

Usando Funciones SQL de una Fila de Caracteres, Números y Fechas

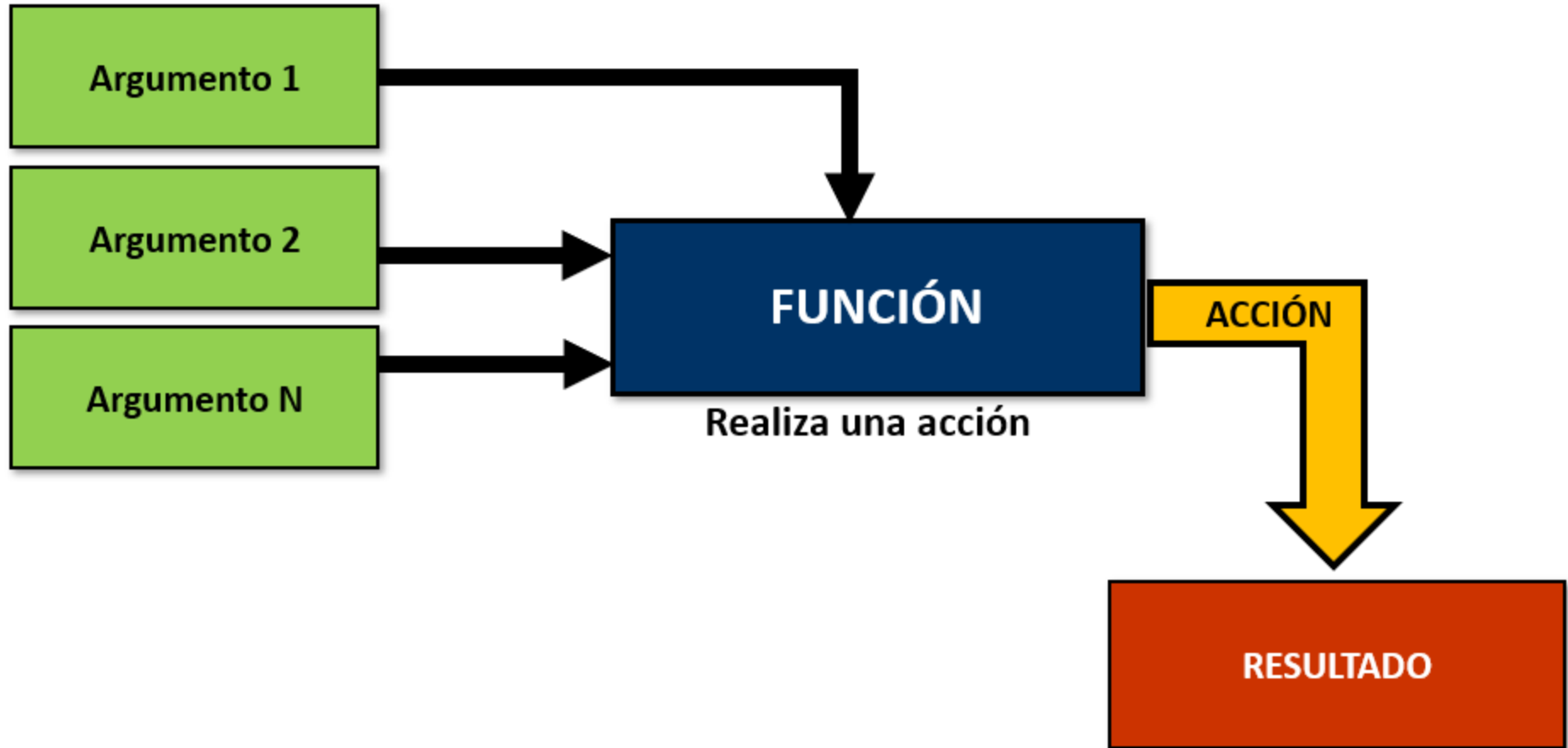


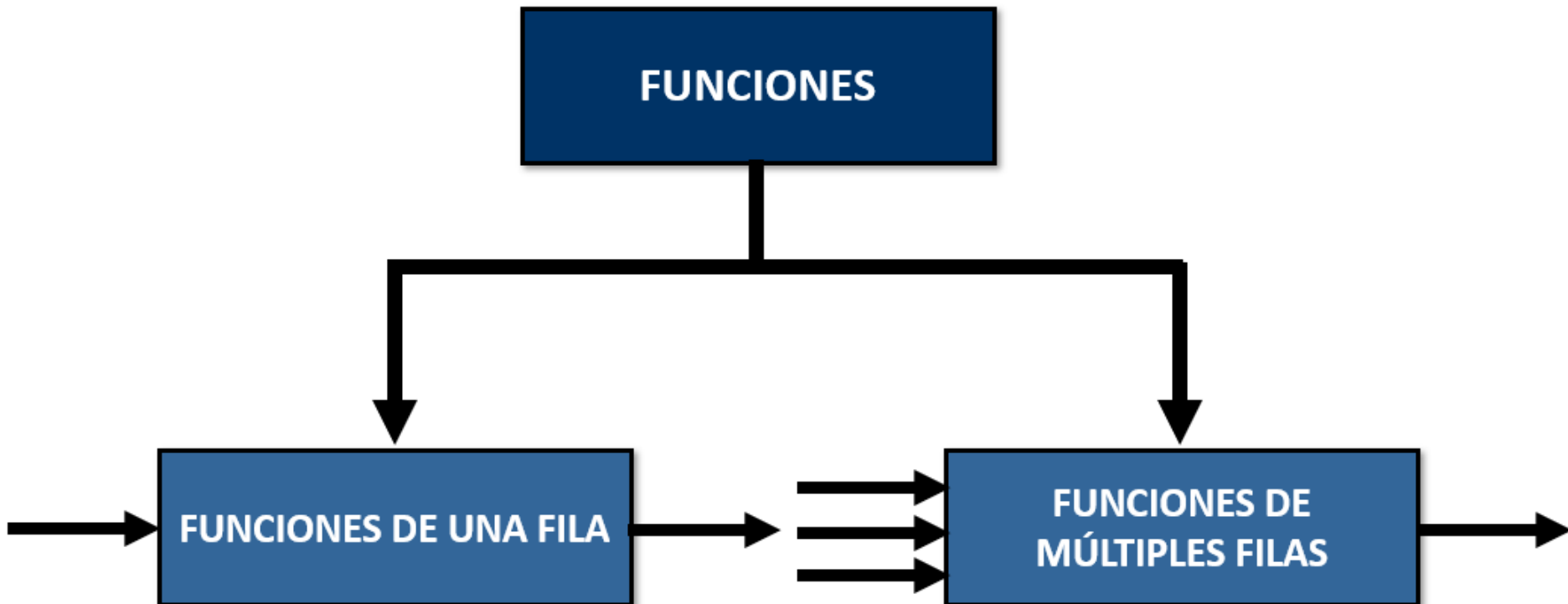
A photograph of a curved library aisle with tall wooden bookshelves filled with books. The perspective is from a lower level looking up at the shelves. The image has a dark, muted color palette.

Alejandra Gajardo San Martín
ale.gajardo@professor.duoc.cl

- Describir las características generales de las Funciones SQL.
- Describir las características de las Funciones SQL que operan con una fila.
- Describir los tipos de Funciones que operan con una sola fila: de caracteres, números, fechas, conversión y generales.
- Cómo convertir, reemplazar o manipular caracteres en sentencias SQL
- Cómo manipular números en sentencias SQL.
- Cómo efectuar operaciones con fechas en sentencias SQL.

Funciones SQL





Se pueden usar en
las cláusulas SELECT,
WHERE, ORDER BY y
SET de una sentencia
SQL

Realizan una acción
por cada fila que se
selecciona, actualiza
o inserta y por cada
una de ellas entrega
un resultado

Aceptan uno o más
argumentos
(constante, variable,
columna, expresión)
y se puede usar en
forma anidada

- Sintaxis

nombre_función [(argumento1, argumento2, ...)]



Funciones para trabajar con Caracteres



**Aceptan como argumentos de entrada datos del tipo
caracter y retornan datos del tipo caracteres o números**

FUNCIONES DE CARACTERES

**FUNCIONES DE CONVERSIÓN
MAYÚSCULAS O MINÚSCULAS**

**FUNCIONES DE TRANSFORMACIÓN
O
MANIPULACIÓN**

- Funciones de conversión del texto a mayúsculas y minúsculas:

LOWER(*columna | expresión*)

UPPER(*columna | expresión*)

INITCAP(*columna | expresión*)

- Funciones de reemplazo o manipulación de caracteres:

SUBSTR(*columna | expresión, m [, n]*)

TRIM(*columna | expresión*)

LENGTH(*columna | expresión*)

LPAD(*columna | expresión, n, 'caracter_de_relleno'*)

RPAD(*columna | expresión, n, 'caracter_de_relleno'*)

INSTR(*columna | expresión, 'texto_buscado', [, m]], n*)

RTRIM(*columna | expresión*)

LTRIM(*columna | expresión*)

REPLACE(*columna | expresión, 'texto_a_buscar', 'texto_reemplazo'*)

CONCAT(*columna1 | expresión1, columna2 | expresión2*)

TRIM('caracter' FROM *columna | expresión*)

- Ejemplo:

```
SELECT last_name, UPPER(last_name), job_id, LOWER(job_id), INITCAP(job_id)
FROM employees;
```

	LAST_NAME	UPPER(LAST_NAME)	JOB_ID	LOWER(JOB_ID)	INITCAP(JOB_ID)
1	Abel	ABEL	SA_REP	sa_rep	Sa_Rep
2	Ande	ANDE	SA_REP	sa_rep	Sa_Rep
3	Atkinson	ATKINSON	ST_CLERK	st_clerk	St_Clerk
4	Austin	AUSTIN	IT_PROG	it_prog	It_Prog
5	Baer	BAER	PR_REP	pr_rep	Pr_Rep
6	Baida	BAIDA	PU_CLERK	pu_clerk	Pu_Clerk
7	Banda	BANDA	SA_REP	sa_rep	Sa_Rep
8	Bates	BATES	SA_REP	sa_rep	Sa_Rep
9	Bell	BELL	SH_CLERK	sh_clerk	Sh_Clerk
10	Bernstein	BERNSTEIN	SA_REP	sa_rep	Sa_Rep
.....					
.....					
104	Walsh	WALSH	SH_CLERK	sh_clerk	Sh_Clerk
105	Weiss	WEISS	ST_MAN	st_man	St_Man
106	Whalen	WHALEN	AD_ASST	ad_asst	Ad_Asst
107	Zlotkey	ZLOTKEY	SA_MAN	sa_man	Sa_Man

- Ejemplo:

```
SELECT last_name apellido, CONCAT('Su salario es ', salary),  
       SUBSTR(last_name , 2,3), LENGTH(last_name), INSTR(last_name, 'a')  
FROM employees;
```

⚡ APELLIDO	⚡ CONCAT('SUSALARIOES',SALARY)	⚡ SUBSTR(LAST_NAME,2,3)	⚡ LENGTH(LAST_NAME)	⚡ INSTR(LAST_NAME,'A')
1 King	Su salario es 24000	ing	4	0
2 Kochhar	Su salario es 17000	och	7	6
3 De Haan	Su salario es 17000	e H	7	5
4 Hunold	Su salario es 9000	uno	6	0
5 Ernst	Su salario es 6000	rns	5	0
6 Austin	Su salario es 4800	ust	6	0
7 Pataballa	Su salario es 4800	ata	9	2
8 Lorentz	Su salario es 4200	ore	7	0
9 Greenberg	Su salario es 12008	ree	9	0
10 Faviat	Su salario es 9000	avi	6	2
.....				
104 Mavris	Su salario es 6500	avr	6	2
105 Baer	Su salario es 10000	aer	4	2
106 Higgins	Su salario es 12008	igg	7	0
107 Gietz	Su salario es 8300	iet	5	0

- Ejemplo:

```
SELECT last_name "Apellido", INSTR(last_name,'e', 1, 2) "Resultado INSTR",  
       SUBSTR(last_name, -2, 2) "Resultado SUBSTR",  
       TRIM('B' FROM last_name) "Resultado 1er. TRIM",  
       salary, TRIM(1 FROM salary) "Resultado 2do. TRIM"  
FROM employees  
WHERE salary between 9500 AND 10000  
ORDER BY last_name;
```

	Apellido	Resultado INSTR	Resultado SUBSTR	Resultado 1er. TRIM	SALARY	Resultado 2do. TRIM
1	Baer	0	er	Baer	10000	0000
2	Bernstein	7	in	Bernstein	9500	9500
3	Bloom	0	om	Bloom	10000	0000
4	Fox	0	ox	Fox	9600	9600
5	Greene	4	ne	Greene	9500	9500
6	King	0	ng	King	10000	0000
7	Sully	0	ly	Sully	9500	9500
8	Tucker	0	er	Tucker	10000	0000

- Ejemplo:

```
SELECT last_name apellido, REPLACE(last_name, 'A', 'Hola'), salary salario,
       LPAD(salary,10,'*'), RPAD(salary,10,'*')
FROM employees
ORDER BY last_name;
```

	APELLIDO	REPLACE(LAST_NAME,'A','HOLA')	SALARIO	LPAD(SALARY,10,'*')	RPAD(SALARY,10,'*')
1	Abel	Holabel	11000	*****11000	11000*****
2	Ande	Holande	6400	*****6400	6400*****
3	Atkinson	Holatkinson	2800	*****2800	2800*****
4	Austin	Holaustin	4800	*****4800	4800*****
5	Baer	Baer	10000	*****10000	10000*****
6	Baida	Baida	2900	*****2900	2900*****
7	Banda	Banda	6200	*****6200	6200*****
8	Bates	Bates	7300	*****7300	7300*****
9	Bell	Bell	4000	*****4000	4000*****
10	Bernstein	Bernstein	9500	*****9500	9500*****
.....					
104	Walsh	Walsh	3100	*****3100	3100*****
105	Weiss	Weiss	8000	*****8000	8000*****
106	Whalen	Whalen	4400	*****4400	4400*****
107	Zlotkey	Zlotkey	10500	*****10500	10500*****



Funciones para trabajar con Números

Manipulan números, aceptan un valor numérico como argumento y retornan un valor numérico







ROUND(columna | expresión, n)

TRUNC(columna | expresión, n)

MOD(m, n)

- Ejemplo:

```
SELECT ROUND(1234.5678,2), ROUND(1234.5678), ROUND(1235.5678, -1),  
       TRUNC(1234.5678,2), TRUNC(1234.5678), TRUNC(1234.5678, -2)  
FROM DUAL;
```

 ROUND(1234.5678,2)	 ROUND(1234.5678)	 ROUND(1235.5678,-1)	 TRUNC(1234.5678,2)	 TRUNC(1234.5678)	 TRUNC(1234.5678,-2)
1234,57	1235	1240	1234,56	1234	1200

- Ejemplo:

```
SELECT employee_id "ID. EMPLEADO",  
       first_name || ' ' || last_name "NOMBRE EMPLEADO",  
       salary*.023, ROUND(salary*.023), TRUNC(salary*.023)  
FROM employees;
```

	ID. EMPLEADO	NOMBRE EMPLEADO	SALARY*.023	ROUND(SALARY*.023)	TRUNC(SALARY*.023)
1	100	Steven King	552	552	552
2	101	Neena Kochhar	391	391	391
3	102	Lex De Haan	391	391	391
4	103	Alexander Hunold	207	207	207
5	104	Bruce Ernst	138	138	138
6	105	David Austin	110,4	110	110
7	106	Valli Pataballa	110,4	110	110
8	107	Diana Lorentz	96,6	97	96
9	108	Nancy Greenberg	276,184	276	276
10	109	Daniel Faviet	207	207	207
.....					
104	203	Susan Mavris	149,5	150	149
105	204	Hermann Baer	230	230	230
106	205	Shelley Higgins	276,184	276	276
107	206	William Gietz	190,9	191	190

- Ejemplo:

```
SELECT employee_id, salary,  
       salary*commission_pct*.763 "RESULTADO 1",  
       ROUND((salary*commission_pct)*.763) "RESULTADO 2"  
FROM employees  
WHERE commission_pct IS NOT NULL;
```

	EMPLOYEE_ID	SALARY	RESULTADO 1	RESULTADO 2
1	145	14000	4272,8	4273
2	146	13500	3090,15	3090
3	147	12000	2746,8	2747
4	148	11000	2517,9	2518
5	149	10500	1602,3	1602
6	150	10000	2289	2289
7	151	9500	1812,125	1812
8	152	9000	1716,75	1717
9	153	8000	1220,8	1221
10	154	7500	1144,5	1145
32	176	8600	1312,36	1312
33	177	8400	1281,84	1282
34	178	7000	801,15	801
35	179	6200	473,06	473

- Ejemplo:

```
SELECT last_name, salary, MOD(salary, 5000)
FROM employees
WHERE job_id='ST_MAN';
```

	LAST_NAME	SALARY	MOD(SALARY,5000)
1	Weiss	8000	3000
2	Fripp	8200	3200
3	Kaufling	7900	2900
4	Vollman	6500	1500
5	Mourgos	5800	800

Funciones para trabajar con Fechas



Por defecto las fechas se visualizan: día, las tres primeras letras del mes y el año en dos dígitos (según el siglo)

La Base de Datos almacena la fecha en un formato interno: siglo, año, mes, día, horas, minutos y segundos

Cuando se consulta por una fecha en particular debe ir entre comillas simples

Las fechas se almacenan como números, por lo tanto se pueden realizar cálculos usando operadores aritméticos

Todas las funciones de fechas retornan un tipo de dato fecha excepto la función MONTHS_BETWEEN que retorna un valor numérico

SYSDATE

MONTHS_BETWEEN(*fecha1*,*fecha2*)

NEXT_DAY(*fecha*,*día*)

ADD_MONTHS(*fecha*,*n*)

LAST_DAY(*fecha*)

ROUND(*fecha*,['formato'])

TRUNC(*fecha*,['formato'])

- Ejemplo:

```
SELECT MONTHS_BETWEEN('01/ENE/2019','01/OCT/2018') "MONTHS_BETWEEN",  
       ADD_MONTHS('20/SEP/2019',6) "ADD_MONTHS",  
       NEXT_DAY('11/MAR/2019','DOMINGO') "NEXT_DAY",  
       LAST_DAY('01/FEB/2019') "LAST_DAY"  
FROM dual;
```

MONTHS_BETWEEN	ADD_MONTHS	NEXT_DAY	LAST_DAY
3	20/03/2020	17/03/2019	28/02/2019

- Ejemplo (asumiendo que la fecha actual (SYSDATE) es 10/10/2019):

```
SELECT ROUND(SYSDATE, 'MONTH'), ROUND(SYSDATE,'YEAR'),  
       TRUNC(SYSDATE,'MONTH'), TRUNC(SYSDATE,'YEAR')  
FROM dual;
```

ROUND(SYSDATE,'MONTH')	ROUND(SYSDATE,'YEAR')	TRUNC(SYSDATE,'MONTH')	TRUNC(SYSDATE,'YEAR')
01/10/2019	01/01/2020	01/10/2019	01/01/2019

- Ejemplo:

```
SELECT employee_id "ID EMPLEADO", first_name || ' ' || last_name "NOMBRE EMPLEADO",  
       ROUND(MONTHS_BETWEEN(SYSDATE,hire_date)/12) "AÑOS TRABAJADOS"  
FROM employees;
```

	ID EMPLEADO	NOMBRE EMPLEADO	AÑOS TRABAJADOS
1	100	Steven King	16
2	101	Neena Kochhar	14
3	102	Lex De Haan	19
4	103	Alexander Hunold	14
5	104	Bruce Ernst	12
6	105	David Austin	14
7	106	Valli Pataballa	14
8	107	Diana Lorentz	13
9	108	Nancy Greenberg	17
10	109	Daniel Faviet	17
.....			
104	203	Susan Mavris	17
105	204	Hermann Baer	17
106	205	Shelley Higgins	17
107	206	William Gietz	17

- Ejemplo:

```
SELECT employee_id "ID EMPLEADO", first_name || ' ' || last_name "NOMBRE EMPLEADO",  
       salary salario, ROUND(MONTHS_BETWEEN(SYSDATE,hire_date)/12) "AÑOS TRABAJADOS",  
       ROUND(salary*(ROUND(MONTHS_BETWEEN(SYSDATE,hire_date)/12)/100)  
       "PORC. SALARIO SEGUN AÑOS TRAB"  
FROM employees;
```

ID EMPLEADO	NOMBRE EMPLEADO	SALARIO	AÑOS TRABAJADOS	PORC. SALARIO SEGUN AÑOS TRAB
1	100 Steven King	24000	16	3840
2	101 Neena Kochhar	17000	14	2380
3	102 Lex De Haan	17000	19	3230
4	103 Alexander Hunold	9000	14	1260
5	104 Bruce Ernst	6000	12	720
6	105 David Austin	4800	14	672
7	106 Valli Pataballa	4800	14	672
8	107 Diana Lorentz	4200	13	546
9	108 Nancy Greenberg	12008	17	2041
10	109 Daniel Faviot	9000	17	1530
104	203 Susan Mavris	6500	17	1105
105	204 Hermann Baer	10000	17	1700
106	205 Shelley Higgins	12008	17	2041
107	206 William Gietz	8300	17	1411

Fecha - Fecha = Número de días

Fecha + Número = Fecha

Fecha - Número = Fecha

- Ejemplo:

```
SELECT last_name, hire_date, ROUND((SYSDATE - hire_date) / 7) "SEMANAS CONTRATADO",  
       hire_date - 2 "FECHA CONTRATO MENOS 2 DIAS",  
       hire_date + 10 "FECHA CONTRATO MAS 10 DIAS"  
FROM employees  
WHERE department_id = 90;
```

	LAST_NAME	HIRE_DATE	SEMANAS CONTRATADO	FECHA CONTRATO MENOS 2 DIAS	FECHA CONTRATO MAS 10 DIAS
1	King	17/06/2003	851	15/06/2003	27/06/2003
2	Kochhar	21/09/2005	733	19/09/2005	01/10/2005
3	De Haan	13/01/2001	978	11/01/2001	23/01/2001

Función EXTRACT para trabajar con Fechas

Obtiene la parte especificada (día, mes, año, horas, minutos, etc.) a partir de un valor de fecha y hora. El valor obtenido es numérico

No se puede usar la función EXTRACT para obtener HOUR, MINUTE o SECOND a partir de un valor de DATE

HOUR, MINUTE o SECOND se pueden extraer sólo desde valores TIMESTAMP

- Sintaxis:

```
EXTRACT ( { YEAR | MONTH | DAY | HOUR | MINUTE | SECOND }  
| { TIMEZONE_HOUR | TIMEZONE_MINUTE } | { TIMEZONE_REGION  
| TIMEZONE_ABBR } FROM { fecha | intervalo } )
```

Función EXTRACT para trabajar con Fechas

- Ejemplo:

```
SELECT first_name || ' ' || last_name || ' ingresó a la empresa el ' ||  
       EXTRACT (DAY FROM hire_date) || ' de ' || EXTRACT (MONTH FROM hire_date) ||  
       ' del año ' || EXTRACT(YEAR FROM hire_date) "INFORMACION DE CONTRATOS"  
FROM employees  
ORDER BY EXTRACT(YEAR FROM hire_date) DESC, last_name;
```

	INFORMACION DE CONTRATOS
1	Sundar Ande ingresó a la empresa el 24 de 3 del año 2008
2	Amit Banda ingresó a la empresa el 21 de 4 del año 2008
3	Girard Geoni ingresó a la empresa el 3 de 2 del año 2008
4	Douglas Grant ingresó a la empresa el 13 de 1 del año 2008
5	Charles Johnson ingresó a la empresa el 4 de 1 del año 2008
6	Sundita Kumar ingresó a la empresa el 21 de 4 del año 2008
7	David Lee ingresó a la empresa el 23 de 2 del año 2008
8	Steven Markle ingresó a la empresa el 8 de 3 del año 2008
9	Mattea Marvins ingresó a la empresa el 24 de 1 del año 2008
10	Hazel Philtanker ingresó a la empresa el 6 de 2 del año 2008
.....	
104	Shelley Higgins ingresó a la empresa el 7 de 6 del año 2002
105	Susan Mavris ingresó a la empresa el 7 de 6 del año 2002
106	Den Raphaely ingresó a la empresa el 7 de 12 del año 2002
107	Lex De Haan ingresó a la empresa el 13 de 1 del año 2001

- Se describieron las características generales de las Funciones SQL.
- Se describieron las características de las Funciones SQL que operan con una fila.
- Se describieron los tipos de Funciones que operan con una sola fila: de caracteres, numéricas, de fechas, de conversión y generales.
- Se explicó cómo convertir, reemplazar o manipular caracteres en sentencias SQL
- Se explicó cómo manipular números en sentencias SQL.
- Se explicó cómo efectuar operaciones con fechas en sentencias SQL.