUTTEN	D11
ELIZABETH	PIANKA	D11
JENNIFER	LUTZ	D11
WILLIAM	JONES	D11
REBA	JOHN	D11
DAVID	BROWN	D11
BRUCE	ADAMSON	D11
DELORES	QUINTANA	C01
HEATHER	NICHOLLS	C01
KIM	NATZ	C01
SALLY	KWAN	C01
MICHAEL	THOMPSON	B01
GREG	ORLANDO	A00
SEAN	O CONNELL	A00



3. *select DISTINCT edlevel from employee ORDER BY edlevel desc;*

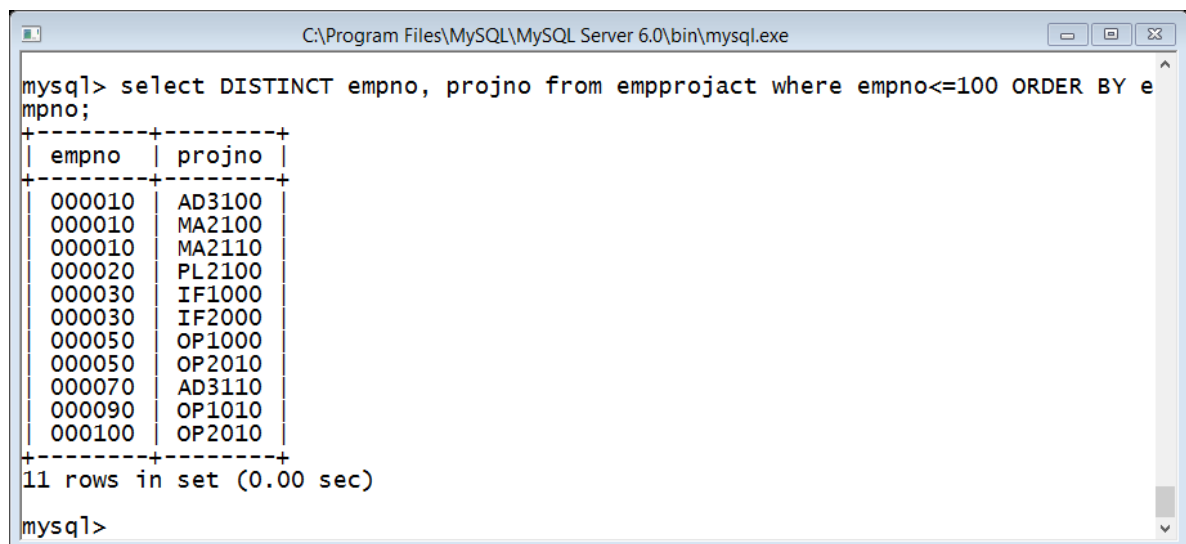


```
C:\Program Files\MySQL\MySQL Server 6.0\bin\mysql.exe

mysql> select DISTINCT edlevel from employee ORDER BY edlevel desc;
+-----+
| edlevel |
+-----+
|      20 |
|      19 |
|      18 |
|      17 |
|      16 |
|      15 |
|      14 |
|      12 |
+-----+
8 rows in set (0.00 sec)

mysql>
```

4. *select DISTINCT empno, projno from empproject where empno<=100 ORDER BY empno;*

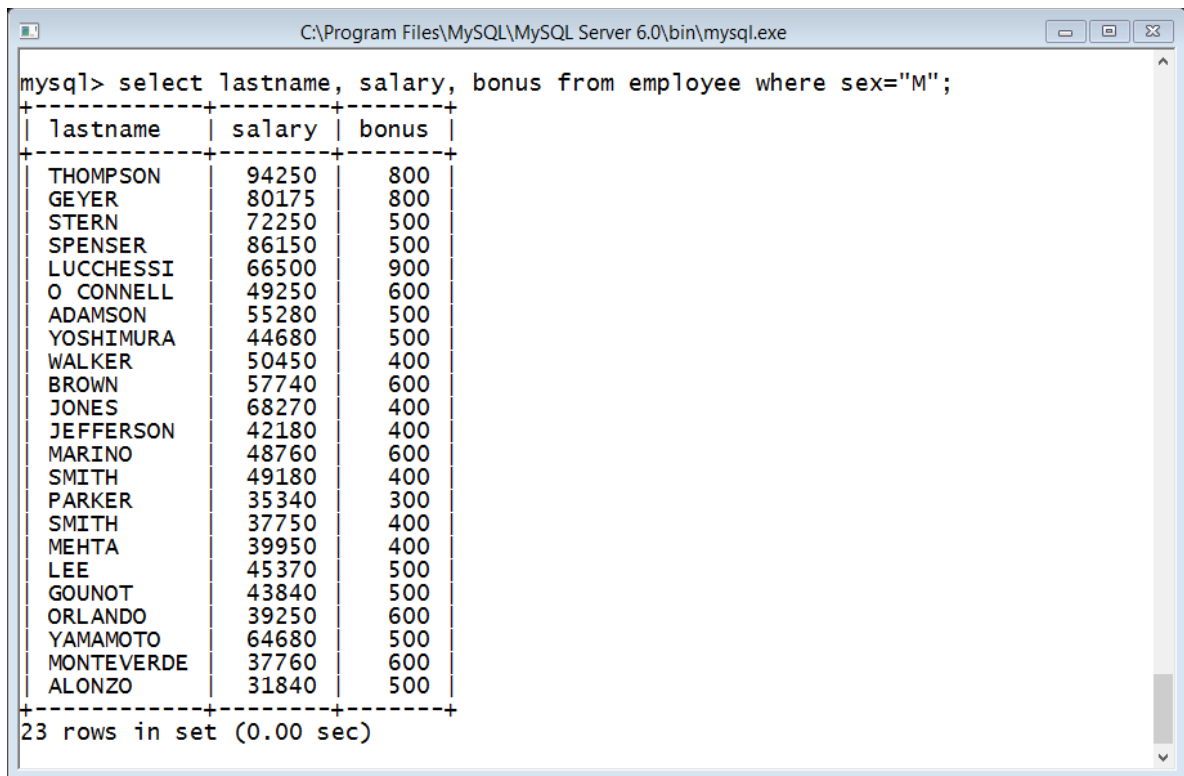


```
C:\Program Files\MySQL\MySQL Server 6.0\bin\mysql.exe

mysql> select DISTINCT empno, projno from empproject where empno<=100 ORDER BY empno;
+-----+-----+
| empno | projno |
+-----+-----+
| 000010 | AD3100 |
| 000010 | MA2100 |
| 000010 | MA2110 |
| 000020 | PL2100 |
| 000030 | IF1000 |
| 000030 | IF2000 |
| 000050 | OP1000 |
| 000050 | OP2010 |
| 000070 | AD3110 |
| 000090 | OP1010 |
| 000100 | OP2010 |
+-----+-----+
11 rows in set (0.00 sec)

mysql>
```

5. *select lastname, salary, bonus from employee where sex="M";*

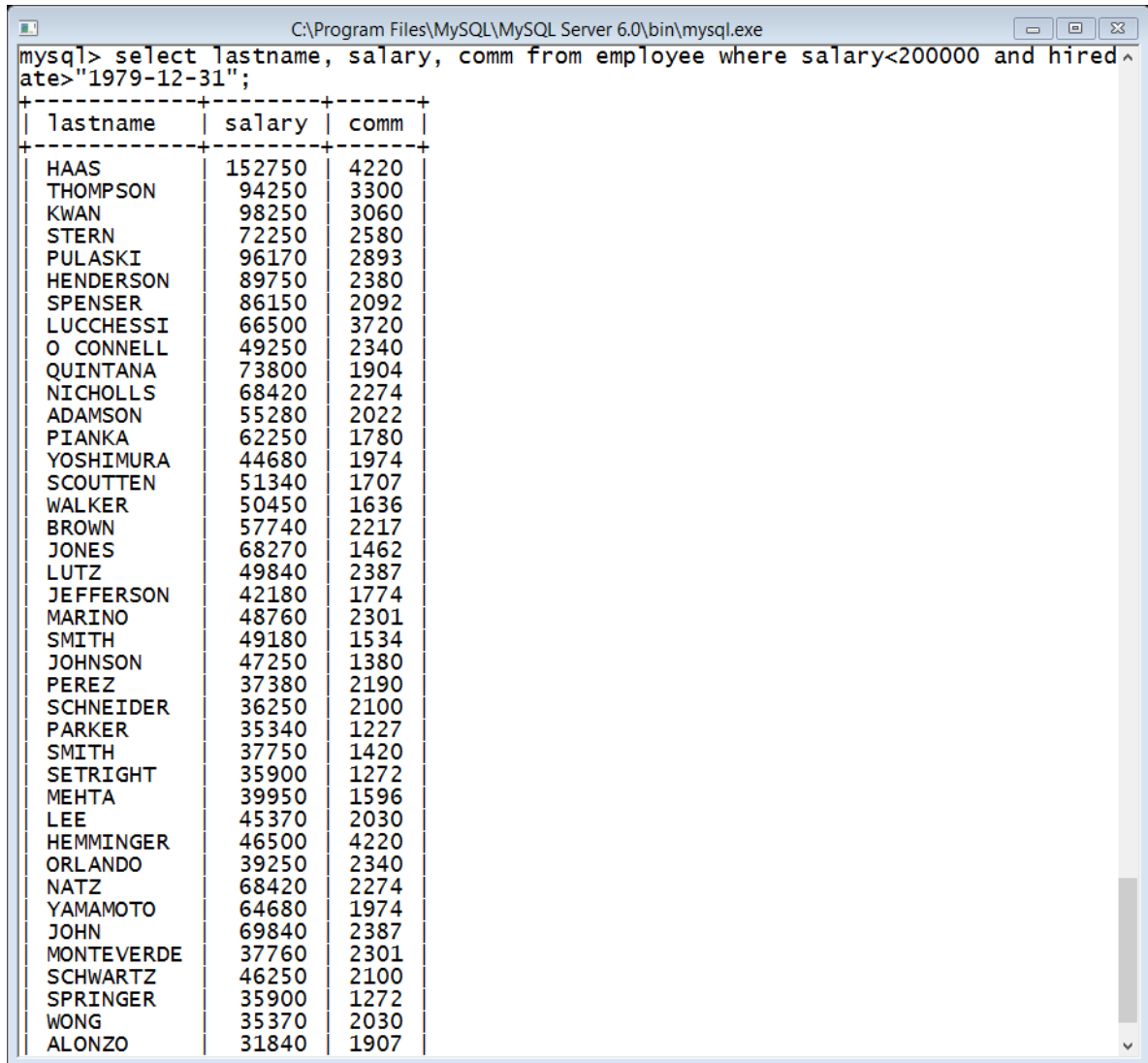


```
mysql> select lastname, salary, bonus from employee where sex="M";
```

lastname	salary	bonus
THOMPSON	94250	800
GEYER	80175	800
STERN	72250	500
SPENSER	86150	500
LUCCHESI	66500	900
O CONNELL	49250	600
ADAMSON	55280	500
YOSHIMURA	44680	500
WALKER	50450	400
BROWN	57740	600
JONES	68270	400
JEFFERSON	42180	400
MARINO	48760	600
SMITH	49180	400
PARKER	35340	300
SMITH	37750	400
MEHTA	39950	400
LEE	45370	500
GOUNOT	43840	500
ORLANDO	39250	600
YAMAMOTO	64680	500
MONTEVERDE	37760	600
ALONZO	31840	500

23 rows in set (0.00 sec)

6. *select lastname, salary, comm from employee where salary<200000 and hiredate>"1979-12-31";*



```
mysql> select lastname, salary, comm from employee where salary<200000 and hiredate>"1979-12-31";
```

lastname	salary	comm
HAAS	152750	4220
THOMPSON	94250	3300
KWAN	98250	3060
STERN	72250	2580
PULASKI	96170	2893
HENDERSON	89750	2380
SPENSER	86150	2092
LUCCHESSI	66500	3720
O CONNELL	49250	2340
QUINTANA	73800	1904
NICHOLLS	68420	2274
ADAMSON	55280	2022
PIANKA	62250	1780
YOSHIMURA	44680	1974
SCOUTTEN	51340	1707
WALKER	50450	1636
BROWN	57740	2217
JONES	68270	1462
LUTZ	49840	2387
JEFFERSON	42180	1774
MARINO	48760	2301
SMITH	49180	1534
JOHNSON	47250	1380
PEREZ	37380	2190
SCHNEIDER	36250	2100
PARKER	35340	1227
SMITH	37750	1420
SETRIGHT	35900	1272
MEHTA	39950	1596
LEE	45370	2030
HEMMINGER	46500	4220
ORLANDO	39250	2340
NATZ	68420	2274
YAMAMOTO	64680	1974
JOHN	69840	2387
MONTEVERDE	37760	2301
SCHWARTZ	46250	2100
SPRINGER	35900	1272
WONG	35370	2030
ALONZO	31840	1907

7. *select lastname, salary, bonus,comm from employee where salary>22000 and bonus in (400,500) and comm<1900 ORDER BY lastname;*

```
C:\Program Files\MySQL\MySQL Server 6.0\bin\mysql.exe

mysql> select lastname, salary, bonus,comm from employee where salary>22000 and
bonus in (400,500) and comm<1900 ORDER BY lastname;
+-----+-----+-----+-----+
| lastname | salary | bonus | comm |
+-----+-----+-----+-----+
| JEFFERSON | 42180 | 400 | 1774 |
| JONES | 68270 | 400 | 1462 |
| MEHTA | 39950 | 400 | 1596 |
| PIANKA | 62250 | 400 | 1780 |
| SCOUTTEN | 51340 | 500 | 1707 |
| SMITH | 49180 | 400 | 1534 |
| SMITH | 37750 | 400 | 1420 |
| WALKER | 50450 | 400 | 1636 |
+-----+-----+-----+-----+
8 rows in set (0.00 sec)

mysql>
```

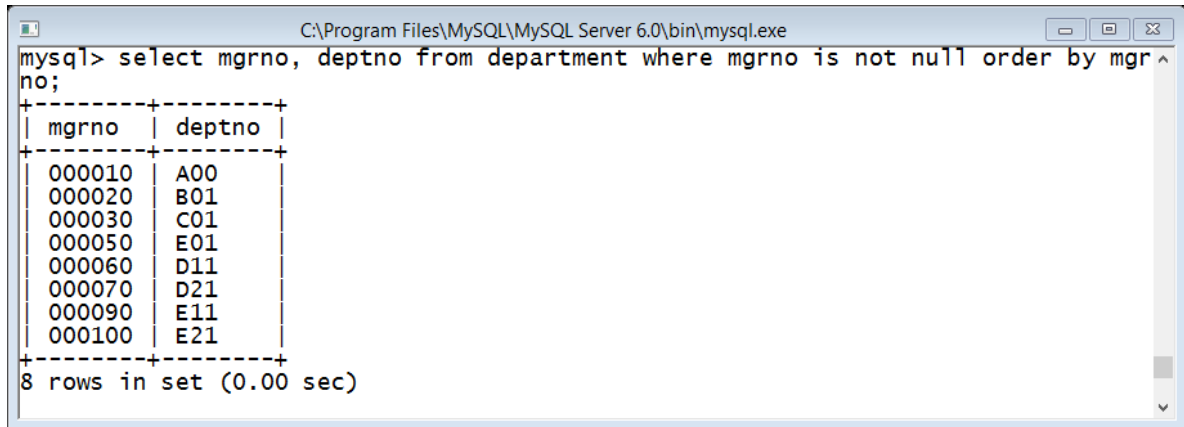
8. *select projno, actno, emstdate, emendate from empproject where projno like "AD%" and actno in(10,80,180) order by actno, projno;*

```
C:\Program Files\MySQL\MySQL Server 6.0\bin\mysql.exe

mysql> select projno, actno, emstdate, emendate from empproject where
-> projno like "AD%" and actno in(10,80,180) order by actno, projno;
+-----+-----+-----+-----+
| projno | actno | emstdate | emendate |
+-----+-----+-----+-----+
| AD3100 | 10 | 2002-01-01 | 2002-07-01 |
| AD3110 | 10 | 2002-01-01 | 2003-02-01 |
| AD3111 | 80 | 2002-04-15 | 2002-10-15 |
| AD3111 | 80 | 2002-09-15 | 2003-01-01 |
| AD3112 | 80 | 2002-08-15 | 2002-10-15 |
| AD3112 | 80 | 2002-10-15 | 2002-12-01 |
| AD3113 | 80 | 2002-01-01 | 2002-03-01 |
| AD3113 | 80 | 2002-03-01 | 2002-04-15 |
| AD3113 | 80 | 2002-01-01 | 2002-03-01 |
| AD3113 | 80 | 2002-03-01 | 2002-04-01 |
| AD3111 | 180 | 2002-10-15 | 2003-01-01 |
| AD3112 | 180 | 2002-08-15 | 2003-01-01 |
| AD3113 | 180 | 2002-03-01 | 2002-04-15 |
| AD3113 | 180 | 2002-04-15 | 2002-06-01 |
| AD3113 | 180 | 2002-06-01 | 2002-07-01 |
+-----+-----+-----+-----+
15 rows in set (0.00 sec)

mysql>
```

9. *select mgrno, deptno from department where mgrno is not null order by mgrno;*

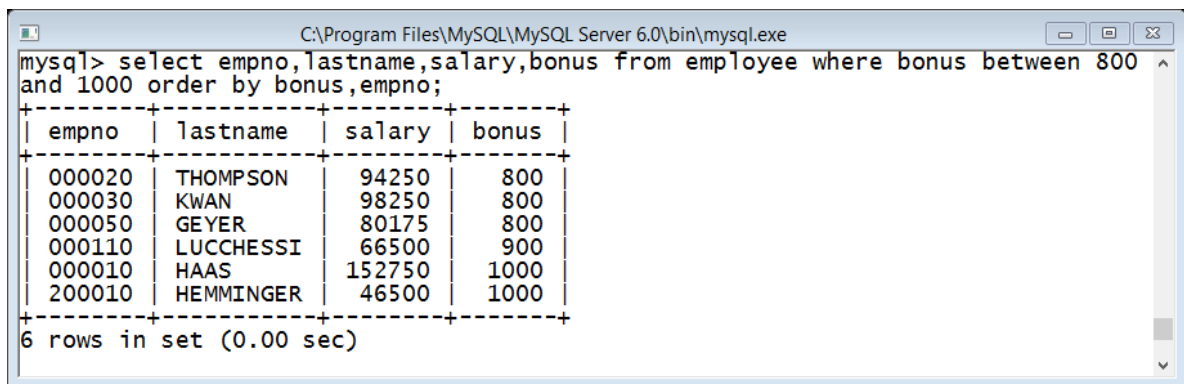


```
mysql> select mgrno, deptno from department where mgrno is not null order by mgrno;
```

mgrno	deptno
000010	A00
000020	B01
000030	C01
000050	E01
000060	D11
000070	D21
000090	E11
000100	E21

8 rows in set (0.00 sec)

10. *select empno,lastname,salary,bonus from employee where bonus between 800 and 1000 order by bonus,empno;*

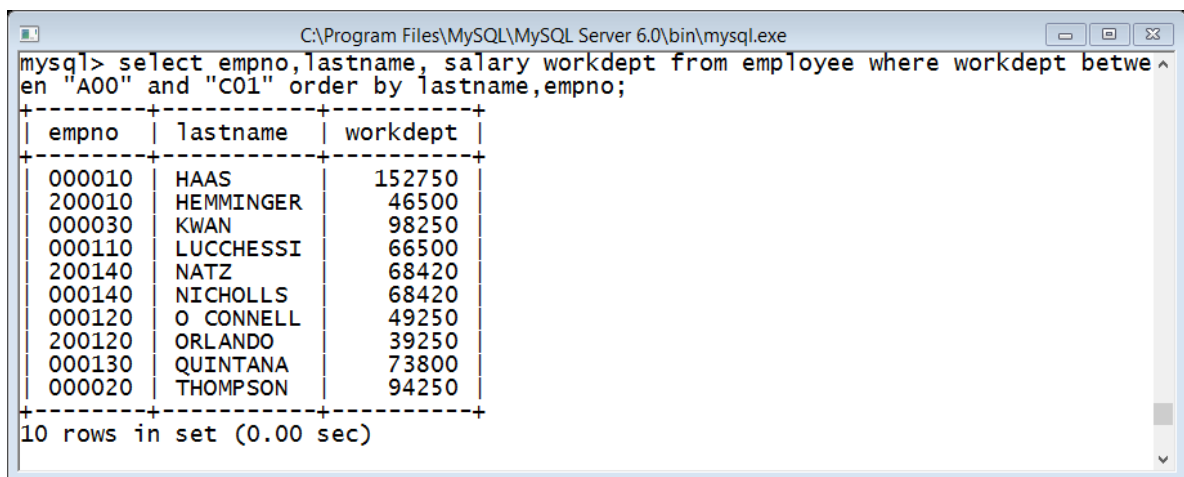


```
mysql> select empno,lastname,salary,bonus from employee where bonus between 800 and 1000 order by bonus,empno;
```

empno	lastname	salary	bonus
000020	THOMPSON	94250	800
000030	KWAN	98250	800
000050	GEYER	80175	800
000110	LUCCHESI	66500	900
000010	HAAS	152750	1000
200010	HEMMINGER	46500	1000

6 rows in set (0.00 sec)

11. *select empno,lastname, salary workdept from employee where workdept between "A00" and "C01" order by lastname,empno;*

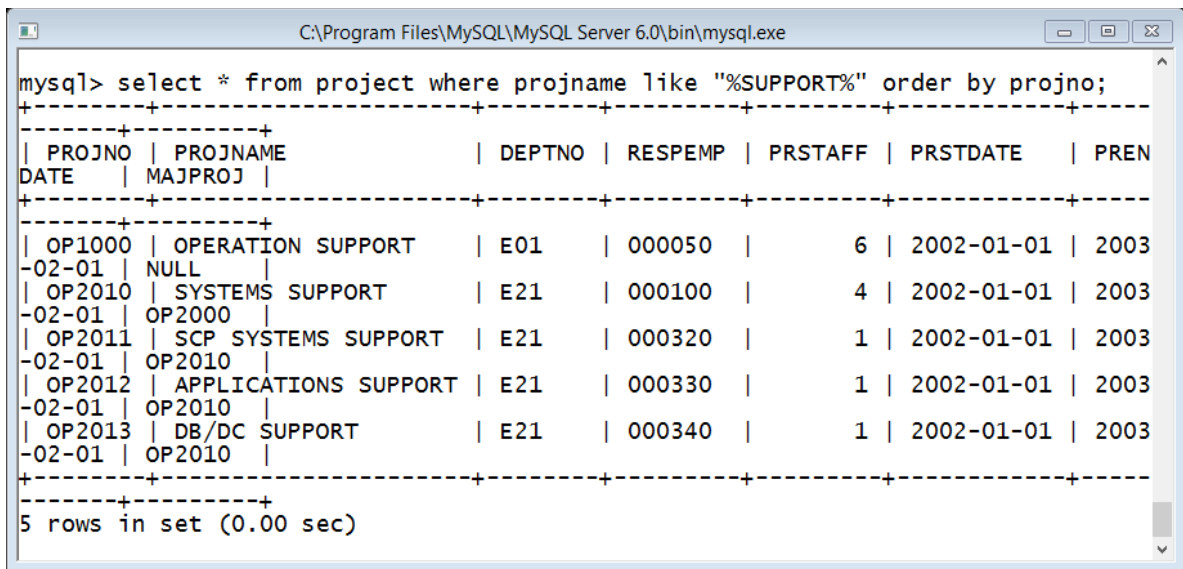


```
mysql> select empno,lastname, salary workdept from employee where workdept between "A00" and "C01" order by lastname,empno;
```

empno	lastname	workdept
000010	HAAS	152750
200010	HEMMINGER	46500
000030	KWAN	98250
000110	LUCCHESI	66500
200140	NATZ	68420
000140	NICHOLLS	68420
000120	O CONNELL	49250
200120	ORLANDO	39250
000130	QUINTANA	73800
000020	THOMPSON	94250

10 rows in set (0.00 sec)

12. *select \* from project where projname like "%SUPPORT%" order by projno;*

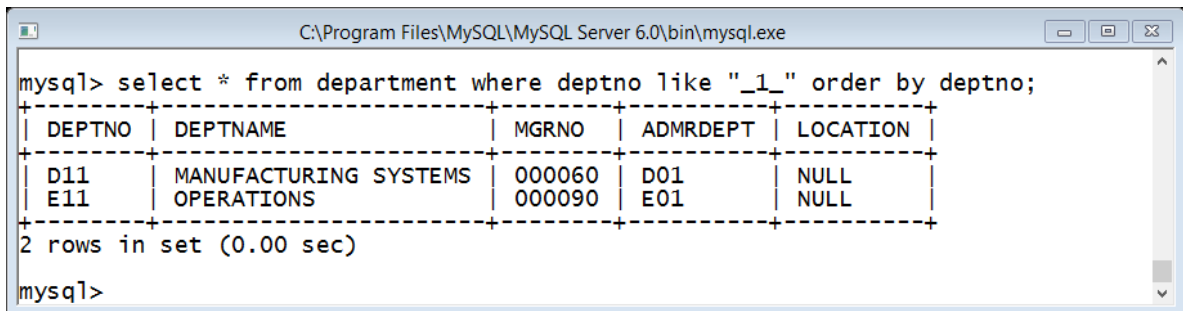


```
mysql> select * from project where projname like "%SUPPORT%" order by projno;
```

PROJNO	PROJNAME	DEPTNO	RESPEMP	PRSTAFF	PRSTDATE	PREN
OP1000	OPERATION SUPPORT	E01	000050	6	2002-01-01	2003
-02-01	NULL					
OP2010	SYSTEMS SUPPORT	E21	000100	4	2002-01-01	2003
-02-01	OP2000					
OP2011	SCP SYSTEMS SUPPORT	E21	000320	1	2002-01-01	2003
-02-01	OP2010					
OP2012	APPLICATIONS SUPPORT	E21	000330	1	2002-01-01	2003
-02-01	OP2010					
OP2013	DB/DC SUPPORT	E21	000340	1	2002-01-01	2003
-02-01	OP2010					

5 rows in set (0.00 sec)

13. *select \* from department where deptno like "\_1\_" order by deptno;*



```
mysql> select * from department where deptno like "_1_" order by deptno;
```

DEPTNO	DEPTNAME	MGRNO	ADMRDEPT	LOCATION
D11	MANUFACTURING SYSTEMS	000060	D01	NULL
E11	OPERATIONS	000090	E01	NULL

2 rows in set (0.00 sec)

```
mysql>
```

## 4. Conclusión

Estas consultas fueron un poco más difíciles ya que pedían demasiadas cosas y en algunas partes no era muy claro cómo se querían ordenar los datos obtenidos, sin embargo nos ayudó a practicar más nuestros comandos de búsqueda y a interpretar mejor lo que se nos está pidiendo, al entender bien que se nos solicita podemos mejorar y optimizar nuestras consultas.

## 5. Bibliografía

1. 13.2.7. *Sintaxis de SELECT*. Recuperado 01 de septiembre de 2013.  
[En línea]. Disponible en:  
<http://dev.mysql.com/doc/refman/5.0/es/select.html>.
2. 13.2.4. *Sintaxis de INSERT*. Recuperado 01 de septiembre de 2013.  
[En línea]. Disponible en:  
<http://dev.mysql.com/doc/refman/5.0/es/insert.html>.