



GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL PROCEDIMIENTO DESARROLLO CURRICULAR GUÍA DE APRENDIZAJE

1. IDENTIFICACIÓN DE LA GUIA DE APRENIZAJE: DERSQL_T2D_01

- **Denominación del Programa de Formación:** ANÁLISIS Y DESARROLLO DE SISTEMAS DE INFORMACIÓN
- **Código del Programa de Formación:** 228106 VERSIÓN 102
- **Nombre del Proyecto:** CONSTRUCCIÓN DE UN SISTEMA DE INFORMACIÓN QUE CUMPLA CON LOS REQUERIMIENTOS DEL CLIENTE EN PROCESOS QUE SE LLEVEN A CABO EN EL SECTOR PRODUCTIVO DEL DEPARTAMENTO DE CALDAS..
- **Fase del Proyecto:** ANÁLISIS
- **Actividad de Proyecto:** DEFINIR LOS REQUERIMIENTOS DEL SISTEMA DE INFORMACIÓN, TENIENDO EN CUENTA LA NECESIDAD DEL CLIENTE.
- **Competencia:** ANALIZAR LOS REQUISITOS DEL CLIENTE PARA CONSTRUIR EL SISTEMA DE INFORMACION
- **Resultados de Aprendizaje Alcanzar:**
 - 436541 - Análisis: Valorar la incidencia de los datos en los procesos del macrosistema, tomando como referente el diccionario de datos y las miniespecificaciones, para la consolidación de los datos que intervienen, de acuerdo con parámetros establecidos
- **Duración de la Guía:** 33 horas

2. PRESENTACION

Seguro que en más de una ocasión hemos escuchado hablar de las bases de datos, y también de modo intrínseco nos hemos hecho nuestra propia concepción de lo que son. El SQL (Estándar Query Language) es un lenguaje estándar de programación para manipular la información que se almacena en las bases de datos.

El lenguaje SQL se utiliza para acceder y manipular datos en cualquier base de datos del mercado, como por ejemplo, para las bases de datos MySQL, Oracle, DB2, SQL Server, Access.

En esta guía, los aprendices conocerán las instrucciones básicas que tiene este lenguaje y desarrollarán habilidades en el manejo de la información con una base de datos

3. FORMULACION DE LAS ACTIVIDADES DE APRENDIZAJE

- **Descripción de la(s) Actividad(es)**



SERVICIO NACIONAL DE APRENDIZAJE SENA
Procedimiento de Desarrollo Curricular
GUÍA DE APRENDIZAJE

Actividades de Aprendizaje:

- * Realizar diagramas Entidad - Relación, empleando una herramienta CASE.
- * Realizar diccionarios de datos.
- * Reconocer los diferentes conceptos de un Modelo Relacional (Tabla, Campo, Llave Primaria, Llave Foránea, Relaciones)
- * Realizar modelos relacionales a partir de diagramas entidad - relación.

Actividad de reflexión inicial

Los aprendices deben consultar lo siguiente:

- Cuál es la diferencia entre un lenguaje de Programación y un lenguaje de consulta de Bases de datos

Luego de la actividad de consulta se hará una socialización con aprendices e instructor

Actividades de contextualización e identificación de conocimientos necesarios para el aprendizaje

Pasos para realizar el Diagrama Entidad Relación, Paso a Modelo Relación y Diagrama de Tablas Final

Los aprendices deberán tener en cuenta los siguientes pasos, los cuales les servirán como un resumen a grandes rasgos de lo que se debe hacer en cada parte del proceso:

Pasos para realizar Diagrama Entidad Relación (DER)

- 1) Leer muy bien el enunciado del problema e identificar la lista de posibles Entidades
- 2) Realizar el Diagrama Entidad Relación con las Entidades (rectángulos), Relaciones (Rombos) y Atributos (Óvalos)
- 3) Toda entidad debe tener un atributo identificador (el cual se convertirá en llave primaria). Se deberá subrayar este atributo
- 4) Realizar cardinalidad de todas las relaciones

Pasos a Modelo Relacional

- 1) Toda entidad se convertirá en una tabla



SERVICIO NACIONAL DE APRENDIZAJE SENA
Procedimiento de Desarrollo Curricular
GUÍA DE APRENDIZAJE

- 2) En relaciones de **1 a M**, la llave primaria de la entidad que tiene el **1** pasará como llave foránea de la entidad que tiene **M**
- 3) En relaciones de **M a M**, se creará una nueva tabla con la llave primaria de cada una de las entidades que intervienen en la relación. También irá en esta nueva tabla, los atributos que puedan existir en la relación
- 4) En relaciones de **1 a 1**, se elige una de las dos entidades y se toma la llave primaria. Esa llave primaria pasará a ser llave foránea de la otra entidad

Construcción del Diagrama Final de Tablas

- 1) Por último se hará el diagrama final de las tablas de la base de datos
- 2) Para construir el diagrama final de tablas se tendrá como insumo las tablas que se identificaron en el paso a modelo relacional

Actividades de apropiación del conocimiento (Conceptualización y Teorización).

Diccionario de datos

El diccionario de datos es un listado organizado de todos los datos que pertenecen a un sistema. El objetivo de un diccionario de datos es dar precisión sobre los datos que se manejan en un sistema, evitando así malas interpretaciones o ambigüedades. Los diccionarios de datos son buenos complementos a los diagramas de flujo de datos, los diagramas entidad-relación, etc.

Para una base de datos representa un listado organizado de todos los datos pertinentes al sistema con definiciones precisas y rigurosas para que tanto el usuario como el analista tengan un entendimiento en común. Contiene las características lógicas de los datos que se van a utilizar en el sistema, incluyendo nombre, descripción, alias, tipo y características del contenido y así como también llaves primarias y foráneas.

Ejemplo:

Tabla: Curso:

Nombre	Tipo	Longitud	Null	Dominio	PK	FK	Descripción
Id_curso	INT	11		[0-9]	x		Llave única para curso
Nombre	varchar	50		[a-z]			Nombre del curso
fechaInicio	date			"aaaa-mm-dd"			Fecha inicio curso
fechaFin	date			"aaaa-mm-dd"			Fecha fin curso
numeroEstud	tinyint	3		[0-9]			Números estudiantes en el curso



SERVICIO NACIONAL DE APRENDIZAJE SENA
Procedimiento de Desarrollo Curricular
GUÍA DE APRENDIZAJE

Como vimos en el anterior diccionario de datos, hay una columna “tipo” para especificar el tipo de dato del campo en la base de datos. Existen muchos tipos de datos y pueden variar entre los diferentes sistemas gestores de bases de datos

Actividad 01

Los aprendices deben consultar lo siguiente:

- ¿Cuáles son los tipos de datos que maneja MySQL?

Luego de realizarse la actividad, se hará una socialización con aprendices e instructor donde se resolverán dudas. Se complementará esta actividad con la lectura del siguiente link

<http://download.nust.na/pub6/mysql/doc/refman/5.0/es/numeric-types.html>

Actividad 02

Ahora los aprendices deberán aplicar el diccionario de datos a 3 ejercicios realizados en la actividad anterior de Diagrama Entidad Relación y paso a modelo relacional según las instrucciones y el formato en Excel brindados por el instructor, dicho archivo se encuentra en la plataforma en la carpeta “Material de apoyo”

Actividad 03

Los aprendices deberán realizar búsquedas en internet y en esta guía para consultar los siguientes términos y funciones SQL y realizar un resumen sobre la funcionalidad y adicional poner un ejemplo en cada uno. Esta actividad deberá ser realizada en el cuaderno:

- | | |
|---------------------|-----------------|
| 3.1 DML | 3.5 SELECT |
| 3.2 DDL | 3.6 UPDATE |
| 3.3 CREATE DATABASE | 3.7 DELETE |
| 3.4 CREATE TABLE | 3.8 INSERT INTO |



SERVICIO NACIONAL DE APRENDIZAJE SENA

Procedimiento de Desarrollo Curricular

GUÍA DE APRENDIZAJE

SQL

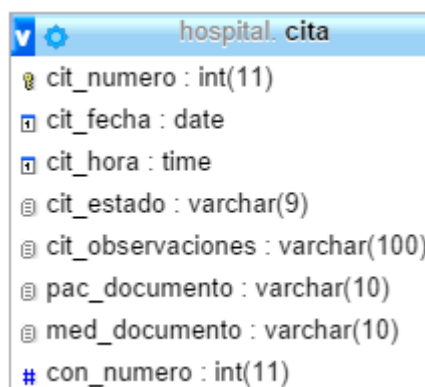
El SQL es un lenguaje estructurado y un estándar ANSI para el acceso y manipulación de los datos de cualquier base de datos.

El SQL se compone de sentencias SQL, cada una con una utilidad diferente, como por ejemplo:

- Creación de una base de datos (CREATE DATABASE)
- Creación de una tabla (CREATE TABLE)
- Creación de una vista (CREATE VIEW)
- Creación de un índice de una tabla (CREATE INDEX)
- Creación de procedimientos almacenados (CREATE PROCEDURE)
- Creación de disparadores (CREATE TRIGGER)
- Consultar los datos almacenados en una tabla (SELECT)
- Insertar datos en una tabla (INSERT)
- Modificar datos ya existentes en una tabla (UPDATE)
- Borrar datos almacenados en una tabla (DELETE)
- Dar permiso de acceso a los datos de una tabla (GRANT)
- Eliminar permisos de acceso a datos de una tabla (REVOKE)
- Finalizar la transacción de una sentencia SQL (COMMIT)
- Retroceder la transacción de una sentencia SQL (ROLLBACK).

Un sistema de base de datos suele contener varias bases de datos. Cada base de datos suele contener varias tablas. Las tablas almacenan los datos organizados por filas. Cada fila contiene varios campos.

Cada campo tiene una serie de propiedades, como por ejemplo, el nombre del campo, su longitud, tipo de dato que se va a almacenar, etc.



Esta es una tabla Cita que pertenece a la BD Hospital



Sentencias SQL:

Las sentencias SQL no son sensibles a mayúsculas y minúsculas, es decir, 'SELECT' y 'select' son dos palabras iguales para SQL.

Algunos sistemas de bases de datos necesitan un punto y coma después de cada sentencia SQL para ejecutarse correctamente

`CREATE DATABASE mibasedeatos;` (Esta sentencia crea una base de datos)

Este punto y coma también puede servir para separar una sentencia SQL de otra sentencia SQL en la misma línea.

`CREATE DATABASE mibasededatos; CREATE TABLE mitabla1;` (la segunda sentencia crea la tabla mitabla1 en la base de datos mibasededatos)

Las sentencias SQL se pueden agrupar por funcionalidades:

1.- Lenguaje de definición de datos (DDL).

DDL está compuesto por sentencias para la creación (CREATE), modificación (ALTER) y borrado (DROP) de los componentes principales de una base de datos:

- base de datos (DATABASE)
- tablas (TABLE)
- vistas (VIEW)
- índices (INDEX)
- procedimientos almacenados (PROCEDURE)
- disparadores (TRIGGER).

2.- Lenguaje de manipulación de datos (DML).

DML está compuesto por sentencias que sirven para :

- consultar (SELECT)
- insertar (INSERT)
- modificar (UPDATE)
- borrar (DELETE)

3.- Lenguaje de control de datos (DCL).



SERVICIO NACIONAL DE APRENDIZAJE SENA
Procedimiento de Desarrollo Curricular
GUÍA DE APRENDIZAJE

DCL está compuesto por sentencias SQL para controlar las funciones de administración :

- Confirmar la operacion (COMMIT)
- Retroceder la operacion (ROLLBACK)
- Dar permisos (GRANT)
- Quitar permisos (REVOKE)

1. LENGUAJE DE DEFINICIÓN DE DATOS

El lenguaje de definición de datos DDL, permite generar las estructuras de las tablas y las relaciones que van a existir entre ellas. Adicional a las funcionalidades mencionadas anteriormente, se puede realizar las siguientes

- Generar Claves Primarias
- Generar Claves Secundarias
- Generar relaciones por medio de restricciones de claves foráneas
- Generar Actualizaciones y Eliminaciones en cascada
- Cambiar nombres a la tablas y campos
- Determinar si un campo acepta valores null o no

Crear una Tabla

Para crear una tabla por medio de la consola del SMBDS (Sistema Manejador de Bases de Datos) se utiliza la siguiente sintaxis.

```
CREATE TABLE example (  
    id INT,  
    data VARCHAR(100)  
);
```

La sentencia anterior crea en la base de datos seleccionada, una tabla llamada **example** con dos campos

- **Id** de tipo INT (entero)
- **data** de tipo Varchar con una longitud de 100 caracteres.

Para crear una tabla que contenga un campo autonúmerico se hace de la siguiente manera

```
CREATE TABLE example_autoincrement (  
    id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,  
    data VARCHAR(100)  
);
```

El campo **id**, es un campo de tipo entero que no va a permitir nulos y adicional es auto incrementable, y por ser así, debe considerarse como una clave Primaria (Primary Key). Adicionando las palabras **Primary Key** se puede determinar el campo que va a ser la clave principal de la tabla.



SERVICIO NACIONAL DE APRENDIZAJE SENA
Procedimiento de Desarrollo Curricular
GUÍA DE APRENDIZAJE

La creación de la tabla, va a depender del motor de bases de datos que se esté utilizando, por ejemplo:

MySQL:

```
CREATE TABLE Persons
(
P_Id int NOT NULL,
LastName varchar(255) NOT NULL,
FirstName varchar(255),
Address varchar(255),
City varchar(255),
PRIMARY KEY (P_Id)
)
```

SQL Server / Oracle / MS Access:

```
CREATE TABLE Persons
(
P_Id int NOT NULL PRIMARY KEY,
LastName varchar(255) NOT NULL,
FirstName varchar(255),
Address varchar(255),
City varchar(255)
)
```

Si en una tabla se desea generar una Clave Primaria Compuesta se debe hacer de la siguiente manera:

MySQL / SQL Server / Oracle / MS Access:

```
CREATE TABLE Persons
(
P_Id int NOT NULL,
LastName varchar(255) NOT NULL,
FirstName varchar(255),
Address varchar(255),
City varchar(255),
CONSTRAINT pk_PersonID PRIMARY KEY (P_Id,LastName)
)
```

Si después de haber generado la tabla, se desean hacer cambios sobre la misma, entonces se debe utilizar una sentencia **ALTER** para hacer las respectivas modificaciones, ejemplo:



SERVICIO NACIONAL DE APRENDIZAJE SENA
Procedimiento de Desarrollo Curricular
GUÍA DE APRENDIZAJE

MySQL / SQL Server / Oracle / MS Access:

```
ALTER TABLE Persons  
ADD CONSTRAINT pk_PersonID PRIMARY KEY (P_Id, LastName)
```

Otro ejemplo, borrar la clave primaria de una tabla

MySQL:

```
ALTER TABLE Persons  
DROP PRIMARY KEY
```

SQL Server / Oracle / MS Access:

```
ALTER TABLE Persons  
DROP CONSTRAINT pk_PersonID
```

Para la generación de las claves foráneas, se debe utilizar la siguiente sintaxis:

Tabla Ordenes, teniendo en cuenta la tabla personas, la relación sería

1 Persona realiza Muchas Ordenes

MySQL:

```
CREATE TABLE Orders  
(  
O_Id int NOT NULL,  
OrderNo int NOT NULL,  
P_Id int,  
PRIMARY KEY (O_Id),  
FOREIGN KEY (P_Id) REFERENCES Persons(P_Id)  
)
```

SQL Server / Oracle / MS Access:

```
CREATE TABLE Orders  
(  
O_Id int NOT NULL PRIMARY KEY,  
OrderNo int NOT NULL,  
P_Id int FOREIGN KEY REFERENCES Persons(P_Id)  
)
```



SERVICIO NACIONAL DE APRENDIZAJE SENA
Procedimiento de Desarrollo Curricular
GUÍA DE APRENDIZAJE

Siempre al generar la clave foránea, se debe tener en cuenta hacer la referencia a la tabla donde se encuentra la clave primaria.

Actividad 04

Los aprendices deberán realizar la creación de la base de datos de los 3 ejercicios de la Actividad 02 utilizando las sentencias vistas de DDL. El instructor dará las indicaciones sobre cómo utilizar la plataforma para realizar dicho proceso. Se harán uso de los programas **XAMPP** y **PHPMyAdmin**

2.- Lenguaje de manipulación de datos (DML).

A continuación se presentan las instrucciones del Lenguaje DML que son utilizadas para manipular el CRUD (Create Read Update Delete) de una aplicación.

Sentencia INSERT

La sentencia INSERT INTO se utiliza para insertar nuevas filas en una tabla. Es posible insertar una nueva fila en una tabla de dos formas distintas:

La primera forma no especifica los nombres de las columnas en las que se insertan los datos, sólo sus valores:

```
INSERT INTO table_name  
VALUES (value1,value2,value3,...);
```

La segunda forma especifica tanto los nombres de columna y los valores que se insertan:

```
INSERT INTO table_name (column1,column2,column3,...)  
VALUES (value1,value2,value3,...);
```

Ejemplo:

Dada la siguiente tabla personas:

nombre	apellido1	apellido2
ANTONIO	PEREZ	GOMEZ
LUIS	LOPEZ	PEREZ
ANTONIO	GARCIA	BENITO



SERVICIO NACIONAL DE APRENDIZAJE SENA
Procedimiento de Desarrollo Curricular
GUÍA DE APRENDIZAJE

Si queremos insertar una nueva fila en la tabla personas, lo podemos hacer con cualquiera de las dos sentencias siguientes:

```
INSERT INTO personas  
VALUES ('PEDRO', 'RUIZ', 'GONZALEZ')
```

```
INSERT INTO personas (nombre, apellido1, apellido2)  
VALUES ('PEDRO', 'RUIZ', 'GONZALEZ')
```

Cualquiera de estas sentencias anteriores produce que se inserte una nueva fila en la tabla personas, quedando así dicha tabla:

nombre	apellido1	apellido2
ANTONIO	PEREZ	GOMEZ
LUIS	LOPEZ	PEREZ
ANTONIO	GARCIA	BENITO
PEDRO	RUIZ	GONZALEZ

En la segunda sentencia hay que tener en los campos de la tabla que no permiten valores nulos, porque esos campos son los mínimos que deben aparecer en la sentencia insert into.

Las sentencias insert mencionadas anteriormente, solo insertan un solo registro, si lo que se desea es insertar varios registros utilizando solamente una consulta insert se debe realizar de la siguiente manera:

```
INSERT INTO personas (nombre, apellido1, apellido2)  
VALUES ('LUIS', 'ROJAS', 'GONZALEZ'),  
      ('ANA', 'POLAR', 'VARGAS'),  
      ('CARLOS', 'DIAZ', 'RIOS');
```

Actividad 05

Ahora los aprendices deberán insertar datos en cada uno de los 3 ejercicios de la Actividad 02 según instrucciones del instructor. Esta actividad deberá ser subida a la plataforma BlackBoard en la actividad: “Entrega Evidencias Actividad 07”, la cual incluirá lo realizado en las **anteriores actividades**



SERVICIO NACIONAL DE APRENDIZAJE SENA
Procedimiento de Desarrollo Curricular
GUÍA DE APRENDIZAJE

Sentencia SELECT

La sentencia SELECT es utilizada para seleccionar datos de una base de datos. El resultado se almacena en una tabla de resultados, llamado el conjunto de resultados.

Sintaxis SELECT SQL

```
SELECT column_name, column_name  
FROM table_name;
```

y

```
SELECT * FROM table_name;
```

Para los ejemplos, tendremos la siguiente tabla de personas denominada “personas”.

Estos son los datos almacenados en la tabla “personas”

nombre	apellido1	apellido2
ANTONIO	PEREZ	GOMEZ
ANTONIO	GARCIA	BENITO
LUIS	LOPEZ	PEREZ

Si queremos consultar todos los datos de la tabla “personas”

```
SELECT * FROM personas
```

Este será el resultado:

nombre	apellido1	apellido2
ANTONIO	PEREZ	GOMEZ
ANTONIO	GARCIA	BENITO
LUIS	LOPEZ	PEREZ

Si queremos consulta todos los nombres y primer apellido de todas las personas

```
SELECT nombre, apellido1 FROM personas
```

Este será el resultado:



SERVICIO NACIONAL DE APRENDIZAJE SENA
Procedimiento de Desarrollo Curricular
GUÍA DE APRENDIZAJE

nombre	apellido1
ANTONIO	PEREZ
ANTONIO	GARCIA
LUIS	LOPEZ

Clausula where en Select

La cláusula WHERE se utiliza para hacer filtros en las consultas, es decir, seleccionar solamente algunas filas de la tabla que cumplan una determinada condición.

El valor de la condición debe ir entre comillas simples ''.

Por ejemplo:

Seleccionar las personas cuyo nombre sea ANTONIO

```
SELECT * FROM personas  
WHERE nombre = 'ANTONIO'
```

nombre	apellido1	apellido2
ANTONIO	PEREZ	GOMEZ
ANTONIO	GARCIA	BENITO

Los **operadores AND y OR** se utilizan para filtrar resultados con 2 condiciones.

El operador **AND** mostrará los resultados cuando se cumplan las 2 condiciones.

```
Condición1 AND condición2
```

El operador **OR** mostrará los resultados cuando se cumpla alguna de las 2 condiciones.

```
Condicion1 OR condicion2
```

En la tabla personas

nombre	apellido1	apellido2
ANTONIO	PEREZ	GOMEZ
ANTONIO	GARCIA	BENITO
LUIS	LOPEZ	PEREZ



SERVICIO NACIONAL DE APRENDIZAJE SENA
Procedimiento de Desarrollo Curricular
GUÍA DE APRENDIZAJE

La siguiente sentencia (ejemplo AND) dará el siguiente resultado:

```
SELECT * FROM personas  
WHERE nombre = 'ANTONIO'  
AND apellido1 = 'GARCIA'
```

nombre	apellido1	apellido2
ANTONIO	GARCIA	BENITO

La siguiente sentencia (ejemplo OR) dará el siguiente resultado:

```
SELECT * FROM personas  
WHERE nombre = 'ANTONIO'  
OR apellido1 = 'GARCIA'
```

nombre	apellido1	apellido2
ANTONIO	PEREZ	GOMEZ
ANTONIO	GARCIA	BENITO

También se pueden combinar AND y OR, como el siguiente ejemplo:

```
SELECT * FROM personas  
WHERE nombre = 'ANTONIO'  
AND (apellido1 = 'GARCIA' OR apellido1 = 'LOPEZ')
```

Nombre	apellido1	apellido2
ANTONIO	GARCIA	BENITO

Actividad 06

Los aprendices deberán realizar el taller del documento: [Taller 01 SQL.docx](#) el cual se encuentra en la carpeta “Material de Apoyo”, y que permitirá de una manera práctica aplicar los conceptos DDL y DML vistos anteriormente.

Esta actividad deberá ser subida a la plataforma BlackBoard en la actividad: “Entrega evidencias Actividad 06”



SERVICIO NACIONAL DE APRENDIZAJE SENA
Procedimiento de Desarrollo Curricular
GUÍA DE APRENDIZAJE

Actividad 09

Los aprendices deberán realizar búsquedas en internet para consultar los siguientes términos y funciones SQL y realizar un resumen sobre la funcionalidad en documento de Word, adicional poner un ejemplo en cada uno. Al final de la actividad el instructor

- | | |
|---------------|--------------|
| 1. INNER JOIN | 8. AVG() |
| 2. LEFT JOIN | 9. SUM() |
| 3. RIGHT JOIN | 10. ORDER BY |
| 4. LIKE | 11. GROUP BY |
| 5. COUNT | 12. LIMIT |
| 6. MAX() | 13. BETWEEN |
| 7. MIN() | |

Clausula Order By

ORDER BY se utiliza para ordenar los resultados de una consulta, según el valor de la columna especificada.

Por defecto, se ordena de forma ascendente (ASC) según los valores de la columna.

Si se quiere ordenar por orden descendente se utiliza la palabra DES

```
SELECT nombre_columna(s)
FROM nombre_tabla
ORDER BY nombre_columna(s) ASC|DESC
```

Por ejemplo, en la tabla personas :

Nombre	apellido1	apellido2
ANTONIO	PEREZ	GOMEZ
LUIS	LOPEZ	PEREZ
ANTONIO	GARCIA	BENITO



SERVICIO NACIONAL DE APRENDIZAJE SENA
Procedimiento de Desarrollo Curricular
GUÍA DE APRENDIZAJE

```
SELECT nombre, apellido1  
FROM personas  
ORDER BY apellido1 ASC
```

Esta es la consulta resultante:

Nombre	apellido1
LUIS	LOPEZ
ANTONIO	GARCIA
ANTONIO	PEREZ

Ejemplo de ordenación descendiente (DESC)

```
SELECT nombre, apellido1  
FROM personas  
ORDER BY apellido1 DESC
```

Esta es la consulta resultante:

nombre	apellido1
ANTONIO	PEREZ
ANTONIO	GARCIA
LUIS	LOPEZ

Ambiente Requerido

- Ambiente de Sistemas 01 con 30 sillas, 30 computadores
- **Materiales**
 - Computadores (30)
 - Televisor (1)
 - Resma tamaño carta (1)
 - Marcadores (2)
 - Lápiz (1)
 - Lapicero (1)

4. ACTIVIDADES DE EVALUACIÓN

Tome como referencia las técnicas e instrumentos de evaluación citados en la guía de Desarrollo Curricular



SERVICIO NACIONAL DE APRENDIZAJE SENA
Procedimiento de Desarrollo Curricular
GUÍA DE APRENDIZAJE

Evidencias de Aprendizaje	Criterios de Evaluación	Técnicas e Instrumentos de Evaluación
Evidencias de Conocimiento : Respuestas con procedimiento del instrumento de evaluación Evidencias de Desempeño: Asistencia y participación activa en las diferentes actividades propuestas Evidencias de Producto: Respuestas y procedimiento de los talleres realizados	Modela la base de datos, a partir de la valoración de la información obtenida en el diccionario de datos y el análisis de los procesos, de acuerdo con las necesidades del sistema de información requerido.	Interrogatorio. Cuestionario: DER_C_01 Observación: DER_D_01 Valoración del Producto: DER_P_01

5. GLOSARIO DE TERMINOS

Entidad: Representa una "cosa" u "objeto" del mundo real con existencia independiente, es decir, se diferencia unívocamente de otro objeto o cosa, incluso siendo del mismo tipo, o una misma entidad.

Ejemplo: Una casa (Aunque sea exactamente igual a otra, aún se diferenciará en su dirección).

Atributos: Los atributos son las características que definen o identifican a una entidad. Estas pueden ser muchas, y el diseñador solo utiliza o implementa las que considere más relevantes. Los atributos son las propiedades que describen a cada entidad en un conjunto de entidades.

En un conjunto de entidades, cada entidad tiene valores específicos asignados para cada uno de sus atributos, de esta forma, es posible su identificación unívoca

Relaciones: conexiones entre dos o más conjuntos de entidades. Describe cierta dependencia entre entidades o permite la asociación de las mismas. Una relación tiene sentido al expresar las entidades que relaciona.

Diagrama o Modelo Entidad Relación (DER): denominado por sus siglas en inglés, E-R "Entity relationship", o del español DER "Diagrama de Entidad Relación") es una herramienta para el modelado de



SERVICIO NACIONAL DE APRENDIZAJE SENA
Procedimiento de Desarrollo Curricular
GUÍA DE APRENDIZAJE

datos que permite representar las entidades relevantes de un sistema de información así como sus interrelaciones y propiedades

Bases de Datos (BD): es un banco de información que contienen datos relativos a diversas temáticas y categorizados de distinta manera, pero que comparten entre sí algún tipo de vínculo o relación que busca ordenarlos y clasificarlos en conjunto.

Base de datos relacional: es un tipo de base de datos (BD) que cumple con el modelo relacional (el modelo más utilizado actualmente para implementar las BD ya planificadas). Permite establecer interconexiones o relaciones entre los datos (que están guardados en tablas), y a través de dichas conexiones relacionar los datos de ambas tablas, de ahí proviene su nombre: "modelo relacional"

Sistema de información: es un conjunto de elementos orientados al tratamiento y administración de datos e información, organizados y listos para su uso posterior, generados para cubrir una necesidad o un objetivo.

6. REFERENTES BIBLIOGRAFICOS

Construya o cite documentos de apoyo para el desarrollo de la guía, según lo establecido en la guía de desarrollo curricular

- Documento técnico: "[ModeloER.pdf](#)"
- Documento técnico: "[Diagrama Entidad Relación - ejemplos - conceptos.pdf](#)"
- Documento técnico: "[ModeloRelacional-guia-estudio.pdf](#)"
- Documento técnico: "[Base de datos relacionales.pdf](#)"
- Documento técnico: "[ElModeloRelacional.pdf](#)"
- <http://es.slideshare.net/josecuartas/transformar-modelo-entidad-relacion-a-modelo-logico>
- <http://es.slideshare.net/nayis2010/modelo-entidad-relacin-extendido-18067292>

7. CONTROL DEL DOCUMENTO

	Nombre	Cargo	Dependencia	Fecha
Autor (es)	Julian Humberto Salazar Pineda	Instructor/Contratista	Centro de Procesos Industriales y construcción	01-03-2021



SERVICIO NACIONAL DE APRENDIZAJE SENA
Procedimiento de Desarrollo Curricular
GUÍA DE APRENDIZAJE

8. CONTROL DE CAMBIOS (diligenciar únicamente si realiza ajustes a la guía)

	Nombre	Cargo	Dependencia	Fecha	Razón del Cambio
Autor (es)					