Base de datos 03-10-22

Sintaxis:

INSERT INTO nombre tabla ([ítem de dato] , ) VALUES ([VALORES])

Rojo palabras claves.

SELECT [\*,[ ítem de datos, ] , ‘texto’, operación, función ej: YEAR [NOW()] AS [ALIAS-NOMBRE QUE LE DAMOS]

FROM nombre tabla

WHERE condiciones (LIKE, IN BETWEEN, OPERADORES LOGICOS ( =, - , > <. !=));

Order by: la forma de ordenar el resultado/salida, tiene el atributo que puede ir ascendente o descendente del ítem de datos. (ASC/DESC, no pueden ir los dos, solo uno, cuando no se especifica, es en orden ascendente).

Limit: condiciona la cantidad de filas que dan como resultado, puede operar con dos parámetros o uno, solo condiciona la cantidad de datos que devuelve. Con el primer parámetro indicamos desde donde, y con el segundo la cantidad, ej (0, 10- o 0,20).

SUM: Devuelve la suma de los valores seleccionados.

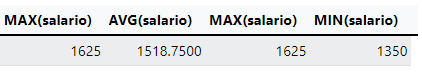
MAX: Devuelve el valor máximo de un ítem de dato.

MIN: Devuelve el mínimo valor de un ítem de dato.

AVG: Promedio de unos ítems de datos seleccionados.

Ejercicio.

1. Listar salario, máximo, mínimo y promedio, de los empleados cuyo salario sea menor a 2500 y que sean oficio ‘vendedor’.



1. Listar los primeros 5 : apellidos, oficios, salarios, comisiones y fechas de alta de los empleados que tengan salario entre 2mil y 5mil ordenados por oficio en forma descendente y apellido en forma ascendente



BASE DE DATOS 04/10.

HAVING: Si la condición es para una función del grupo o de LOS grupos, se utiliza HAVING.

Ejercicios

1. Determinar por cada departamento de los empleados que contengan una letra A en el apellido, la cantidad de empleados y el total de salarios.

2. Determinar el salario mínimo, máximo y promedio por cada oficio.

3. Mostrar cantidad de empleados, salario total, y salario máximo por cada oficio de los empleados, con un salario superior a 2mil, para aquellos oficios que tengan un salario promedio superior a 1500.

INNER: Este va a mostrar solamente los que coinciden

LEFT: Le da prioridad a la tabla que está a su izquierda.

RIGHT: Le da prioridad a la tabla de la derecha, inclusive los de la derecha que no tengan equivalencia con la tabla de la izquierda.

INNER JOIN LEFT RIGHT

En el inner join, es lo que comparten solamente. En el left, es lo que comparten más lo de la izquierda, y en el right, es lo que comparten y lo de la derecha.

Ejercicios.

1. Listar los empleados de oficio vendedor mostrando el nombre del departamento y la localidad a la que pertenece.
2. Listar los empleados del departamento ventas ordenado por salario en forma descendente
3. Determinar la cantidad de empleados que trabajen en una localidad que empiece con la letra M.

18/10/22

Repaso.

SELECT \*(VA A TENER UN RESULTADO, Y LO VA A MOSTRAR, LO QUE QUIERO QUE SALGA COMO RESPUESTA DE MI CONSULTA).

FROM (SIRVE PARA DETERMINAR SOBRE QUE TABLA, O DE CUAL, UNA O MAS, DE DONDE SE VA A SELECCIONAR. CUANDO NECESITO MAS DE UNA, TENGO QUE HACER LA RELACION EN EL WHERE, O CON INNER JOIN EN EL FROM).

WHERE (SIRVE PARA ESTABLECER UNA CONDICION DE MANERA TAL QUE LAS FILAS QUE CUMPLAN CON LA CONDICION SE SELECCIONEN, Y LAS QUE NO, NO).

GROUP BY (POR CADA O PARA QUE…. USAMOS GROUP BY, SE RELACIONA GENERALMENTE A UNA FUNCION DE CONTEO O DE SUMAS, RESTAS, ETC, SOLO LO USAMOS CUANDO PIDE ALGUN CRITERIO DE AGRUPACION, EJ: POR OFICIO, POR DPTO, ETC).

HAVING (ES UN FILTRO, NORMALMENTE VA CON EL COUNT (\*), CONDICIONA LOS GRUPOS QUE AGRUPAMOS EN EL GROUP BY, LE DECIMOS QUE GRUPOS QUIERE QUE TOME Y CUALES NO).

EL ‘WHERE’ Y EL HAVING, HACEN LO MISMO, PERO LA DIFERENCIA ENTRE ELLOS ES QUE EL WHERE SE UTILIZA SOLO CUANDO LA CONDICION SE APLICA SOBRE LOS ITEMS DE LA TABLA DEL FROM.

ORDER BY (SOLO SIRVE PARA ORDENAR CIERTA INFORMACION).

LIMIT (LIMITAMOS PARA QUE NOS MUESTRE CIERTA CANTIDAD DE FILAS O COLUMNAS, O QUE NOS MUESTRE A PARTIR DE TAL FILA O COLUMNA, ETC).

UPDATE (MODIFICA EL VALOR DE UNO O MAS ATRIBUTOS, DE UNA O MAS TABLAS, SE PUEDE HACER EL INNER JOIN EN EL UPDATE).