

FUNCIONES CADENA									
ORACLE			MYSQL			POSTGRESQL			DESCRIPCIÓN
NOMBRE	EJEMPLO	RESULTADO	NOMBRE	EJEMPLO	RESULTADO	NOMBRE	EJEMPLO	RESULTADO	
ASCII	SELECT ASCII('V') FROM DUAL;	86	ASCII	SELECT ASCII('V');	86	ASCII	SELECT ASCII('x');	120	Devuelve el valor de código ASCII del carácter más a la izquierda de la cadena str. Devuelve 0 si str es una cadena vacía. Devuelve NULL si str es NULL
CHR	SELECT CHR(86) FROM DUAL;	V	CHAR	SELECT CHAR(86);	V	CHR	SELECT CHR(65);	A	Interpreta los argumentos como enteros y devuelve una cadena que consiste en los caracteres dados por los valores de los códigos ASCII de esos enteros. Los valores NULL se saltan
				SELECT CHAR(77,121,83,81,'76');	MySQL				
CONCAT	SELECT CONCAT('PEPE ', 'PEREZ') FROM DUAL;	PEPE PEREZ	CONCAT	SELECT CONCAT('My', 'S', 'QL');	MySQL				Devuelve la cadena resultante de concatenar los argumentos
			CONCAT_WS	SELECT CONCAT_WS(",","First name","Second name","Last Name");	First name,Second name,Last Name				
	SELECT ('ORACLE' '9IAS') FROM DUAL;	ORACLE 9IAS					SELECT 'Post' 'greSQL';	PostgreSQL	
LENGTH	SELECT LENGTH('PEPE PEREZ') FROM DUAL;	10	LENGTH, CHAR_LENGTH Y CHARACTER_LENGTH	SELECT LENGTH('text');	4	LENGTH, CHAR_LENGTH Y CHARACTER_LENGTH	SELECT CHAR_LENGTH('jose');	4	Numero de caracteres de una cadena
LOWER	SELECT LOWER('PEPE PEREZ') FROM DUAL;	pepe perez	LOWER	SELECT LOWER('MYSQL');	mysql	LOWER	SELECT LOWER('TOM');	tom	Devuelve la cadena str con todos los caracteres cambiados a minúsculas
			LCASE	SELECT LCASE('MYSQL');	mysql				
REPLACE	SELECT REPLACE('PEPE PEREZ','PE','PA') FROM DUAL;	PAPA PAREZ	REPLACE	SELECT REPLACE('mysql', 'm', 'M');	Mysql	REPLACE Y OVERLAY	SELECT REPLACE('Txxxxas' PLACING 'hom' FROM 2 FOR 4);	Thomas	Devuelve la cadena char cambiando cada aparición de 'buscar' por 'cambiar'. Por defecto la cadena de reemplazo es nula
			INSERT	SELECT INSERT('Quadratic', 3, 4, 'What');	QuWhattic		SELECT OVERLAY('TXXXxas' PLACING 'hom' FROM 2 FOR 4);		
INSTR	SELECT INSTR('pepe perez','per') FROM DUAL;	6	INSTR	SELECT INSTR('foobarbar', 'bar');	4	POSITION	SELECT POSITION('om' IN 'Thomas');	3	Devuelve la posición de la primera aparición de la subcadena substr dentro de la cadena str
			FIND_IN_SET	SELECT FIND_IN_SET('b','a,b,c,d');	2				
			POSITION	SELECT POSITION('CD', 'ABCDE');	3	STRPOS	SELECT STRPOS('high', 'ig');	2	
			LOCATE	SELECT LOCATE('bar', 'foobarbar');	4				

SUBSTR	SELECT SUBSTR('PEPE PEREZ', 6, 3) FROM DUAL;	PER	SUBSTRING Y MID	SELECT SUBSTRING('MySQL con Clase',7);	con Clase	SUBSTRING	SELECT SUBSTRING('Thomas' FROM 2 FOR 3);	hom	Devuelve una subcadena de char de n caracteres a partir de la posicion m
---------------	--	-----	------------------------	--	-----------	------------------	--	-----	--

FUNCIONES CADENA									
ORACLE			MYSQL			POSTGRESQL			DESCRIPCIÓN
NOMBRE	EJEMPLO	RESULTADO	NOMBRE	EJEMPLO	RESULTADO	NOMBRE	EJEMPLO	RESULTADO	
TRIM	SELECT TRIM(' PEPE PEREZ ') FROM DUAL;	PEPE PEREZ	TRIM	SELECT TRIM(' bar ');	bar	TRIM	SELECT TRIM(' Tom ');	Tom	Esta función permite quitar los blancos por delante o por detras (o ambos a la vez) de una cadena. Con LEADING, quitamos por delante. Con TRAILING quitamos por detras. Con BOTH o no poniendo nada, quitamos de ambos lados.
	SELECT TRIM(BOTH 'X' FROM 'XXXPEPE PEREZXXX') FROM DUAL;	PEPE PEREZ		SELECT TRIM(BOTH 'x' FROM 'xxxbarxxx');	bar		SELECT TRIM(BOTH 'x' FROM 'xTomxx');	Tom	
UPPER	SELECT UPPER('Pepe Perez') FROM DUAL;	PEPE PEREZ	UPPER	SELECT UPPER('Hej')	HEJ	UPPER	SELECT UPPER('tom');	TOM	Devuelve la cadena en mayusculas
			UCASE	SELECT LCASE('Hej')	HEJ				
LPAD	SELECT LPAD('6940004',9,'0') FROM DUAL;	006940004	LPAD	SELECT LPAD('hi',4,'??');	??hi	LPAD	SELECT LPAD('hi',4,'??');	??hi	Rellena la cadena por la izquierda hasta la longitud n con el caracter definido, por defecto es blanco
RPAD	SELECT RPAD('6940004',9,'0') FROM DUAL;	694000400	RPAD	SELECT RPAD('hi',4,'??');	hi??	RPAD	SELECT RPAD('hi',4,'??');	hi??	Rellena la cadena por la derecha hasta la longitud n con el caracter definido, por defecto es blanco
			REPEAT	SELECT REPEAT('MySQL', 3);	MySQLMySQLMySQL	REPEAT	SELECT REPEAT('Pg', 4);	PgPgPgPg	Devuelve una cadena que consiste en la cadena str repetida count veces. Si count <= 0, devuelve una cadena vacía. Devuelve NULL si str o count son NULL:
			REVERSE	SELECT REVERSE('abc');	cba				Devuelve la cadena str con el orden de los caracteres invertido
			SPACE	SELECT CONCAT('hola', SPACE(6), 'Baby');	hola Baby				Devuelve una cadena que consiste en N caracteres espacio
INITCAP	SELECT INITCAP('pepe perez') FROM DUAL;	Pepe Perez				INITCAP	SELECT INITCAP('hi THOMAS');	Hi Thomas	Devuelve la cadena con cada palabra empezando con mayuscula y el resto en minusculas
TRANSLATE	SELECT TRANSLATE ('EL PERRO DE SAN ROQUE...', 'RP', 'GT') FROM DUAL;	EL TEGGO DE SAN GOQUE...				TRANSLATE	SELECT TRANSLATE('12345', '14', 'ax');	a23x5	Permite reemplazar caracteres en una cadena

FUNCIONES DE FECHA									
ORACLE			MYSQL			POSTGRESQL			DESCRIPCIÓN
NOMBRE	EJEMPLO	RESULTADO	NOMBRE	EJEMPLO	RESULTADO	NOMBRE	EJEMPLO	RESULTADO	
SYSDATE	SELECT SYSDATE FROM DUAL;	25/08/09	CURDATE CURRENTDATE	SELECT CURDATE();	2009-08-25	CURRENT_DATE	SELECT CURRENT_DATE;	2009-08-25	Devuelve la fecha actual del servidor
SYSDATE	SELECT TO_CHAR(SYSDATE, 'HH24:MM:SS') FROM DUAL;	11:08:44	CURTIME CURRENTTIME	SELECT CURTIME();	11:08:44	CURRENT_TIME	SELECT CURRENT_TIME;	11:08:44.781-04:30	Devuelve la hora actual del servidor
ADD_MONTHS	SELECT ADD_MONTHS(TO_DATE('31/10/2007','DD/MM/YYYY'), 4) FROM DUAL;	29/02/2008	DATE_ADD	SELECT DATE_ADD('1998-01-02', INTERVAL 31 DAY);	1998-02-02	+	SELECT DATE '2001-09-28' + INTEGER '7';	2001-10-05	Permite agregar intervalos de Años, Meses, Días, Horas, Segundos y Minutos
				SELECT DATE_ADD('1997-12-31 23:59:59', INTERVAL 1 SECOND);	1998-01-01 00:00:00		SELECT DATE '2001-09-28' + INTERVAL '1 HOUR';	2001-09-28 01:00:00	
			ADDDATE	SELECT ADDDATE('1998-01-02', INTERVAL 31 DAY);	1998-02-02		SELECT DATE '2001-09-28' + TIME '03:00';	2001-09-28 03:00:00	
				SELECT ADDDATE('1998-01-02', 31);	1998-02-02		SELECT INTERVAL '1 DAY' + INTERVAL '1 HOUR';	1 day 01:00:00	
			ADDTIME	SELECT ADDTIME('1997-12-31 23:59:59.999999', "1 1:1:1.000002");	1998-01-02 01:01:01.0000 01		SELECT TIMESTAMP '2001-09-28 01:00' + INTERVAL '23 HOURS';	2001-09-29 00:00:00	
				SELECT ADDTIME("01:00:00.999999", "02:00:00.999998");	03:00:01.9999 97		SELECT TIME '01:00' + INTERVAL '3 HOURS';	04:00:00	
MONTHS_BETWEEN	SELECT MONTHS_BETWEEN(TO_DATE('31/10/2007','DD/MM/YYYY'), TO_DATE('01/06/2007','DD/MM/YYYY')) FROM DUAL;	4,97	DATEDIFF	SELECT DATEDIFF('1997-12-31 23:59:59','1997-12-30');	1	-	SELECT - INTERVAL '23 HOURS';	-23:00:00	Permite restar intervalos de fechas y horas
				SELECT DATEDIFF('1997-12-31 23:59:59','1997-12-30');	-30		SELECT DATE '2001-10-01' - DATE '2001-09-28';	3	
			DATE_SUB	SELECT DATE_SUB('1998-01-01 00:00:00', INTERVAL '1 1:1:1' DAY_SECOND);	1997-12-30 22:58:59		SELECT DATE '2001-10-01' - INTEGER '7';	2001-09-24	
				SELECT DATE_SUB('1998-01-02', INTERVAL 31 DAY);	1997-12-02		SELECT DATE '2001-09-28' - INTERVAL '1 HOUR';	2001-09-27 23:00:00	
			-	SELECT '1998-01-01' - INTERVAL 1 SECOND;	1997-12-31 23:59:59		SELECT TIMESTAMP '2001-09-28 23:00' - INTERVAL '23 HOURS';	2001-09-28 00:00:00	
			DAYNAME	SELECT DAYNAME('1998-02-05');	Thursday				Devuelve el nombre del día de la semana para una fecha
			MONTHNAME	SELECT MONTHNAME('1998-02-05');	February				Devuelve el nombre del mes para la fecha date

FUNCIONES MATEMATICAS									
ORACLE			MYSQL			POSTGRESQL			DESCRIPCIÓN
NOMBRE	EJEMPLO	DEVUELVE	RESULTADO	EJEMPLO	RESULTADO	NOMBRE	EJEMPLO	RESULTADO	
ABS	SELECT ABS(-89) FROM DUAL;	89	ABS	SELECT ABS(-32);	32	ABS	SELECT ABS(-17.4);	17.4	Valor absoluto de un numero
CEIL	SELECT CEIL(45.1242) FROM DUAL;	46	CEILING Y CEIL	SELECT CEILING(1.23);	2	CEILING Y CEIL	SELECT CEIL(1.23);	2	Redondea n hasta el valor superior
FLOOR	SELECT FLOOR(45.9542) FROM DUAL;	45	FLOOR	SELECT FLOOR(1.95);	1	FLOOR	SELECT FLOOR(1.95);	1	Redondea n al valor inferior
GREATEST	SELECT GREATEST(1,3,6,89,56,3) FROM DUAL;	89				GREATEST	SELECT GREATEST(1,3,6,89,56,3);	89	Devuelve el mayor de una lista de expresiones
LEAST	SELECT LEAST(1,3,6,89,56,3) FROM DUAL;	1				LEAST	SELECT LEAST(1,3,6,89,56,3);	1	Devuelve el menor de una lista de expresiones
MOD	SELECT MOD(13,5) FROM DUAL;	3	MOD Y %	SELECT MOD(234, 10); SELECT 234 % 10;	4	MOD	SELECT MOD(9,4);	1	Devuelve el resto de dividir M entre N. Devuelve M si N es 0.
ROUND	SELECT ROUND(45.1242, 2) FROM DUAL;	45.12	ROUND	SELECT ROUND(-1.58);	-2	ROUND	SELECT ROUND(42.5);	43	Devuelve el argumento X, redondeado al entero más cercano. Con dos argumentos redondea a un número con D decimales
	SELECT ROUND(-45.8242) FROM DUAL;	-46		SELECT ROUND(1.298, 2);	1.30		SELECT ROUND(42.4382, 2);	42.44	
			RAND	SELECT RAND();	0.4076504236 1585	RANDOM	SELECT RANDOM();	0.0826392685 994509	Devuelve un valor aleatorio entre 0.0 y 1.0
TRUNC	SELECT TRUNC(45.1242, 2) FROM DUAL;	45.12	TRUNCATE	SELECT TRUNCATE(1.223,1);	1.2	TRUNC	SELECT TRUNC(42.4382, 2);	42.43	Devuelve el número X, truncado a D decimales. Si D es 0, el resultado no tendrá punto decimal o parte fraccionaria

FUNCIONES DE CONVERSION Y FORMATOS									
ORACLE			MYSQL			POSTGRESQL			DESCRIPCIÓN
NOMBRE	EJEMPLO	RESULTADO	NOMBRE	EJEMPLO	RESULTADO	NOMBRE	EJEMPLO	RESULTADO	
TO_CHAR, TO_DATE Y TO_NUMBER	SELECT TO_CHAR(sysdate) FROM DUAL;	###	FORMAT	SELECT FORMAT(12332.123456, 4);	12,332.1235	TO_CHAR, TO_DATE, TO_NUMBER Y TO_TIMESTAMP	SELECT TO_CHAR(CURRENT_TIMES TAMP, 'HH12:MI:SS');	11:05:04	Funciones que permiten dar formatos a los campos Numericos y Date
	SELECT TO_CHAR(sysdate, 'dd/mm/yyyy') FROM DUAL;	15/06/2001		SELECT FORMAT(12332.1,4);	12,332.1000		SELECT TO_CHAR(125.45, '999');	125	
	SELECT TO_CHAR(sysdate, 'Mon') FROM DUAL;	JUN		SELECT FORMAT(12332.2,0);	12,332		SELECT TO_CHAR(125.8::real, '999D9')	125,8	
	SELECT TO_CHAR(sysdate, 'Month') FROM DUAL;	JUNIO		SELECT FORMAT(12332.123456, 2);	12,332.12		SELECT TO_CHAR(-125.8, '999D99S');	125,80-	
	SELECT TO_CHAR(123.456, '09999') FROM DUAL;	00123	DATE_FORMAT	SELECT DATE_FORMAT('1997-10-04 22:23:00', '%H:%i:%s');	22:23:00		SELECT TO_CHAR(current_timestamp, 'HH12:MI:SS');	03:37:05	
	SELECT TO_CHAR(123.456, '09999.9') FROM DUAL;	00123.5		SELECT DATE_FORMAT('1997-10-04 22:23:00', '%W %M %Y');	Saturday October 1997		SELECT TO_DATE('05 Dec 2000', 'DD Mon YYYY');	2000-12-05	
	SELECT TO_CHAR(123456, 'FM999,999,999') FROM DUAL;	123,46		SELECT DATE_FORMAT('1997-10-04 22:23:00', '%D %y %a %d %m %b %j');	4th 97 Sat 04 10 Oct 277		SELECT TO_NUMBER('12,454.8-', '99G999D9S');	-12454.8	
	SELECT TO_DATE('31/10/2007','DD/M M/YYYY') FROM DUAL;	31/10/2007		SELECT DATE_FORMAT('1997-10-04 22:23:00', '%H %k %l %r %T %S %w');	22 22 10 10:23:00 PM 22:23:00 00 6		SELECT TO_TIMESTAMP('05 Dec 200 0', 'DD Mon YYYY')	2000-12-05 00:00:00-04	
	SELECT TO_NUMBER('123') FROM DUAL;	123		SELECT DATE_FORMAT('1999-01-01', "%X %V");	1998 52		SELECT TO_TIMESTAMP(1284352323);	2010-09-13 00:02:03- 04:30	
CASE Y DECODE	SELECT CASE WHEN ESTADO <= 0 THEN 'bien' WHEN ESTADO >= 1 AND ESTADO <= 5 THEN 'regular'ELSE 'mal' END FROM T_PEDIDOS;	Dependen del Valor de la Variable (ESTADO)	CASE	SELECT CASE X WHEN 1 THEN "uno" WHEN 2 THEN "varios" ELSE "muchos" END;	Dependen del Valor de la Variable (X)	CASE	SELECT A, CASE WHEN A = 1 THEN 'one' WHEN A = 2 THEN 'two' ELSE 'other' END FROM test;	Dependen del Valor de la Variable (A)	Evalua diferentes expresiones para dar un valor de salida
			IF	SELECT IF(1>2,2,3);	3				
				SELECT IF(1<2,'yes','no');	yes				
				SELECT IF(STRCMP('test','test1'),'no',y es');	no				
	SELECT DECODE(ESTADO, 0, 'bien', 1, 'regular', 'mal') FROM T_PEDIDOS;		IFNULL	SELECT IFNULL(1,0);	1				
				SELECT IFNULL(NULL,10);	10				
				SELECT IFNULL(1/0,10);	10				

				SELECT IFNULL(1/0,'yes');	yes				
--	--	--	--	---------------------------	-----	--	--	--	--

OTRAS FUNCIONES									
ORACLE			MYSQL			POSTGRESQL			DESCRIPCIÓN
NOMBRE	EJEMPLO	RESULTADO	NOMBRE	EJEMPLO	RESULTADO	NOMBRE	EJEMPLO	RESULTADO	
CONSULTA DE VISTA	SELECT INSTANCE_NAME FROM SYS.V_\$INSTANCE;	IPASME	DATABASE	SELECT DATABASE();	Nombre Base de Datos	CURRENT_DATA BASE	SELECT CURRENT_DATABASE();	posgres	Devuelve el nombre de la base de datos actual
	SELECT SID FROM V\$SESSION WHERE USERNAME = 'XXX';	25	CONNECTION_ID	SELECT CONNECTION_ID();	40				Devuelve el ID (ID del hilo) de una conexión. Cada conexión tiene su propio y único ID
USER	SELECT USER FROM DUAL;	BASJMEENZE	USER, SESSION_USER Y SYSTEM_USER	SELECT USER(); SELECT SUBSTRING_INDEX(USER(), "@",1);	davida@localh davida	USER , SESSION_USER Y CURRENT_USER	SELECT CURRENT_USER;	posgres	Devuelve el nombre de usuario y el del host para el que está autenticada la conexión actual
CONSULTA DE VISTA	SELECT VERSION FROM SYS.V_\$INSTANCE;	9.2.0.1.0	VERSION	SELECT VERSION();	5.0.16-nt	VERSION	SELECT VERSION();	PostgreSql 8.4.0	Devuelve una cadena que indica la versión del servidor de la Base de Datos
						INET_CLIENT_ADDR	SELECT INET_CLIENT_ADDR();	127.0.0.1	Devuelve la Dirección IP del Cliente
						INET_CLIENT_PORT	SELECT INET_CLIENT_PORT();	3128	Devuelve el Puerto del Cliente
CONSULTA DE VISTA	SELECT HOST_NAME FROM SYS.V_\$INSTANCE;	ipasmeprod.lo caldomain				INET_SERVER_ADDR	SELECT INET_SERVER_ADDR();	172.16.0.120	Devuelve la Dirección IP del Servidor
						INET_SERVER_PORT	SELECT INET_SERVER_PORT();	5432	Devuelve el Puerto del Servidor
CONSULTA DE VISTA	SELECT MACHINE FROM V\$SESSION WHERE USERNAME = 'BASJMEENZE';	IPASME.INT\ BAS02							Devuelve el Nombre de la Maquina
	SELECT TERMINAL FROM V\$SESSION WHERE USERNAME = 'BASJMEENZE';	BAS02							
	SELECT PROGRAM FROM V\$SESSION WHERE USERNAME = 'BASJMEENZE';	sqlplusw.exe							Devuelve el Nombre del programa que se esta ejecutando en la Base de Datos
						GENERATE_SERIES	SELECT * FROM GENERATE_SERIES(2,10,2);	2 4 6 8 10	Permite Generar series como lo hace excel

FUNCIONES DE GRUPO									
ORACLE			MYSQL			POSTGRESQL			DESCRIPCIÓN
NOMBRE	EJEMPLO	RESULTADO	NOMBRE	EJEMPLO	RESULTADO	NOMBRE	EJEMPLO	RESULTADO	
AVG	SELECT AVG(Columna) FROM Tabla;	134.5678	AVG	SELECT STUDENT_NAME, AVG(TEST_SCORE) FROM STUDENT GROUP BY STUDENT_NAME;	8.54	AVG	SELECT "COLUMNA", AVG("COLUMNAX") FROM "ESQUEMA"."TABLA" GROUP BY "COLUMNA";	8.54	Determina la media del valor de una columna
COUNT	SELECT COUNT(*) FROM TABLA;	10	COUNT	SELECT COUNT(*) FROM TABLA;	15	COUNT	SELECT COUNT(*) FROM "ESQUEMA"."TABLA";	20	Cuenta el numero de valores en un columna o el numero de filas en una tabla
	SELECT COUNT(DISTINCT COLUMNA) FROM TABLA;	24		SELECT COUNT(DISTINCT COLUMNA) FROM TABLA;	24		SELECT COUNT(DISTINCT "COLUMNA") FROM "ESQUEMA"."TABLA";	24	
MIN	SELECT MIN(COLUMNA) FROM TABLA;	10	MIN	SELECT MIN(COLUMNA) FROM TABLA;	11	MIN	SELECT MIN("IdEmpleado") FROM "public"."Empleados";	1	Determina el mayor valor de una columna
MAX	SELECT MAX(COLUMNA) FROM TABLA;	15	MAX	SELECT MAX(COLUMNA) FROM TABLA;	32	MAX	SELECT MAX("IdEmpleado") FROM "public"."Empleados";	4	Determina el mayor valor de una columna
SUM	SELECT SUM(COLUMNA) FROM TABLA;	150	SUM	SELECT SUM(COLUMNA) FROM TABLA;	120	SUM	SELECT SUM("IdEmpleado") FROM "public"."Empleados";	1255	Devuelve la suma del campo de la Tabla indicado
			GROUP_CONCAT	SELECT IdPropietario, GROUP_CONCAT(Forma) FROM pagopropietario WHERE IdPropietario = 10 GROUP BY IdPropietario;	10 A, D, D, E, D, C				Devuelve una cadena con la concatenación de los valores de un grupo

INSTRUCCIÓN	ORACLE	MYSQL	POSTGRESQL	DESCRIPCIÓN
	EJEMPLO			
INSERT	INSERT INTO T_PEDIDOS (CODPEDIDO,ESTADO) VALUES (125,2);	INSERT INTO GENTE (NOMBRE, FECHANAC) VALUES ('FULANO','1974-04-12');	INSERT INTO "PRODUCTS" ("PRODUCT_NO", "NAME", "PRICE") VALUES (2, 'CHEESE 1', 19.99);	Inserta un registro en una tabla, especificando los campos de la Tabla
		INSERT INTO GENTE SET NOMBRE='FULANO', FECHANAC='1954-12-07';		
	INSERT INTO T_PEDIDOS VALUES (125,2,'PEPE');	INSERT INTO GENTE VALUES ('MENGANO', '1974-02-03');	INSERT INTO "PRODUCTS" VALUES (1, 'Cheese', 9.99);	Inserta un registro en una tabla, sin especificar los campos de la misma
		INSERT INTO GENTE VALUES ('TULANO','2000-12-02'), ('PEGANO','1993-02-10');	INSERT INTO "PRODUCTS" VALUES (1, 'CHEESE', 9.99), (2, 'CHEESE TREES', 10.99);	Inserta varios registros en una tabla (No aplica en oracle)
	INSERT INTO T_PEDIDOS (ESTADO) VALUES (2);	INSERT INTO GENTE VALUES ('MENGANO', DEFAULT);	INSERT INTO PRODUCTS (PRODUCT_NO, NAME, PRICE) VALUES (1, 'CHEESE', DEFAULT);	Inserta un registro en una tabla, tomando los valores por defecto, en caso de MySql y PostgreSql se coloca la palabra clave DEFAULT y en caso de oracle se especifican los campos a insertar pero se omite los campos que tiene valores por defecto
			INSERT INTO PRODUCTS DEFAULT VALUES;	
	INSERT INTO T_PEDIDOS_BAK SELECT * FROM T_PEDIDOS WHERE CODPEDIDO < 100;	INSERT INTO GENTE1 SELECT * FROM GENTE WHERE NOMBRE = 'FULANO';	INSERT INTO "PRODUCTS1" SELECT * FROM "PRODUCTS" WHERE "PRODUCT_NO" = 1;	Insertar registro en una tabla partiendo de una subconsulta SELECT
REPLACE		REPLACE INTO GENTE (NOMBRE, FECHANAC) VALUES ('TULANO','2000-12-02'), ('PEGANO','1993-02-10');		Existe una sentencia REPLACE, que es una alternativa para INSERT, que sólo se diferencia en que si existe algún registro anterior con el mismo valor para una clave primaria o única, se elimina el viejo y se inserta el nuevo en su lugar.
UPDATE	UPDATE T_PEDIDOS SET NOMBRE = NULL;	UPDATE CIUDAD SET POBLACION = POBLACION * 1.10;	UPDATE "PRODUCTS" SET "PRICE" = 10;	Actualiza todos los valores de una columna de la tabla
	UPDATE T_PEDIDOS SET NOMBRE = 'JUAN' WHERE CODPEDIDO = 125;	UPDATE CIUDAD SET POBLACION = POBLACION * 1.10 WHERE CIUDAD = 'MADRID';	UPDATE "PRODUCTS" SET "PRICE" = 11, "NAME" = 'CHEESE 1' WHERE "PRODUCT_NO" = 1;	Actualiza un valor de una columnas para un subconjunto de filas de una tabla
	UPDATE T_PEDIDOS SET NOMBRE = 'JUAN', ESTADO = 1 WHERE CODPEDIDO = 125;	UPDATE CIUDAD SET NOMBRE = 'ESPAÑA', POBLACION = POBLACION * 1.10 WHERE CIUDAD = 'MADRID';	UPDATE "PRODUCTS" SET "PRICE" = 11, "NAME" = 'CHEESE 1' WHERE "PRODUCT_NO" = 1;	Actualiza varios valores de columnas para un subconjunto de filas de una tabla
DELETE	DELETE FROM T_PEDIDOS;	DELETE FROM CIUDAD;	DELETE FROM "PRODUCTS";	Borra todos los registros de una tabla
	DELETE FROM T_PEDIDOS WHERE CODPEDIDO = 125;	DELETE FROM CIUDAD WHERE CIUDAD = 'ESPAÑA';	DELETE FROM "PRODUCTS" WHERE "PRICE" = 10;	Borra una o más filas de una tabla, dependiendo de la condición WHERE
TRUNCATE	TRNCATE TABLE T_PEDIDOS;	TRUNCATE TABLE CIUDAD;	TRUNCATE TABLE "PRODUCTS";	Borra todos los registros de una tabla
SELECT	SELECT * FROM T_PEDIDOS;	SELECT * FROM CIUDAD;	SELECT * FROM "PRODUCTS";	La selección sobre una tabla consiste en elegir un subconjunto de filas que cumplan (o no) algunas condiciones determinadas, las instrucción SELECT se complemente con las siguientes instrucciones: WHERE, LIMIT, GROUP BY, HAVING, ORDER BY, etc.
	SELECT A, B + C FROM TABLE1;	SELECT A, B + C FROM TABLE1;	SELECT A, B + C FROM "TABLE1";	
	SELECT DISTINCT NOMBRE FROM T_PEDIDOS;	SELECT DISTINCT FECHA FROM GENTE;	SELECT DISTINCT "PRICE" FROM "PRODUCTS";	
	SELECT MAX(ESTADO) FROM T_PEDIDOS;	SELECT MAX(NOMBRE) FROM GENTE;	SELECT MAX("PRICE") FROM "PRODUCTS";	
	SELECT MIN(ESTADO) FROM T_PEDIDOS;	SELECT MIN(NOMBRE) FROM GENTE;	SELECT MIN("PRICE") FROM "PRODUCTS";	

COMMIT: Guarda los cambios de la transacción en curso (solo Oracle)

ROLLBACK: Deshace los cambios de la transacción en curso (solo Oracle)

	SELECT AFIL_CEDULA, FLOOR((MONTHS_BETWEEN(SYSDATE, AFIL_FECHA_NACIMIENTO) / 12)) AS EDAD FROM REGIS_AFILIADO WHERE AFIL_CEDULA = '011558593';	SELECT NOMBRE, FECHA, (DATEDIFF(CURRENT_DATE(), FECHA) / 365) AS EDAD FROM GENTE;		DI, etc.
--	---	---	--	----------

INSTRUCCIÓN	ORACLE	MYSQL	POSTGRESQL	DESCRIPCIÓN
	EJEMPLO			
SELECT		SELECT * FROM GENTE WHERE NOMBRE = "Mengano";		de filas que cumplan (o no) algunas condiciones determinadas, las instrucción SELECT se complemente con las siguientes instrucciones: WHERE, LIMIT, GROUP BY, HAVING, ORDER BY
		SELECT * FROM GENTE WHERE FECHA >= "1986-01-01" AND FECHA < "2000-01-01";		
		SELECT NOMBRE, FECHA FROM GENTE GROUP BY FECHA;		
		SELECT FECHA, COUNT(*) AS CUENTA FROM GENTE GROUP BY FECHA;		
		SELECT CIUDAD, MAX(TEMPERATURA) FROM MUESTRAS GROUP BY CIUDAD HAVING MAX(TEMPERATURA)>16;		
		SELECT * FROM GENTE ORDER BY FECHA;		
		SELECT * FROM GENTE ORDER BY FECHA DESC;		
		SELECT * FROM GENTE LIMIT 3;		