



Universidad Autónoma de Querétaro  
Facultad de Informática  
Taller Oracle 11g



**Instrucciones:** Cree las siguientes consultas. Adjunte, para cada ejercicio, la captura de pantalla del resultado.

1. Muestre el apellido de cada empleado, así como la fecha de contratación y la fecha de revisión de salario, que es el primer lunes después de seis meses de servicio. Etiquete la columna REVIEW. Formatee las fechas para que aparezca en un formato similar a "Monday, the Thirtyfirst of July, 2000".

```
SELECT last_name, hire_date,  
       TO_CHAR(NEXT_DAY(ADD_MONTHS(hire_date, 6), 'LUNES'),  
               'fmDay, "the" Ddspth "of" Month, YYYY') REVIEW  
FROM employees;
```

	LAST_NAME	HIRE_DATE	REVIEW
1	King	17/06/87	LUNES , the Twenty-First of DICIEMBRE, 1987
2	De Haan	21/09/89	LUNES , the Twenty-Sixth of MARZO, 1990
3	De Haan	13/01/93	LUNES , the Nineteenth of JULIO, 1993
4	Ronald	03/01/90	LUNES , the Ninth of JULIO, 1990
5	Ernat	21/05/91	LUNES , the Twenty-Fifth of NOVIEMBRE, 1991
6	Austin	28/06/97	LUNES , the Twenty-Sixth of DICIEMBRE, 1997
7	Peraballa	08/02/98	LUNES , the Tenth of AGOSTO, 1998
8	Loents	07/02/99	LUNES , the Ninth of AGOSTO, 1999
9	Greenberg	17/08/94	LUNES , the Twentieth of FEBRERO, 1995
10	Faviet	16/08/94	LUNES , the Twentieth of FEBRERO, 1995
11	Chen	28/09/97	LUNES , the Thirtieth of MARZO, 1998
12	Sciarra	30/09/97	LUNES , the Ninth of ABRIL, 1998
13	Uman	07/03/98	LUNES , the Fourteenth of SEPTIEMBRE, 1998
14	Popp	07/12/99	LUNES , the Twelfth of JUNIO, 2000
15	Raghealy	07/12/94	LUNES , the Twelfth of JUNIO, 1995
16	Shoo	18/05/98	LUNES , the Twentieth of NOVIEMBRE, 1998
17	Deida	24/12/97	LUNES , the Twenty-Ninth of JUNIO, 1998
18	Tobias	24/07/97	LUNES , the Twenty-Sixth of ENERO, 1998

2. Muestre el apellido, la fecha de contratación y el día de la semana en el que comenzó a trabajar el empleado. Etiquete a la última columna DAY. Ordene los resultados por la última columna.

```
SELECT last_name, hire_date, TO_CHAR(hire_date, 'DAY') DAY  
FROM employees  
ORDER BY TO_CHAR(hire_date, 'd') ASC;
```

	LAST_NAME	HIRE_DATE	DAY
1	Kaufling	01/05/95	LUNES
2	OConnell	21/06/99	LUNES
3	Fatel	06/04/98	LUNES
4	Ernszt	10/03/97	LUNES
5	Bernstein	24/03/97	LUNES
6	Olsen	30/03/98	LUNES
7	Sully	04/03/94	LUNES
8	Smith	10/03/97	LUNES
9	Doran	15/12/97	LUNES
10	Marvins	24/01/00	LUNES
11	Bloom	23/03/98	LUNES
12	Grant	24/08/99	LUNES
13	Fleaur	23/02/98	LUNES
14	Sullivan	21/06/99	LUNES
15	Everett	03/03/97	LUNES
16	Mikkilineni	28/09/98	LUNES
17	Ernat	21/05/91	VIERNES
18	Faviet	16/08/94	VIERNES



**Universidad Autónoma de Querétaro**  
**Facultad de Informática**  
**Taller Oracle 11g**



3. Cree una consulta que muestre el apellido y las comisiones de los empleados. Si un empleado no gana comisión, ponga "Sin Comisión". Etiquete a la última columna COMM.

```
SELECT last_name, NVL(TO_CHAR(commission_pct), 'Sin Comisión') COMM  
FROM employees;
```

	LAST_NAME	COMM
1	Kling	Sin Comisión
2	Rothbar	Sin Comisión
3	De Haan	Sin Comisión
4	Ronald	Sin Comisión
5	Ernat	Sin Comisión
6	Austin	Sin Comisión
7	Pataballa	Sin Comisión
8	Lorentz	Sin Comisión
9	Greenberg	Sin Comisión
10	Faviet	Sin Comisión
11	Ozer	Sin Comisión
12	Sciarra	Sin Comisión
13	Urman	Sin Comisión
14	Popp	Sin Comisión
15	Raphaely	Sin Comisión
16	Rhoo	Sin Comisión
17	Baida	Sin Comisión
18	Tobias	Sin Comisión

4. Cree una consulta que muestre el apellido de los empleados y que indique las cantidades de sus salarios con asteriscos. Cada asterisco significa mil dólares. Ordene los datos por salario en orden descendente. Etiquete la columna EMPLOYEES\_AND\_THEIR\_SALARIES.

```
SELECT last_name, salary, RPAD('*', TRUNC(salary/1000), '*')  
EMPLOYEES_AND_THEIR_SALARIES  
FROM employees  
ORDER BY EMPLOYEES_AND_THEIR_SALARIES DESC;
```

	LAST_NAME	SALARY	EMPLOYEES_AND_THEIR_SALARIES
1	Kling	24000	*****
2	Rothbar	17000	*****
3	De Haan	17000	*****
4	Russell	14000	*****
5	Partners	13500	*****
6	Bartstein	13000	*****
7	Higgins	12000	*****
8	Erasmus	12000	*****
9	Greenberg	12000	*****
10	Cambrault	11000	*****
11	Abel	11000	*****
12	Ozer	11500	*****
13	Raphaely	11000	*****
14	Kling	10000	*****
15	Tucker	10000	*****
16	Elstney	10500	*****
17	Baer	10000	*****
18	Bloom	10000	*****

5. Usando la función DECODE, escribir una consulta que presente los grados de todos los empleados basado en los valores de la columna JOB\_ID, usando los siguientes datos:



**Universidad Autónoma de Querétaro**  
**Facultad de Informática**  
**Taller Oracle 11g**



JOB_ID	GRADO
AD_PRES	A
ST_MAN	B
IT_PROG	C
SA_REP	D
ST_CLERK	E
Ninguno de los anteriores	F

```
SELECT job_id,  
       DECODE (job_id,  
               'AD_PRES','A',  
               'ST_MAN','B',  
               'IT_PROG','C',  
               'SA_REP','D',  
               'ST_CLERK','E',  
               'F') AS GRADO  
FROM employees;
```



Universidad Autónoma de Querétaro  
Facultad de Informática  
Taller Oracle 11g



Resultado de la Consulta x

SQL | Se han r

JOB_ID	GRADO
1 AC_ACCOUNT	F
2 AC_MGR	F
3 AD_ASST	F
4 AD_PRES	A
5 AD_VP	F
6 AD_VP	F
7 FI_ACCOUNT	F
8 FI_ACCOUNT	F
9 FI_ACCOUNT	F
10 FI_ACCOUNT	F
11 FI_ACCOUNT	F
12 FI_MGR	F
13 HR_REP	F
14 IT_PROG	C
15 IT_PROG	C
16 IT_PROG	C
17 IT_PROG	C
18 IT_PROG	C
19 IT_PROG	C

6. Resuelva el ejercicio anterior utilizando la sintaxis de la función CASE.

```
SELECT job_id,  
       CASE job_id  
         WHEN 'AD_PRES' THEN 'A'  
         WHEN 'ST_MAN' THEN 'B'  
         WHEN 'IT_PROG' THEN 'C'  
         WHEN 'SA_REP' THEN 'D'  
         WHEN 'ST_CLERK' THEN 'E'  
         ELSE 'F'  
       END AS "GRADO"  
FROM employees;
```



Universidad Autónoma de Querétaro  
Facultad de Informática  
Taller Oracle 11g



Resultado de la Consulta x		
SQL   Se han r		
JOB_ID	GRADO	
1 AC_ACCOUNT	F	
2 AC_MGR	F	
3 AD_ASST	F	
4 AD_PRES	A	
5 AD_VP	F	
6 AD_VP	F	
7 FI_ACCOUNT	F	
8 FI_ACCOUNT	F	
9 FI_ACCOUNT	F	
10 FI_ACCOUNT	F	
11 FI_ACCOUNT	F	
12 FI_MGR	F	
13 HR_REP	F	
14 IT_PROG	C	
15 IT_PROG	C	
16 IT_PROG	C	
17 IT_PROG	C	
18 IT_PROG	C	
19 MK_MAN	F	