



#>/<>

HACK A BOSS

<CODE YOUR TALENT>

IMPLEMENTACIÓN

#>/<>

HACK A BOSS

<CODE YOUR TALENT>

*"La tecnología, bien utilizada, es uno
de los mayores catalizadores sociales
que han existido nunca"*

2020 EDITION



ÍNDICE

01.

Intro

- Definición
- Modelos
- BBDD relacionales

02.

Diseño

- Proceso
- Modelo Entidad-Relación
- Diagramas ER

03.

Implementación

- Creación y modificación
- Consultas
- Transacciones

#>/<>

HACK A BOSS

<CODE YOUR TALENT>

REPASO

BBDD Relacionales

- Se compone de 1 o más **tablas**.
- El nombre de las tablas es único, no pueden existir 2 con el mismo nombre.
- Cada tabla es un conjunto de **campos** (columnas) y **registros** (filas).
- Se pueden crear relaciones entre tablas usando claves primarias y foráneas.
- Las claves **primarias** son la clave principal de un registro.
- Las claves **foráneas** son claves es una tabla hija que se corresponden con la clave primaria de la tabla padre.

#>/<>

HACK A BOSS

<CODE YOUR TALENT>

RDBMS

RDBMS/SGBDR

Relational Database Management System o sistema de gestión de bases de datos relacional.

Software que permiten el almacenamiento, modificación y extracción de la información en una base de datos.

#>/<>

HACK A BOSS

<CODE YOUR TALENT>



ORACLE®

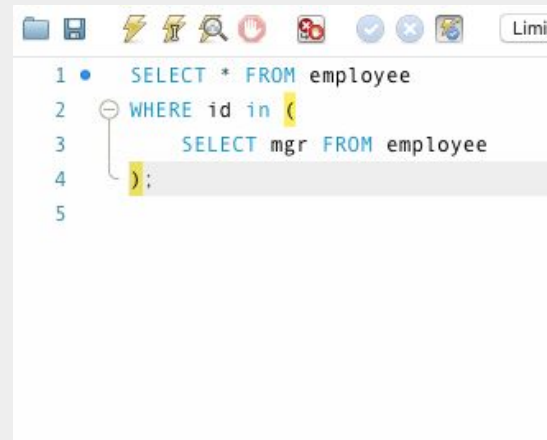
DATABASE

SQL

SQL

Structured Query Language.

Un lenguaje proporcionado por el sistema de gestión de base de datos que permite a los usuarios llevar a cabo las tareas de consulta o manipulación de los datos, organizados por el modelo de datos adecuado



```
1 • SELECT * FROM employee
2 WHERE id in (
3     SELECT mgr FROM employee
4 );
5
```

IMPLEMENTACIÓN

Modelo Relacional

Una vez seleccionado el SGDB

- Convertir el Modelo conceptual en una estructura dependiente del SGDB.
- En BD relacionales: Modelo Relacional
- Definir las tablas, campos y restricciones
- Definir las relaciones
- Resolver posibles problemas

#>/<>

HACK A BOSS

<CODE YOUR TALENT>

SQL

Data Manipulation Language

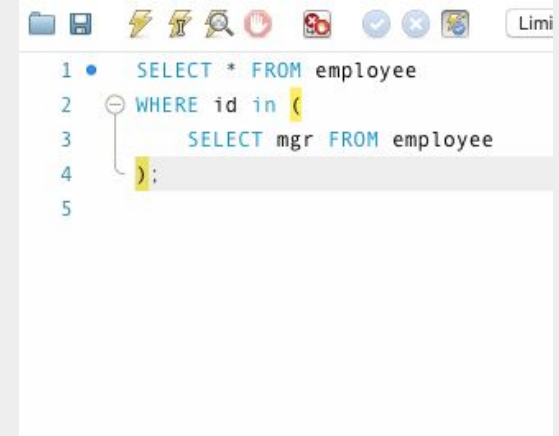
Se emplean para trabajar con los datos

- SELECT -> Consultar datos
- INSERT -> Insertar datos
- UPDATE -> Actualizar datos
- DELETE -> Eliminar datos

Data Definition Language

Se emplean para modificar la base de datos

- CREATE -> Crear objetos
- ALTER -> Modificar objetos
- DROP -> Eliminar objetos



```
1 • SELECT * FROM employee
2 WHERE id in (
3     SELECT mgr FROM employee
4 );
5
```

#>/<>

HACK A BOSS

<CODE YOUR TALENT>

MySQL DATA TYPES

DATE TYPE	SPEC	DATA TYPE	SPEC
CHAR	String (0 - 255)	INT	Integer (-2147483648 to 2147483647)
VARCHAR	String (0 - 255)	BIGINT	Integer (-9223372036854775808 to 9223372036854775807)
TINYTEXT	String (0 - 255)	FLOAT	Decimal (precise to 23 digits)
TEXT	String (0 - 65535)	DOUBLE	Decimal (24 to 53 digits)
BLOB	String (0 - 65535)	DECIMAL	"DOUBLE" stored as string
MEDIUMTEXT	String (0 - 16777215)	DATE	YYYY-MM-DD
MEDIUMBLOB	String (0 - 16777215)	DATETIME	YYYY-MM-DD HH:MM:SS
LONGTEXT	String (0 - 4294967295)	TIMESTAMP	YYYYMMDDHHMMSS
LOBLOB	String (0 - 4294967295)	TIME	HH:MM:SS
TINYINT	Integer (-128 to 127)	ENUM	One of preset options
SMALLINT	Integer (-32768 to 32767)	SET	Selection of preset options
MEDIUMINT	Integer (-8388608 to 8388607)	BOOLEAN	TINYINT(1)

CREATE

```
CREATE DATABASE databasename;
```

```
CREATE TABLE table name (  
    column1 datatype,  
    column2 datatype,  
    column3 datatype,  
    ....  
);
```

```
CREATE TABLE new table name AS  
    SELECT column1, column2,...  
    FROM existing_table_name  
    WHERE ....;
```

```
CREATE DATABASE testDB;
```

```
CREATE TABLE Persons (  
    PersonID int,  
    LastName varchar(255),  
    FirstName varchar(255),  
    Address varchar(255),  
    City varchar(255)  
);
```

```
CREATE TABLE TestTable AS  
    SELECT customername, contactname  
    FROM customers;
```

DROP

```
DROP DATABASE databasename;
```

```
DROP DATABASE testDb;
```

```
DROP TABLE table_name;
```

```
DROP TABLE Shippers;
```

```
TRUNCATE TABLE table_name;
```

```
TRUNCATE TABLE Shippers;
```

```
#>/<>
```

HACK A BOSS

<CODE YOUR TALENT>

ALTER

```
ALTER TABLE table name  
ADD column_name datatype;
```

```
ALTER TABLE Customers  
ADD Email varchar(255);
```

```
ALTER TABLE table name  
DROP COLUMN column_name;
```

```
ALTER TABLE Customers  
DROP COLUMN Email;
```

```
ALTER TABLE table name  
MODIFY COLUMN column_name datatype;
```

```
ALTER TABLE Customers  
MODIFY COLUMN Email varchar(196);
```

CONSTRAINTS

Restricciones aplicadas a columnas o tablas para controlar o limitar los datos que insertan o modifican.

Los tipos más comunes son:

- NOT NULL - Asegura que no puede tener valores NULL
- UNIQUE - Asegura que todos los valores son únicos (no se repiten)
- PRIMARY KEY - Valor identificador (combina NOT NULL y UNIQUE)
- FOREIGN KEY - Identificador de una relación con otra tabla
- CHECK - Asegura que se cumple una condición concreta: CHECK (Age>=18)
- DEFAULT - Define un valor por defecto que se usará cuando no se especifica ningún valor
- INDEX - Crea un índice que se usará para consultar datos de forma más rápida

CONSTRAINTS

```
CREATE TABLE table name (  
    column1 datatype constraint,  
    column2 datatype constraint,  
    column3 datatype constraint,  
    ....  
);
```

```
ALTER TABLE table name  
MODIFY column1 datatype constraint
```

```
CREATE TABLE Persons (  
    ID int NOT NULL,  
    LastName varchar(255) NOT NULL,  
    FirstName varchar(255) NOT NULL,  
    Age int  
);
```

```
ALTER TABLE Persons  
MODIFY Age int NOT NULL;
```



#>/<>

HACK A BOSS

<CODE YOUR TALENT>

#<THANX!>

#>/<>

HACK A BOSS

<CODE YOUR TALENT>

+34 919 04 23 63

www.hackaboss.com

Av.Linares Rivas 50-51, 15005, A Coruña

2020 EDITION