**TEMA 3**

**CASO PRÁCTICO 3** (pág. 55)

*Seleccionamos de la tabla EMPLE a todos los empleados del departamento 20 (DEPT\_NO = 20). Además, la consulta debe aparecer ordenada por la columna APELLIDO. Los campos que hay que consultar son: número de empleado, apellido, oficio y número de departamento.*

**SELECT** EMP\_NO, APELLIDO, OFICIO, DEPT\_NO **FROM** EMPLE **WHERE** DEPT\_NO = 20 ORDER BY APELLIDO;

*Consulta de los empleados cuyo oficio sea ‘ANALISTA’ ordenador por númeor de empleado (columna EMP\_NO).*

**SELECT** \* **FROM** EMPLE **WHERE** OFICIO = 'ANALISTA' **ORDER BY** EMP\_NO;

*Seleccionar de la tabla EMPLE aquellas filas del departamento 10 cuyo oficio sea ‘ANALISTA’. La consulta se ha de ordenar de modo descendente por APELLIDO y también de manera descendente por número de empleado (columna EMP\_NO).*

**SELECT** \* **FROM** EMPLE **WHERE** DEPT\_NO = 10 **AND** OFICIO = 'ANALISTA' **ORDER BY** APELLIDO DESC, EMP\_NO DESC;

**ACTIVIDAD PROPUESTA 3** (pág. 56)

*A partir de la tabla ALUM0405 que contiene los datos de los alumnos matriculados en el curso 2004/2005 para un centro de enseñanza, obtén los siguientes datos:*

*· Todos los datos de los alumnos*

**SELECT** \* **FROM** ALUM0405;

*· DNI, nombre, apellidos, curso, nivel y clase*

**SELECT** DNI, NOMBRE, APELLIDOS, CURSO, NIVEL, CLASE **FROM** ALUM0405;

*· Todos los datos de los alumnos cuya población sea ‘GUADALAJARA’*

**SELECT** \* **FROM** ALUM0405 **WHERE** POBLACION = 'GUADALAJARA';

*· NOMBRE y APELLIDOS de todos los alumnos cuya población sea ‘GUADALAJARA’*

**SELECT** NOMBRE, APELLIDOS **FROM** ALUM0405 **WHERE** POBLACION = 'GUADALAJARA';

*· DNI, NOMBRE, APELLIDOS, CURSO, NIVEL Y CLASE de todos los alumnos ordenador por APELLIDOS y NOMBRE ascendentemente.*

**SELECT** DNI, NOMBRE, APELLIDOS, CURSO, NIVEL, CLASE **FROM** ALUM0405 **ORDER BY** APELLIDOS, NOMBRE;

**CASO PRÁCTICO 4** (pág. 57)

*Utilizamos alias para las columnas DEPT\_NO y DNOMBRE.*

**SELECT** DNOMBRE "Departamento", DEPT\_NO "Número Departamento" **FROM** DEPART;

**CASO PRÁCTICO 5** (pág. 57)

*Disponemos de la tabla NOTAS\_ALUMNOS, que contiene las notas de los alumnos de primer curso de ciclo DAI obtenidas en los tres módulos. Queremos obtener la nota media de cada alumno. Visualizamos por cada uno de ellos su nombre y su nota media.*

**SELECT** NOMBRE\_ALUMNO "Nombre Alumno", (NOTA1+NOTA2+NOTA3)/3 "Nota Media" **FROM** NOTAS\_ALUMNOS;

**CASO PRÁCTICO 6** (pág. 59)

*A partir de la tabla NOTAS\_ALUMNOS deseamos obtener aquellos nombres de alumnos que tengan un 7 en NOTA1 y cuya media sea mayor que 6.*

**SELECT** NOMBRE\_ALUMNO **FROM** NOTAS\_ALUMNOS **WHERE** NOTA1=7 **AND** (NOTA1+NOTA2+NOTA3)/3 > 6;

**CASO PRÁCTICO 7** (pág. 60)

*A partir de la tabla EMPLE, obtén aquellos apellidos que empiecen por una 'J':*

**SELECT** APELLIDO **FROM** EMPLE **WHERE** APELLIDO **LIKE** 'J%';

*Obtén aquellos apellidos que tengan una 'R' en la segunda posición:*

**SELECT** APELLIDO **FROM** EMPLE **WHERE** APELLIDO **LIKE** '\_R%';

*Obtén aquellos apellidos que empiecen por 'A' y tengan una 'O' en su interior:*

**SELECT** APELLIDO **FROM** EMPLE **WHERE** APELLIDO **LIKE** 'A%O%';

*Consideramos la tabla LIBRERÍA. Consultamos las filas cuyo tema sea 'LABORES':*

**SELECT** \* **FROM** LIBRERIA **WHERE** TEMA **LIKE** 'LABORES%';

**CASO PRÁCTICO 8** (pág. 61)

*Consulta los apellidos de la tabla EMPLE cuyo número de departamento sea 10 o 30.*

**SELECT** APELLIDO **FROM** EMPLE **WHERE** DEPT\_NO **IN** (10, 30);

*Consula los apellidos de la tabla EMPLE cuyo número de departamento no sea ni 10 ni 30.*

**SELECT** APELLIDO **FROM** EMPLE **WHERE** DEPT\_NO **NOT IN** (10, 30);

*Consulta los apellidos de la tabla EMPLE cuyo oficio sea 'VENDEDOR', 'ANALISTA' o 'EMPLEADO'.*

**SELECT** APELLIDO **FROM** EMPLE **WHERE** OFICIO **IN** ('VENDEDOR', 'ANALISTA', 'EMPLEADO');

*Consulta los apellidos de la tabla EMPLE cuyo oficio no sea ni 'VENDEDOR' ni 'ANALISTA' ni 'EMPLEADO'.*

**SELECT** APELLIDO **FROM** EMPLE **WHERE** OFICIO **NOT IN** ('VENDEDOR', 'ANALISTA', 'EMPLEADO');

**CASO PRÁCTICO 9** (pág. 63)

*A partir de la tabla EMPLE, obtén el apellido y el salario de los empleados cuyo salario esté comprendido entre 1500 y 2000:*

**SELECT** APELLIDO, SALARIO **FROM** EMPLE **WHERE** SALARIO **BETWEEN** 1500 **AND** 2000;

*A partir de la tabla EMPLE, obtén el apellido y el salario de los empleados cuyo salario no esté comprendido entre 1500 y 2000:*

**SELECT** APELLIDO, SALARIO **FROM** EMPLE **WHERE** SALARIO **NOT BETWEEN** 1500 **AND** 2000;

**CASO PRÁCTICO 10** (pág. 64)

*A partir de la tabla EMPLE, obtén el APELLIDO, SALARIO Y DEPT\_NO de los empleados cuyo salario sea mayor de 2000 en los departamentos 10 o 20:*

**SELECT** APELLIDO, SALARIO, DEPT\_NO **FROM** EMPLE **WHERE** SALARIO > 2000 **AND** ( DEPT\_NO = 10 **OR** DEPT\_NO = 20);

**SELECT** APELLIDO, SALARIO, DEPT\_NO **FROM** EMPLE **WHERE** SALARIO > 2000 **AND** DEPT\_NO **IN** (10, 20);

**CASO PRÁCTICO 11** (pág. 65)

*Con la tabla EMPLE, obtén el APELLIDO de los empleados con el mismo OFICIO que 'GIL'.*

**SELECT** APELLIDO **FROM** EMPLE **WHERE** OFICIO = (**SELECT** OFICIO **FROM** EMPLE **WHERE** APELLIDO = 'GIL');

**ACTIVIDAD PROPUESTA 4** (pág. 66)

*Muestra los datos (apellido, oficio, salario y fecha de alta) de aquellos empleados que desempeñen el mismo oficio que "JIMENEZ" o que tengan un salario mayor o igual que "FERNANDEZ".*

**SELECT** APELLIDO, OFICIO, SALARIO, FECHA\_ALT **FROM** EMPLE **WHERE** OFICIO = (**SELECT** OFICIO **FROM** EMPLE **WHERE** APELLIDO = "JIMENEZ") **OR** SALARIO >= (**SELECT** SALARIO **FROM** EMPLE **WHERE** APELLIDO = "FERNANDEZ");

**CASO PRÁCTICO 12** (pág. 68)

*Tablas EMPLE y DEPART: datos de los empleados que trabajen en 'MADRID' o 'BARCELONA'*

**SELECT** \* **FROM** EMPLE **WHERE** DEPT\_NO **IN** (**SELECT** DEPT\_NO **FROM** DEPART **WHERE** LOC **IN** ('MADRID', 'BARCELONA'));

*Consulta los apellidos y oficios de todos los empleados del departamento 20 cuyo trabajo sea idéntico al de cualquiera de los empleados del departamento 'VENTAS':*

**SELECT** APELLIDO, OFICIO **FROM** EMPLE **WHERE** DEPT\_NO = 20 **AND** OFICIO **IN** (**SELECT** OFICIO **FROM** EMPLE **WHERE** DEPT\_NO = (**SELECT** DEPT\_NO **FROM** DEPART **WHERE** DNOMBRE = 'VENTAS')) ;

*Obtén el apellido de los empleados con el mismo oficio y salario que 'GIL'.*

**SELECT** APELLIDO **FROM** EMPLE **WHERE** (OFICIO, SALARIO) = (**SELECT** OFICIO, SALARIO **FROM** EMPLE **WHERE** APELLIDO = 'GIL');

**ACTIVIDAD PROPUESTA 5** (pág. 69)

*Presenta los apellidos y oficios de los empleados que tienen el mismo trabajo que "JIMENEZ".*

**SELECT** APELLIDO, OFICIO **FROM** EMPLE **WHERE** OFICIO = (**SELECT** OFICIO **FROM** EMPLE **WHERE** APELLIDO = "JIMENEZ");

*Muestra el APELLIDO, OFICIO y SALARIO de los empleados del departamento de "FERNANDEZ" que tengan su mismo salario.*

**SELECT** APELLIDO, OFICIO, SALARIO **FROM** EMPLE **WHERE** (DPTO\_NO, SALARIO) = (**SELECT** DPTO\_NO, SALARIO **FROM** EMPLE **WHERE** APELLIDO = "FERNANDEZ");

**CASO PRÁCTICO 13** (pág. 70)

*Obtener de las tablas EMPLE y DEPART los datos de APELLIDO, OFICIO, número de empleado (EMP\_NO), nombre de departamento (DNOMBRE) y localidad (LOC) de los empleados. Estas tablas tienen en común el campo DEPT\_NO.*

**SELECT** APELLIDO, OFICIO, EMP\_NO, DNOMBRE, LOC **FROM** EMPLE, DEPART **WHERE** EMPLE.DEPT\_NO = DEPART.DEPT\_NO;

*Dadas las tablas ALUMNOS, ASIGNATURAS Y NOTAS, relacionadas mediante los campos DNI (NOTAS-ALUMNOS) y COD (NOTAS-ASIGNATURAS), obtener el nombre de alumno (APENOM), asignatura (NOMBRE) y nota (NOTA).*

**SELECT** APENOM, NOMBRE, NOTA **FROM** ALUMNOS, ASIGNATURAS, NOTAS **WHERE** (NOTAS.DNI = ALUMNOS.DNI) **AND** (NOTAS.COD = ASIGNATURAS.COD);

*Obtén los nombres de los alumnos matriculados en FOL.*

**SELECT** APENOM **FROM** ALUMNOS, ASIGNATURAS, NOTAS **WHERE** (ALUMNOS.DNI = NOTAS.DNI) **AND** NOTAS.COD = (**SELECT** ASIGNATURAS.COD **WHERE** ASIGNATURAS.NOMBRE = 'FOL');

**SELECT** APENOM **FROM** ALUMNOS, ASIGNATURAS, NOTAS **WHERE** ALUMNOS.DNI = NOTAS.DNI **AND** NOTAS.COD = ASIGNATURAS.COD **AND** NOMBRE = 'FOL';

**ACTIVIDAD PROPUESTA 6** (pág. 71)

*Visualiza los nombres de alumnos que tengan una nota entre 7 y 8 en la asignatura de "FOL".*

**SELECT** APENOM **FROM** ALUMNOS, ASIGNATURAS, NOTAS **WHERE** (ALUMNOS.DNI = NOTAS.DNI) **AND** (NOTAS.COD = ASIGNATURAS.COD) **AND** (ASIGNATURAS.NOMBRE = 'FOL') **AND** (NOTA > 7 AND NOTA < 8 );

¿CÓMO SERÍA CON JOIN?

*Visualiza los nombres de asignaturas que no tengan suspensos.*

**SELECT** NOMBRE **FROM** ASIGNATURAS, NOTAS **WHERE** ASIGNATURAS.COD = NOTAS.COD **AND** ASIGNATURAS.COD **NOT IN** (**SELECT** NOTAS.COD **FROM** NOTAS **WHERE** NOTA < 5);

**ACTIVIDADES COMPLEMENTARIAS** (pág. 74)

TABLAS EMPLE Y DEPART

*1) Selecciona apellido, oficio y localidad de los departamentos de los empleados cuyo oficio sea 'ANALISTA'.*

**SELECT** APELLIDO, OFICIO, LOC **FROM** EMPLE, DEPART **WHERE** (EMPLE.DPTO\_NO = DEPART.DPTO\_NO) **AND** DPTO\_NO = (**SELECT** DPTO\_NO **FROM** EMPLE **WHERE** OFICIO = 'ANALISTA');

*2) Obtén los datos de los empleados cuyo director (columna DIR de la tabla EMPLE) sea 'CEREZO'.*

**SELECT** \* **FROM** EMPLE **WHERE** DIR='CEREZO';

*3) Obtén los datos de los empleados del departamento de 'VENTAS'.*

**SELECT** \* **FROM** EMPLE, DEPART **WHERE** (EMPLE.DPTO\_NO = DEPART.DPTO\_NO) **AND** (DEPART.NOMBRE = 'VENTAS');

*4) Obtén los datos de los departamentos que NO tengan empleados.*

**SELECT** DEPART.DPTO\_NO, NOMBRE, LOC **FROM** EMPLE, DEPART “D” **WHERE** D.DPTO\_NO NOT IN (**SELECT** D.DPTO\_NO **FROM** EMPLE, DEPART **WHERE** DEPART.DPTO\_NO = EMPLE.DPTO\_NO);

*5) Obtén los datos de los departamentos que tengan empleados.*

**SELECT** DEPART.DPTO\_NO, NOMBRE, LOC **FROM** EMPLE, DEPART “D” **WHERE** D.DPTO\_NO NOT IN (**SELECT** D.DPTO\_NO **FROM** EMPLE, DEPART **WHERE** DEPART.DPTO\_NO = EMPLE.DPTO\_NO);

*6) Obtén el apellido y el salario de los empleados que superen todos los salarios de los empleados del departamento 20.*

**SELECT** APELLIDO, SALARIO **FROM** EMPLE, DEPART **WHERE** EMPLE.DPTO\_NO = DEPART.DPTO\_NO **AND** SALARIO > **ALL** (**SELECT** SALARIO **FROM** EMPLE, DEPART **WHERE** EMPLE.DPTO\_NO = DEPART.DPTO\_NO **AND** DPTO\_NO = 20);

TABLA LIBRERÍA (pág. 60)

*7) Visualiza el tema, estante y ejemplares de las filas de LIBRERIA con ejemplares comprendidos entre 8 y 15.*

**SELECT** TEMA, ESTANTE, EJEMPLARES **FROM** LIBRERIA **WHERE** EJEMPLARES **BETWEEN** (8, 15);

*8) Visualiza las columnas TEMA, ESTANTE y EJEMPLARES de las filas cuyo ESTANTE no esté comprendido entre la 'B' y la 'D'.*

**SELECT** TEMA, ESTANTE, EJEMPLARES **FROM** LIBRERIA **WHERE** ESTANTE **NOT BETWEEN** ('B', 'D');

*9) Visualiza con una sola orden SELECT todos los temas de LIBRERIA cuyo número de ejemplares sea inferior a los que hay en 'MEDICINA'.*

**SELECT** TEMA **FROM** LIBRERIA **WHERE** EJEMPLARES < (**SELECT** EJEMPLARES **FROM** LIBRERIA **WHERE** TEMA = 'MEDICINA');

*10) Visualiza los temas de LIBRERIA cuyo número de ejemplares no esté entre 15 y 20, ambos incluidos.*

**SELECT** TEMA **FROM** LIBRERIA **WHERE** EJEMPLARES **NOT BETWEEN** (15,20);

TABLAS ALUMNOS, ASIGNATURAS, NOTAS (pág. 71)

*11) Visualiza todas las asignaturas que contengan tres letras 'o' en su interior y tengan alumnos matriculados de 'Madrid'.*

**SELECT** NOMBRE **FROM** ASIGNATURAS, NOTAS, ALUMNOS **WHERE** (ALUMNOS.DNI = NOTAS.DNI **AND** NOTAS.COD = ASIGNATURAS.COD) **AND** (NOMBRE **LIKE** '%o%o%o%') **AND** POBLA = 'Madrid';

*12) Visualiza los nombres de alumnos de 'Madrid' que tengan alguna asignatura suspendida.*

**SELECT** **DISTINCT** APENOM **FROM** ALUMNOS, NOTAS **WHERE** ALUMNOS.DNI = NOTAS.DNI **AND** POBLA = 'Madrid' **AND** NOTA < 5;

*13) Muestra los nombres de alumnos que tengan las misma nota que tiene 'Díaz Fernández, María' en 'FOL' en alguna asignatura.*

**SELECT** **DISTINCT** APENOM **FROM** ALUMNOS “AL”, ASIGNATURAS “AS”, NOTAS “N” **WHERE** (AL.DNI = N.DNI AND N.COD = AS.COD) **AND** NOTA = (**SELECT** NOTA **FROM** ALUMNOS, NOTAS, ASIGNATURAS **WHERE** NOTAS.DNI = ALUMNOS.DNI **AND** ASIGNATURAS.COD = NOMBRE.COD **AND** APENOM = ‘Díaz Fernández, María” **AND** NOMBRE = ‘FOL’);

*14) Obtén los datos de las asignaturas que no tengan alumnos.*

**SELECT** \* **FROM** ASIGNATURAS **WHERE** COD **NOT IN** (**SELECT** COD **FROM** NOTAS);

*15) Obtén el nombre y apellido de los alumnos que tengan nota en la asignatura con código 1.*

**SELECT** APENOM **FROM** ALUMNOS **WHERE** DNI **IN** (**SELECT** DNI **FROM** NOTAS **WHERE** COD = 1);

**SELECT** APENOM **FROM** ALUMNOS AL, NOTAS N **WHERE** AL.DNI = N.DNI **AND** COD = 1;

*16) Obtén el nombre y apellido de los alumnos que no tengan nota en la asignatura con código 1.*

**SELECT** APENOM **FROM** ALUMNOS **WHERE** DNI **NOT IN** (**SELECT** DNI **FROM** NOTAS **WHERE** COD = 1);

**TEMA 4**

**CASO PRÁCTICO 1** (pág. 76)

*Obtén el valor absoluto del SALARIO – 10.000 para todas las filas de la tabla EMPLE.*

**SELECT** APELLIDO, SALARIO, **ABS**(SALARIO-10.000) **FROM** EMPLE;

*A partir de la tabla EMPLE obtenemos el SALARIO, la COMISION y la suma de ambos:*

**SELECT** APELLIDO, SALARIO, COMISION, (SALARIO+COMISION) “Salario total” **FROM** EMPLE;

*A partir de la tabla EMPLE obtenemos el SALARIO, la COMISION y la suma de ambos, pero si esta última es nula sustituimos su valor por 0.*

**SELECT** APELLIDO, SALARIO, COMISION, SALARIO+ **NVL**(COMISION, 0) “Salario total” **FROM** EMPLE;

**ACTIVIDAD PROPUESTA 1** (pág. 79)

*¿Cuál sería la salida de estas funciones?*

ABS(146) = 146 POWER(3, -1) = 0.33 ROUND(33.67) = 34

ABS(-30) = 30 POWER(10, 0) = 1 ROUND(-33.67, -2) = 0

CEIL(2) = 2 POWER (3, 2) = 9 ROUND(-33.67, 2) = -33,67

CEIL(-2.3) = -2 MOD(22, 23) = 22 ROUND(-33.27, -1) = -30

CEIL(1.3) = 2 MOD(10, 3) = 1 ROUND(-33.27, 1) = -33.3

CEIL(-2) = -2 TRUNC(67.232) = 67 TRUNC(67.232, 2) = 67.23

FLOOR(-2) = -2 FLOOR(-2.3) = -3 TRUNC (67.232-2) = 0

FLOOR(2) = 2 FLOOR(1.3) = 1 TRUNC(67.58, 1) = 67.5

TRUNC(67.58, -1) = 60

**CASO PRÁCTICO 2** (pág. 80)

*Cálculo del salario medio de los empleados del departamento 10 de la tabla EMPLE.*

**SELECT** **AVG**(SALARIO) **FROM** EMPLE **WHERE** DPTO\_NO = 10;

*Cálculo del número de filas de la tabla EMPLE.*

**SELECT** **COUNT(**\***) FROM** EMPLE;

*Cálculo del número de filas de la tabla EMPLE donde la COMISION no es nula:*

**SELECT COUNT**(COMISION) **FROM** EMPLE;

*Cálculo del máximo salario de la tabla EMPLE.*

**SELECT MAX**(SALARIO) **FROM** EMPLE;

*Obtén el apellido máximo (alfabéticamente) de la tabla EMPLE:*

**SELECT MAX**(APELLIDO) **FROM** EMPLE;

*Obtén el apellido del empleado que tiene mayor salario.*

**SELECT** APELLIDO, SALARIO **FROM** EMPLE **WHERE** SALARIO = (**SELECT** **MAX**(SALARIO) **FROM** EMPLE);

*Obtén el mínimo salario de la tabla EMPLE.*

**SELECT MIN**(SALARIO) **FROM** EMPLE;

*Obtén el apellido del empleado que tiene el mínimo salario.*

**SELECT** APELLIDO, SALARIO **FROM** EMPLE **WHERE** SALARIO = (**SELECT MIN**(SALARIO) **FROM** EMPLE);

*Consigue la suma de todos los salarios de la tabla EMPLE.*

**SELECT SUM**(SALARIO) **FROM** EMPLE;

Obtén la varianza de todos los salarios de la tabla EMPLE.

**SELECT VARIANCE**(SALARIO) **FROM** EMPLE;

**CASO PRÁCTICO 3** (pág. 82)

*Calcula el número de oficios que hay en la tabla EMPLE.*

**SELECT COUNT**(**DISTINCT** OFICIO) “Oficios” **FROM** EMPLE;

**ACTIVIDAD PROPUESTA 2** (pág. 82)

*A partir de la tabla EMPLE, ¿cuántos apellidos empiezan por la letra ‘A’?*

**SELECT COUNT**(**DISTINCT** APELLIDO) **FROM** EMPLE **WHERE** APELLIDO **LIKE** ‘A%’;

*Obtén el apellido o apellidos de empleados que empiecen por la letra ‘A’ y que tengan el máximo salario (de los que empiezan por la letra ‘A’).*

**SELECT** APELLIDO **FROM** EMPLE **WHERE** APELLIDO **LIKE** ‘A%’ **AND** SALARIO = (**SELECT** **MAX**(SALARIO) **FROM** EMPLE **WHERE** APELLIDO **LIKE** ‘A%’);

**CASO PRÁCTICO 4** (pág. 83)

*A partir de la tabla NOTAS\_ALUMNOS, obtén por cada alumno su mayor y su menor nota.*

**SELECT** NOMBRE\_ALUMNO, **GREATEST**(NOTA1, NOTA2, NOTA3) “Mayor”, **LEAST**(NOTA1, NOTA2, NOTA3) “Menor” **FROM** NOTAS\_ALUMNOS;

**CASO PRÁCTICO 5** (pág. 85)

*Devuelve las letras cuyo valor ASCII es 75 y 65.*

**SELECT CHR**(75), **CHR**(65) **FROM** DUAL;

*Obtén el apellido de los empleados de la tabla EMPLE en el siguiente formato:*

*El apellido es: APELLIDO*

**SELECT CONCAT**(‘El apellido es: ‘, APELLIDO) **FROM** EMPLE;

*Visualiza en mayúsculas, minúsculas y tipo título la cadena: ‘oRACle Y sqL’.*

**SELECT UPPER**(‘oRACle Y sqL’) “Mayúsculas”, **LOWER**(‘oRACle Y sqL’) “Minúsculas”, **INITCAP**(‘oRACle Y sqL’) “Título” **FROM** DUAL;

*Para cada fila de la tabla NOTAS\_ALUMNOS obtenemos en una columna el nombre del alumno con una longitud de 30 caracteres y rellenando por la izquierda con puntos y en otra columna lo mismo pero rellenando por la derecha.*

**SELECT LPAD**(NOMBRE\_ALUNO, 30, ‘.’) “Izquierda”, **RPAD**(NOMBRE\_ALUMNO, 30, ‘.’) “Derecha” **FROM** NOTAS\_ALUMNOS;

*Eliminar los espacios en blanco a la izquierda de la cadena ‘ hola’ y a la derecha de ‘ adiós ‘ y concatenar el resultado.*

**SELECT LTRIM**(‘ hola’) **||** **RTRIM** (‘ adiós ‘) **||** ‘\*’ **FROM** DUAL;

*A partir de la tabla MISTEXTOS, quitar los caracteres punto y comillas de la derecha de los títulos.*

**SELECT** **RTRIM**(TITULO, ‘ .” ’) **FROM** MISTEXTOS;

*Quitar las comillas de la izquierda del título.*

**SELECT LTRIM**(TITULO, ‘ ” ’ **FROM** MISTEXTOS;

*Sustituir las letras ‘O’ por ‘AS’ en la cadena ‘BLANCO Y NEGRO’.*

**SELECT REPLACE**(‘BLANCO Y NEGRO’, ‘O’, ‘AS’) **FROM** DUAL;

*Partiendo de la cadena ‘ABCDEFG’, obtenemos en una columna dos caracteres a partir de la tercera posición, en otra dos caracteresa a partir d ela tercera posición empezando por el final y en la última la cadena a partir de su cuarta posición.*

**SELECT SUBSTR**(‘ABCDEFG’, 3, 2) “s1”, SUBSTR(‘ABCDEFG’, -3, 2) “s2”, SUBSTR(‘ABCDEFG’, 4) “s3” **FROM** DUAL;

*Visualiza el apellido y su primera letra para los empleados del departamento 10 de la tabla EMPLE.*

**SELECT** APELLIDO, SUBSTR(APELLIDO, 1, 1) **FROM** EMPLE **WHERE** DEPT\_NO = 10;

*A partir de la cadena ‘LOS PILARES DE LA TIERRA’, sustituye la ‘A’ por ‘a’, la ‘E’ por ‘e’, la ‘I’ por ‘i’, la ‘O’ por ‘o’ y la ‘U’ por ‘u’.*

**SELECT TRANSLATE**(‘LOS PILARES DE LA TIERRA’, ‘AEIOU’, ‘aeiou’) **FROM** DUAL;

**CASO PRÁCTICO 6** (pág. 88)

*Obtén el valor ASCII de ‘A’.*

**SELECT** **ASCII**(‘A’) **FROM** DUAL;

*A partir de la cadena ‘II VUELTA CICLISTA A TALAVERA’ encuentra la segunda ocurrencia de ‘TA’ desde la posición 3.*

**SELECT INSTR**(‘II VUELTA CICLISTA A TALAVERA’, ‘TA’, 3, 2) “Ejemplo” **FROM** DUAL;

*A partir de la tabla MISTEXTOS encuentra la posición de la segunda ocurrencia de la letra ‘A’ en la columna AUTOR a partir del comienzo.*

**SELECT** AUTOR, **INSTR**(AUTOR, ‘A’, 1, 2) **FROM** MISTEXTOS;

*Calcula el número de caracteres de las columnas TITULO y AUTOR para todas las filas de la tabla MISTEXTOS.*

**SELECT** TITULO, **LENGTH**(TITULO), AUTOR, **LENGTH**(AUTOR) **FROM** MISTEXTOS;

**CASO PRÁCTICO 7** (pág. 90)

*Devuelve la fecha del sistema.*

**SELECT** **SYSDATE** **FROM** DUAL;

*A partir de la tabla EMPLE, suma doce meses a la fecha de alta para los empleados del departamento 10.*

**SELECT** **ADD\_MONTHS**(FECHA\_ALT, 12) **FROM** EMPLE **WHERE** DEPT\_NO = 10;

*Obtén de la tabla EMPLE el último día del mes para cada una de las fechas de alta de los empleados del departamento 10.*

**SELECT LAST\_DAY**(FECHA\_ALT) **FROM** EMPLE **WHERE** DEPT\_NO = 10;

*Halla la edad de una persona a partir de su fecha de nacimiento y la SYSDATE.*

**SELECT TRUNC**(**MONTHS\_BETWEEN**(**SYSDATE**, ‘18/11/1964’) / 12) “Edad actual” **FROM** DUAL;

*Si hoy es jueves 19 de octubre de 2006 (SYSDATE), ¿qué fecha será el próximo jueves?*

**SELECT NEXT\_DAY**(**SYSDATE**, ‘JUEVES’) “Siguiente jueves” **FROM** DUAL;

**CASO PRÁCTICO 8** (pág. 92)

*A partir de la tabla EMPLE, obtén la fecha de alta (columna FECHA\_ALT) formateada de manera que aparezca el nombre del mes con todas sus letras (month), el número de día de mes (dd) y el año (yyyy) para aquellos empleados del departamento 10.*

**SELECT TO\_CHAR** (FECHA\_ALT, ‘month DD, YYYY’) “Nueva fecha” **FROM** EMPLE **WHERE** DEPT\_NO = 10;

*Obtén la fecha de alta de forma que aparezca el nombre del mes con tres letras (mon), el número de día del año (ddd), el último dígito del año (y) y los tres últimos dígitos del año (yyy).*

**SELECT TO\_CHAR** (FECHA\_ALT, ‘mon ddd y yyy’) “Fecha” **FROM** EMPLE **WHERE** DEPT\_NO = 10;

**CASO PRÁCTICO 9** (pág. 93)

*Define otro idioma para las fechas de la sesión actual.*

**ALTER SESSION SET NLS\_DATE\_LANGUAGE** = French;

**CASO PRÁCTICO 10** (pág. 97)

*Cambia el formato de la fecha para que aparezca el año con cuatro dígitos.*

**ALTER SESSION SET NLS\_DATE\_FORMAT** = ‘DD/MM/YYYY’;

*Convierte una cadena a tipo DATE.*

**SELECT TO\_DATE**(‘01012006’) **FROM** DUAL;

*Obtén el nombre del mes a partir de la cadena ‘01012007’.*

**SELECT TO\_CHAR**(**TO\_DATE** (‘01012007’, ‘ddmmyyyy’), ‘Month’) “Mes” **FROM** DUAL;

**CASO PRÁCTICO 11** (pág. 99)

*Seleccionar todas las filas y codificar el OFICIO. Si el oficio es PRESIDENTE, codificar con un 1; si es EMPLEADO, con un 2; en cualquier otro caso, codificar con un 5.*

**SELECT** APELLIDO, OFICIO, **DECODE**(UPPER(OFICIO), ‘PRESIDENTE’, 1, ‘EMPLEADO’, 2, 5) “Código” **FROM** EMPLE;

*Calcular el número de bytes que tiene la columna APELLIDO de la tabla EMPLE para aquellos empleados del departamento 10.*

**SELECT** APELLIDO, **VSIZE**(APELLIDO) BYTES **FROM** EMPLE **WHERE** DEPT\_NO = 10;

*Representar en hexadecimal los caracteres 1 al 4 del apellido ‘SALA’ de la tabla EMPLE.*

**SELECT** APELLIDO, **DUMP**(APELLIDO, 16, 1, 4) **FROM** EMPLE **WHERE** APELLIDO **LIKE** ‘SALA’;

*Visualiza el usuario que está conectado y su identificador.*

**SELECT USER, UID FROM** DUAL;

*Muestra el usuario que está conectado.*

**SHOW USER;**

**ACTIVIDADES COMPLEMENTARIAS** (pág. 101)

*1) Dada la tabla EMPLE, obtén el sueldo medio, el número de comisiones no nulas, el máximo sueldo y el mínimo sueldo de los empleados del departamento 30. Emplea el formato adecuado para la salida para las cantidades numéricas.*

**SELECT AVG**(SALARIO), **COUNT**(COMISION), **MAX**(SALARIO), **MIN**(SALARIO) **FROM** EMPLE **WHERE** DEPT\_NO = 30;

*2) Visualiza los temas con mayor número de ejemplares de la tabla LIBRERÍA y que tengan, al menos, una ‘E’ (pueden ser un tema o varios).*

**SELECT** TEMA, **MAX**(EJEMPLARES) **FROM** LIBRERIA **WHERE** EJEMPLARES = **MAX**(EJEMPLARES) **AND** TEMA **LIKE** ‘%E%’;

*3) Dada la tabla MISTEXTOS, ¿qué sentencia SELECT se debe ejecutar para tener este resultado?*

RESULTADO

---------------------------------------------------------------

METODOLOGÍA DE LA PROGRAMACIÓN-^-^-^-^-

INFORMÁTICA BÁSICA-^-^-^-^-^-^-^-^-^-^-^-^-^-

**SELECT CONCAT**( ‘-’ , **RPAD**( **RTRIM**(**LTRIM**(TITULO, ‘”’), ‘.”’ ) , 50, ‘^-’) ) “RESULTADO” **FROM** MISTEXTOS;

*4) Visualiza los títulos de la tabla MISTEXTOS sin los caracteres punto y comillas, y en minúscula de dos formas conocidas.*

**SELECT LOWER(RTRIM(LTRIM**(TITULO, ‘“‘), ‘.”’) ) **FROM** MISTEXTOS;

**SELECT LOWER(TRANSLATE**(TITULO, ‘.”’)) **FROM** MISTEXTOS;

*5) Dada la tabla LIBROS, escribe la sentencia SELECT que visualice dos columnas, una con el AUTOR y otra con el apellido del autor.*

**SELECT** AUTOR, **REPLACE**(AUTOR, **SUBSTR**(AUTOR, **INSTR**(AUTOR, ‘,’, 1, 1) )) “APELLIDO” **FROM** LIBROS;

*6) Escribe la sentencia SELECT que visualice las columnas de AUTOR y otra columna con el nombre del autor (sin el apellido) de la tabla LIBROS.*

**SELECT** AUTOR, **SUBSTR**(AUTOR, **INSTR**(AUTOR, ‘,’, 1, 1) + 1) “NOMBRE” **FROM** LIBROS;

*7) A partir de la tabla LIBROS, realiza una sentencia SELECT que visualice en una columna primero el nombre del autor y luego su apellido.*

**SELECT SUBSTR**(AUTOR, **INSTR**(AUTOR, ‘,’, 1, 1) + 1) “NOMBRE”, **REPLACE**(AUTOR, **SUBSTR**(AUTOR, **INSTR**(AUTOR, ‘,’, 1, 1) )) “APELLIDO” **FROM** LIBROS;

*8) A partir de la tabla LIBROS, realiza una sentencia SELECT para que aparezcan los títulos ordenados por su número de caracteres.*

**SELECT** TITULO **FROM** LIBROS **ORDER** **BY** **LENGTH**(TITULO);

*9) Dada la tabla NACIMIENTOS, realiza una sentencia SELECT que obtenga la siguiente salida: NOMBRE, FECHANAC, FECHA\_FORMATEADA, donde FECHA\_FORMATEADA tiene el siguiente formato: “Nació el 12 de mayo de 1982”.*

select NOMBRE, to\_char(FECHANAC, '"Nació el " DD " de " month " de " YYYY"' ) "Fecha formateada" from NACIMIENTOS; ¿

*10) Dada la tabla LIBRERÍA, haz una sentencia SELECT que visualice el tema, el último carácter del tema que no sea blanco y el número de caracteres de tema (sin contar los blancos de la derecha) ordenador por tema.*

**SELECT** TEMA, **SUBSTR**( **RTRIM**(TEMA, ' ') , -1, 1) “ÚLTIMO CARACTER”, **LENGTH**(**RTRIM**(TEMA, ' ')) “LONGITUD” **FROM** "LIBRERIA" **ORDER BY** TEMA;

*11) A partir de la tabla NACIMIENTOS, visualiza en una columna el NOMBRE seguido de su fecha de nacimiento formateada (quita los blancos del nombre).*

**SELECT** NOMBRE, **REPLACE**(FECHANAC, ‘ ‘) “FECHA DE NACIMIENTO” **FROM** NACIMIENTOS; MAL: tiene q salir todo en la misma columna, así:

select concat(rtrim(NOMBRE, ' '), to\_char(FECHANAC, '" nació el "DD" de "month" de "YYYY')) "Nombre y nacimiento" from NACIMIENTOS;

*12) Convierte la cadena ‘010712’ a fecha y visualiza su nombre de mes en mayúsculas.*

**SELECT TO\_DATE**(‘010712’, **MONTH**) **FROM** DUAL;

*13) Visualiza aquellos temas de la tabla LIBRERÍA cuyos ejemplares sean 7 con el nombre de tema de ‘SEVEN’; el resto de temas que no tengan 7 ejemplares se visualizarán como están.*

**SELECT DECODE**(EJEMPLARES, 7, ’SEVEN’, TEMA) “TEMA” **FROM** LIBRERIA;

*14) A partir de la tabla EMPLE, obtén el apellido de los empleados que lleven más de 15 años trabajando.*

SELECT APELLIDO FROM EMPLE WHERE MONTHS\_BETWEEN(SYSDATE, FECHA\_ALT)/12 > 15;

*15) Selecciona el apellido de los empleados de la tabla EMPLE que lleven más de 16 años trabajando en el departamento ‘VENTAS’.*

SELECT APELLIDO FROM EMPLE WHERE EMPLE.DEPT = DEPART.DEPT AND MONTHS\_BETWEEN(SYSDATE, FECHA\_ALT)/12 > 16 AND DNOMBRE = ‘VENTAS’;

*16) Visualiza el apellido, el salario y el número de departamento de aquellos empleados de la tabla EMPLE cuyo salario sea el mayor de su departamento.*

**SELECT** APELLIDO, SALARIO, DEPT\_NO **FROM** EMPLE **WHERE** SALARIO = (SELECT MAX(SALARIO) FROM EMPLE WHERE DEPT\_NO = \*\*); MAL

*17) Visualiza el apellido, salario y número de departamento de aquellos empleados de la tabla EMPLE cuyo salario supere a la media en su departamento.*

SELECT APELLIDO, SALARIO, DEPT\_NO FROM EMPLE WHERE SALARIO > (SELECT AVG(SALARIO) FROM EMPLE WHERE DEPT\_NO = ); MAL

**TEMA 5**

Caso práctico 1

SELECT DEPT\_NO, COUNT(\*) FROM EMPLE GROUP BY DEPT\_NO;

SELECT DEPT\_NO, COUNT(\*) FROM EMPLE GROUP BY DEPT\_NO HAVING COUNT(\*) > 4;

Actividad propuesta 1

SELECT DEPT\_NO, AVG(SALARIO) FROM EMPLE GROUP BY DEPT\_NO HAVING AVG(SALARIO) >= (SELECT AVG(SALARIO) FROM EMPLE);

Caso práctico 2

---------------

SELECT DEPT\_NO, TO\_CHAR (SUM(SALARIO), '99G999D99' ) "Suma", TO\_CHAR (MAX(SALARIO), '99G999D99' ) "Máximo", TO\_CHAR (MIN(SALARIO), '99G999D99') "Mínimo" FROM EMPLE GROUP BY DEPT\_NO;

SELECT COUNT(\*), OFICIO, DEPT\_NO FROM EMPLE GROUP BY OFICIO, DEPT\_NO;

SELECT MAX(COUNT(\*)) "Máximo" FROM EMPLE GROUP BY DEPT\_NO;

Actividad propuesta 2

---------------------

SELECT DNOMBRE, DEPT\_NO, MAX(COUNT(SELECT COUNT(\*) FROM EMPLE GROUP BY DEPT\_NO HAVING COUNT(\*) > 4)) FROM EMPLE GROUP BY DEPT\_NO HAVING (SELECT COUNT(\*) FROM EMPLE GROUP BY DEPT\_NO HAVING COUNT(\*) > 4);

Caso práctico 99

----------------

SELECT D.DEPT\_NO, DNOMBRE, COUNT(E.EMP\_NO)

FROM EMPLE E, DEPART D

WHERE E.DEPT\_NO = D.DEPT\_NO

GROUP BY D.DEPT\_NO, DNOMBRE;

SELECT D.DEPT\_NO, DNOMBRE, COUNT(E.EMP\_NO)

FROM EMPLE E, DEPART D

WHERE E.DEPT\_NO (+) = D.DEPT\_NO

GROUP BY D.DEPT\_NO, DNOMBRE;

Actividad propuesta 3

---------------------

SELECT D.DEPT\_NO, DNOMBRE, COUNT(\*)

FROM EMPLE E, DEPART D

WHERE E.DEPT\_NO (+) = D.DEPT\_NO

GROUP BY D.DEPT\_NO, DNOMBRE;

SELECT E.DEPT\_NO, DNOMBRE, COUNT(E.EMP\_NO)

FROM EMPLE E, DEPART D

WHERE E.DEPT\_NO (+) = D.DEPT\_NO

GROUP BY D.DEPT\_NO, DNOMBRE;

Caso práctico 3

---------------

SELECT NOMBRE FROM ALUM UNION SELECT NOMBRE FROM NUEVOS;

Caso práctico 4

---------------

SELECT NOMBRE FROM ALUM INTERSECT SELECT NOMBRE FROM ANTIGUOS;

Actividad propuesta 4

---------------------

SELECT NOMBRE FROM ALUM WHERE NOMBRE IN (SELECT NOMBRE FROM ANTIGUOS);

Caso práctico 5

---------------

SELECT NOMBRE, LOCALIDAD FROM ALUM MINUS SELECT NOMBRE, LOCALIDAD FROM ANTIGUOS ORDER BY LOCALIDAD;

SELECT NOMBRE, LOCALIDAD FROM ALUM WHERE NOMBRE NOT IN (SELECT NOMBRE FROM ANTIGUOS) ORDER BY LOCALIDAD;

SELECT NOMBRE FROM ALUMN INTERSECT SELECT NOMBRE FROM NUEVOS MINUS SELECT NOMBRE FROM ANTIGUOS;

SELECT NOMBRE FROM ALUMN WHERE NOMBRE IN (SELECT NOMBRE FROM NUEVOS MINUS SELECT NOMBRE FROM ANTIGUOS);

SELECT NOMBRE FROM ALUMN WHERE NOMBRE IN (SELECT NOMBRE FROM NUEVOS UNION SELECT NOMBRE FROM ANTIGUOS);

SELECT NOMBRE FROM ALUM WHERE NOMBRE IN (SELECT NOMBRE FROM NUEVOS) OR NOMBRE IN (SELECT NOMBRE FROM ANTIGUOS);

Actividad propuesta 5

---------------------

SELECT NOMBRE FROM ALUM WHERE NOMBRE IN (SELECT NOMBRE FROM NUEVOS) OR NOMBRE IN (SELECT NOMBRE FROM ANTIGUOS);

SELECT NOMBRE FROM ALUMN WHERE NOMBRE IN (SELECT NOMBRE FROM NUEVOS UNION SELECT NOMBRE FROM ANTIGUOS);

**Actividades complementarias**

*TABLAS EMPLE Y DEPART*

*1) Partiendo de la tabla EMPLE, visualiza por cada oficio de los empleados del departamento VENTAS la suma de salarios.*

**SELECT** OFICIO**, SUM(**SALARIO**) FROM** EMPLE**,** DEPART **WHERE** EMPLE.DEPT\_NO =DEPART.DEPT\_NO **AND** DNOMBRE = 'VENTAS' **GROUP BY** OFICIO**;**

select sum(salario) "SUMA SALARIO", oficio from emple where dept\_no in (select dept\_no from depart where dnombre like 'VENTAS') group by oficio;

*2) Selecciona aquellos apellidos de la tabla EMPLE cuyo salario sea igual a la media del salario en su departamento.*

**SELECT** DEPT\_NO, APELLIDO, SALARIO **FROM** EMPLE **WHERE** (SALARIO, DEPT\_NO) **IN** (**SELECT** **AVG**(SALARIO), DEPT\_NO **FROM** EMPLE **GROUP** **BY** DEPT\_NO) **ORDER** **BY** DEPT\_NO, SALARIO;

*3) A partir de la tabla EMPLE, visualiza el número de empleados de cada departamento cuyo oficio sea ‘EMPLEADO’.*

**SELECT** DEPT\_NO, **COUNT**(\*) “EMPLEADOS” **FROM** EMPLE **WHERE** OFICIO = ‘EMPLEADO’ **GROUP BY** DEPT\_NO;

*4) Desde la tabla EMPLE, visualiza el departamento que tenga más empleados cuyo oficio sea ‘EMPLEADO’.*

**SELECT** DEPT\_NO, **COUNT**(\*) **FROM** EMPLE **WHERE** OFICIO = ‘EMPLEADO’ **GROUP** **BY** DEPT\_NO **HAVING** **COUNT**(\*) = (**SELECT** **MAX**(**COUNT**(\*)) **FROM** EMPLE **WHERE** OFICIO = ‘EMPLEADO’ **GROUP** **BY** DEPT\_NO);

*5) A partir de las tablas EMPLE y DEPART, visualiza el número de departamento y el nombre de departamento que tenga más empleados cuyo oficio sea ‘EMPLEADO’.*

SELECT DEPT\_NO, DNOMBRE FROM DEPART WHERE DEPT\_NO IN (SELECT DEPT\_NO FROM EMPLE WHERE OFICIO LIKE 'EMPLEADO' GROUP BY DEPT\_NO HAVING COUNT(\*) = (SELECT MAX(COUNT(\*)) FROM EMPLE WHERE OFICIO LIKE 'EMPLEADO' GROUP BY DEPT\_NO));

*6) Busca los departamentos que tienen más de dos personas trabajando en la misma profesión.*

SELECT DEPT\_NO, OFICIO, COUNT(\*) FROM EMPLE GROUP BY OFICIO, DEPT\_NO HAVING COUNT(\*) > 2;

*TABLAS ALUM, ANTIGUOS Y NUEVOS*

*7) Visualiza los nombres de los alumnos de la tabla ALUM que aparezcan en estas dos tablas: ANTIGUOS Y NUEVOS.*

SELECT ALUM.NOMBRE FROM ALUM, NUEVOS, ANTIGUOS WHERE ALUM.NOMBRE = NUEVOS.NOMBRE AND NUEVOS.NOMBRE = ANTIGUOS.NOMBRE ;

SELECT NOMBRE FROM NUEVOS INTERSECT SELECT NOMBRE FROM ANTIGUOS INTERSECT SELECT NOMBRE FROM ALUM;

*8) Escribe las distintas formas en que se puede poner la consulta anterior llegando al mismo resultado.*

SELECT NOMBRE FROM ALUM WHERE NOMBRE IN (SELECT NOMBRE FROM NUEVOS) AND NOMBRE IN (SELECT NOMBRE FROM ANTIGUOS);

*9) Visualiza aquellos nombres de la tabla ALUM que no estén en la tabla ANTIGUOS ni en la tabla NUEVOS.*

SELECT NOMBRE FROM ALUM WHERE NOMBRE NOT IN (SELECT NOMBRE FROM NUEVOS) AND NOMBRE NOT IN (SELECT NOMBRE FROM ANTIGUOS);

SELECT NOMBRE FROM ALUM MINUS SELECT NOMBRE FROM NUEVOS MINUS SELECT NOMBRE FROM ANTIGUOS;

TABLAS PERSONAL, PROFESORES Y CENTROS (hacer DESC de las tablas)

*10) Realiza una consulta en la que aparezca por cada centro y en cada especialidad el número de profesores. Si el centro no tiene profesores, debe aparecer un 0 en la columna de número de profesores. Las columnas a visualizar son: nombre de centro, especialidad y número de profesores.*

select C.NOMBRE, ESPECIALIDAD, DECODE(count(\*), NULL, 0) “NUMERO DE PROFESORES” from PROFESORES “P”, CENTROS “C” where C.COD\_CENTRO = P.COD\_CENTRO group by COD\_CENTRO, ESPECIALIDAD;

*11) Obtén por cada centro el número de empleados. Si el centro carece de empleados, ha de aparecer un 0 como número de empleados.*

*12) Obtén la especialidad con menos profesores.*

TABLAS BANCOS, SUCURSALES, CUENTAS Y MOVIMIENTOS (hacer DESC de las tablas)

TABLA BANCOS: contiene los datos de los bancos, una fila por cada banco, un banco se identifica por el COD\_BANCO

TABLA SUCURSALES: contiene los datos de las sucursales, una fila por sucursal, cada sucursal se identifica por el COD\_BANCO+COD\_SUCUR

TABLA CUENTAS: contiene los datos de las cuentas abiertas en las sucursales de los bancos. Una cuenta se identifica por las columnas COD\_BANCO+COD\_SUCUR+NUM\_CTA. Contiene los saldos de las cuentas. SALDO\_DEBE contiene la suma de reintegros y SALDO\_HABER la suma de ingresos.

TABLA MOVIMIENTOS: contiene los movimientos de las cuentas. Una fila representa un movimiento de una cuenta. La columna TIPO\_MOV puede ser I (ingreso) o R (reintegro).

*13) Obtén el banco con más sucursales. Los datos a obtener son Nombre Banco y Nº Sucursales*

*14) El saldo actual de los bancos de ‘GUADALAJARA’, 1 fila por cada banco. (Nombre banco, saldo debe, saldo haber.)*

*15) Datos de la cuenta o cuentas con más movimientos. (Nombre cta, nº movimientos)*

*16) El nombre de la sucursal que haya tenido más suma de reintegros. (Nombre sucursal, suma reintegros)*

**TEMA 6**

K

**TEMA 7**

K