

GMQTECH
2023-2024

**ACCESO A DATOS
PROYECTO**

**PROYECTO A CARGO DE
SERGIO MÁRQUEZ
DURANTE, 2ºDAM**



ÍNDICE

INTRODUCCIÓN	03
MODELO DE DATOS	04
CREACIÓN DE TABLAS	05
RELACIONES ENTRE TABLAS	06
CREACIÓN DE VISTAS	07
PROCEDIMIENTOS ALMACENADOS	14
VERIFICACIÓN DE LOS RESULTADOS	17

INFORME PROYECTO

ACCESO A DATOS

INTRODUCCIÓN

Como parte de la investigación de la base de datos, he llevado a cabo un proyecto que se enfoca en el diseño e implementación de un sistema de gestión escolar en MySQL. Este proyecto incluye la creación de tablas clave, el desarrollo de procedimientos almacenados y la implementación de transacciones para optimizar la gestión de datos en entornos educativos y con sus respectivas vistas.

OBJETIVO DEL PROYECTO

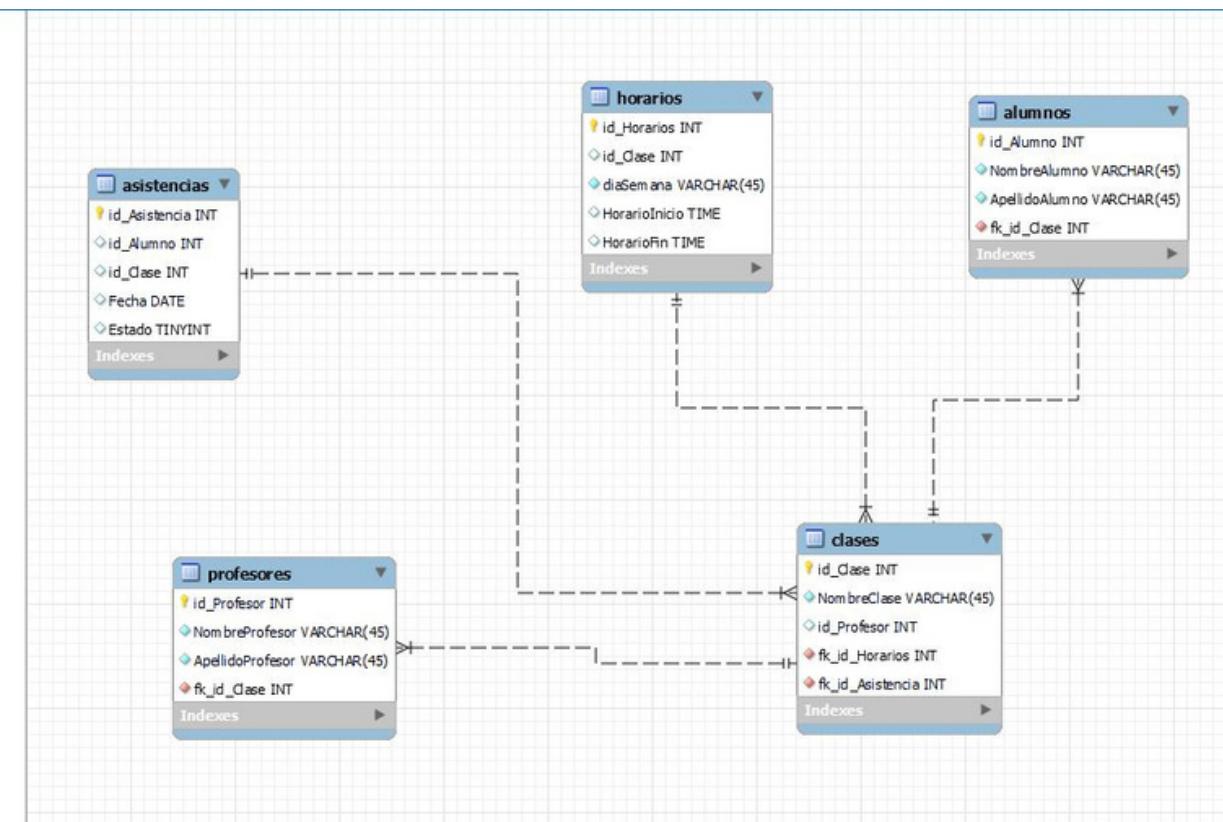
El principal objetivo de este proyecto es proporcionar una plataforma eficiente y robusta para la gestión integral de las instituciones educativas. El sistema cubre aspectos importantes como información detallada sobre estudiantes y profesores, tareas y horarios de clase y seguimiento detallado de la asistencia de los estudiantes.

INFORME PROYECTO

ACCESO A DATOS

MODELO DE DATOS

El primer paso es diseñar el modelo de datos. Así podemos identificar todas las entidades principales y sus relaciones.



INFORME PROYECTO

ACCESO A DATOS

CREACIÓN DE TABLAS

A continuación mostraremos todos las tablas con sus respectivos campos y sus diferentes claves foraneas.

TABLAS:

ALUMNOS:

ID_ALUMNO (PK)
NOMBREALUMNO
APELLOALUMNO
PROFESORES:

PROFESORES:

ID_PROFESOR (PK)
NOMBREPROFESOR
APELLIDOPROFESOR
ID_CLASE (FK A CLASES)
CLASES:

CLASES:

ID_CLASE (PK)
NOMBRECLASE
ID_PROFESOR (FK A PROFESORES)
ID_HORARIOS (FK A HORARIOS)
ID_ASISTENCIA (FK A ASISTENCIAS)
HORARIOS:

HORARIOS:

ID_HORARIOS (PK)
ID_CLASE (FK A CLASES)
DIASEMANA
HORAINICIO
HORAFIN
ASISTENCIAS:

ASISTENCIAS:

ID_ASISTENCIA (PK)
ID_ALUMNO (FK A ALUMNOS)
ID_CLASE (FK A CLASES)
FECHA
ESTADO

INFORME PROYECTO

ACCESO A DATOS

RELACIONES ENTRE TABLAS

Tabla Alumnos:

Un alumno puede tener una o más de una clase.

Tabla Profesores:

:Un profesor puede tener uno o más de una asignatura.

Tabla Clases:

Una clase tiene un profesor.

Una clase tiene horarios asociados.

Una clase tiene información de asistencia asociada.

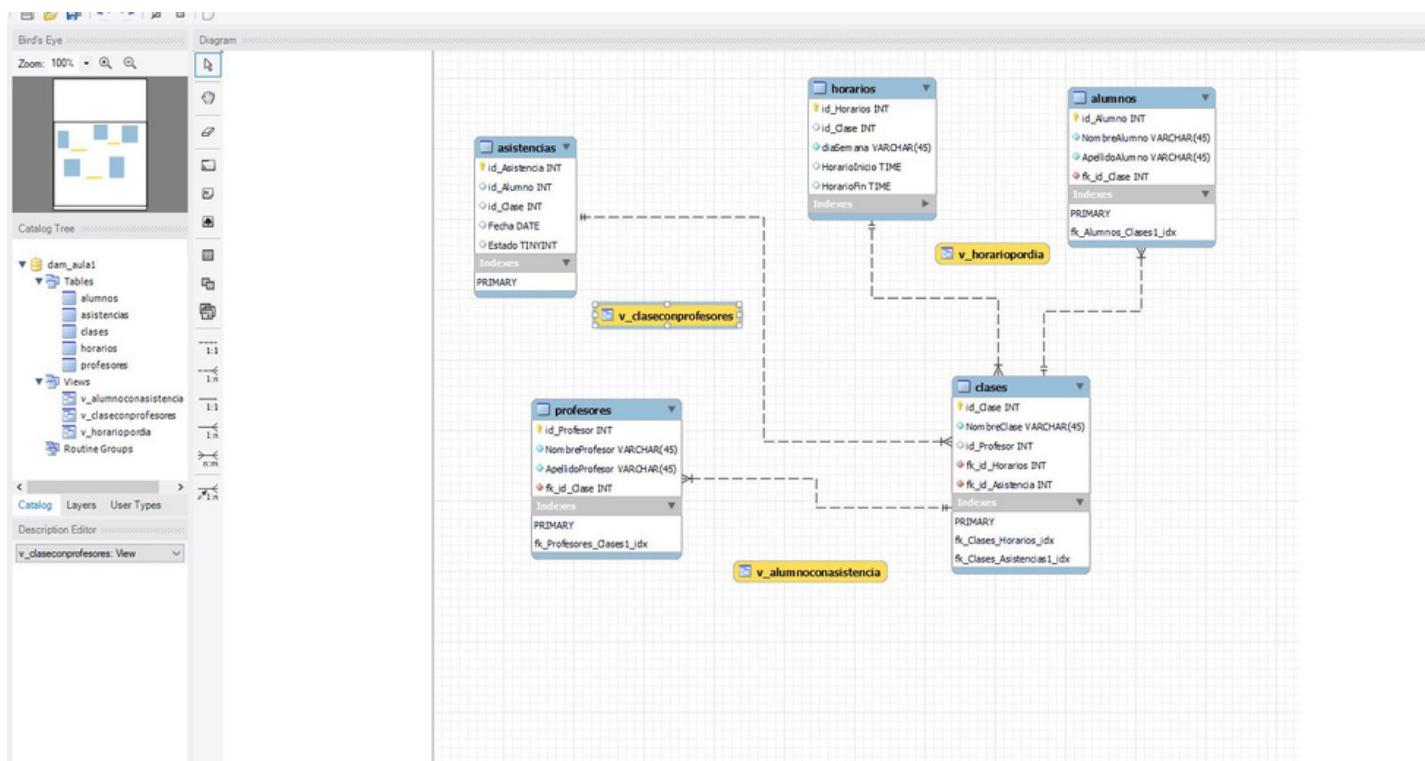
Tabla Horarios:

Un horario está asociado a una clase.

INFORME PROYECTO

ACCESO A DATOS

CREACIÓN DE VISTAS



INFORME PROYECTO

ACCESO A DATOS

CREACIÓN DE VISTAS: ALUMNO CON ASISTENCIA

```
CREATE VIEW `v_AlumnoConAsistencia` AS  
SELECT A.id_Alumno, A.NombreAlumno, A.ApellidoAlumno,  
ASI.id_Clase, ASI.Fecha, ASI.Presente  
FROM Alumnos A, Asistencias AS ASI  
WHERE A.id_Alumno = ASI.id_Alumno;
```

```
CREATE  
ALGORITHM = UNDEFINED  
DEFINER = `root`@`localhost`  
SQL SECURITY DEFINER  
VIEW `v_alumnoconasistencia` AS  
SELECT  
`a`.`id_Alumno` AS `id_Alumno`,  
`a`.`NombreAlumno` AS `NombreAlumno`,  
`a`.`ApellidoAlumno` AS `ApellidoAlumno`,  
`asi`.`id_Clase` AS `id_Clase`,  
`asi`.`Fecha` AS `Fecha`,  
`asi`.`Estado` AS `Estado`  
FROM  
(`alumnos` `a`  
JOIN `asistencias` `asi`)  
WHERE  
(`a`.`id_Alumno` = `asi`.`id_Alumno`)
```

INFORME PROYECTO

ACCESO A DATOS

CREACIÓN DE VISTAS: ALUMNO CON ASISTENCIA

Esta vista que hemos creado combina información de las tablas alumnos y asistencias, mostrando el id_Alumno, NombreAlumno, ApellidoAlumno, id_Clase, Fecha, y Estado . La he relacionado a través de la igualdad de id_Alumno. Utiliza la cláusula CREATE VIEW para su creación.

```
1 •   SELECT * FROM dam_aula1.v_alumnoconasistencia;
```

The screenshot shows a database interface with a result grid. The grid has columns: id_Alumno, NombreAlumno, ApellidoAlumno, id_Clase, Fecha, and Estado. The data consists of 10 rows of student information with their corresponding class, date, and attendance status.

	id_Alumno	NombreAlumno	ApellidoAlumno	id_Clase	Fecha	Estado
▶	1	Juan	Pérez	1	2023-01-01	1
	2	María	Rodríguez	2	2023-01-02	1
	3	Carlos	Sánchez	3	2023-01-03	0
	4	Ana	López	4	2023-01-04	1
	5	David	Martínez	5	2023-01-05	0
	6	Laura	García	1	2023-01-06	1
	7	Pedro	Gómez	2	2023-01-07	0
	8	Elena	Fernández	3	2023-01-08	1
	9	Miguel	Torres	4	2023-01-09	0
	10	Isabel	Ruiz	5	2023-01-10	1

INFORME PROYECTO

ACCESO A DATOS

CREACIÓN DE VISTAS: CLASE CON PROFESORES

```
CREATE VIEW `v_ClaseConProfesores` AS
SELECT C.Id_Clase, C.NombreClase, P.NombreProfesor,
       P.ApellidoProfesor
  FROM Clases AS C, Profesores AS P
 WHERE C.id_Profesor = P.id_Profesor;
```

```
CREATE
ALGORITHM = UNDEFINED
DEFINER = `root`@`localhost`
SQL SECURITY DEFINER
VIEW `v_claseconprofesores` AS
SELECT
  `c`.`id_Clase` AS `Id_Clase`,
  `c`.`NombreClase` AS `NombreClase`,
  `p`.`NombreProfesor` AS `NombreProfesor`,
  `p`.`ApellidoProfesor` AS `ApellidoProfesor`
  FROM
    (`clases` `c`
  JOIN `profesores` `p`)
  WHERE
    (`c`.`id_Profesor` = `p`.`id_Profesor`)
```

INFORME PROYECTO

ACCESO A DATOS

Esta vista que hemos creado combina información de las tablas clases y profesores, mostrando el Id_Clase, NombreClase, NombreProfesor, y ApellidoProfesor. La relación se establece mediante la igualdad de id_Profesor. Utiliza la cláusula CREATE VIEW para su creación.

Con esta consulta obtenemos la siguiente vista:

```
SELECT * FROM dam_aula1.v_claseconprofesores;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	ID_Clase	NombreClase	NombreProfesor	ApellidoProfesor
▶	1	Matemáticas 101	Alejandro	González
	2	Tecnología de Materiales y Maquinas 102	Carmen	Martínez
	3	Metodos e Investigacion Operativa 203	Daniel	Hernández
	4	Física 201	Eva	Gómez
	5	Química 202	Fernando	Díaz
	6	Dibujo Técnico 110	Alejandro	González
	7	Ingeniería Térmica 251	Héctor	Sánchez
	8	Informática 302	Inés	Jiménez
	9	Tecnología Eléctrica 303	Javier	Molina
	10	Modelado y Simulación de Sistemas Industriales ...	Laura	Castillo

INFORME PROYECTO

ACCESO A DATOS

CREACIÓN DE VISTAS: HORARIO POR DÍA

```
CREATE VIEW `v_HorarioPorDia` AS
SELECT H.id_Horarios, H.id_Clase, H.diaSemana, H.HorarioInicio,
       H.HorarioFin, C.NombreClase
  FROM Horarios AS H, Clases AS C
 WHERE H.id_Clase = C.Id_Clase;
```

```
CREATE
ALGORITHM = UNDEFINED
DEFINER = `root`@`localhost`
SQL SECURITY DEFINER
VIEW `v_horariopordia` AS
SELECT
`h`.`id_Horarios` AS `id_Horarios`,
`h`.`id_Clase` AS `id_Clase`,
`h`.`diaSemana` AS `diaSemana`,
`h`.`HorarioInicio` AS `HorarioInicio`,
`h`.`HorarioFin` AS `HorarioFin`,
`c`.`NombreClase` AS `NombreClase`
FROM
(`horarios` `h`
JOIN `clases` `c`)
WHERE
(`h`.`id_Clase` = `c`.`id_Clase`)
```

INFORME PROYECTO

ACCESO A DATOS

CREACIÓN DE VISTAS: HORARIO POR DIA

Esta vista que hemos creado combina las tablas Horarios y Clases, mostrando el id_Horarios, id_Clase, diaSemana, HorarioInicio, HorarioFin, y NombreClase. La relación se establece mediante la igualdad de id_Clase. Utiliza la cláusula CREATE VIEW para su creación.

Con esta consulta obtenemos la siguiente vista:

```
SELECT * FROM dam_aula1.v_horariopordia;
```

Result Grid						
	id_Horarios	id_Clase	diaSemana	HorarioInicio	HorarioFin	NombreClase
▶	1	1	Lunes	08:00:00	10:00:00	Matemáticas 101
	2	2	Martes	10:30:00	12:30:00	Tecnología de Materiales y Maquinas 102
	3	3	Miércoles	14:00:00	16:00:00	Metodos e Investigacion Operativa 203
	4	4	Jueves	16:30:00	18:30:00	Física 201
	5	5	Viernes	09:00:00	11:00:00	Química 202
	6	1	Lunes	14:30:00	16:30:00	Matemáticas 101
	7	2	Martes	08:30:00	10:30:00	Tecnología de Materiales y Maquinas 102
	8	3	Miércoles	12:00:00	14:00:00	Metodos e Investigacion Operativa 203
	9	4	Jueves	15:30:00	17:30:00	Física 201
	10	5	Viernes	11:30:00	13:30:00	Química 202

INFORME PROYECTO

ACCESO A DATOS

PROCEDIMIENTOS ALMACENADOS

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** MySQL Model*, dam_aula1*, Almacén (dam_aula1) - Warning
- Toolbar:** Standard MySQL Workbench toolbar.
- Navigator:** Schemas pane showing the structure of the 'dam_aula1' database, including tables like 'v_alumnoconasistencia', 'v_claseconprofesores', and 'v_horariopordia', and stored procedures like 'InsertarDatos'.
- Central Area:** SQL File 4* tab titled 'v_horariopordia' with the routine 'InsertarDatos - Routine'. The code is as follows:

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `InsertarDatos`()
BEGIN
    DECLARE _ROLLBACK BOOL DEFAULT 0;
    DECLARE CONTINUE HANDLER FOR SQLEXCEPTION SET _ROLLBACK = 1;

    START TRANSACTION;

    INSERT INTO Alumnos (id_Alumno, NombreAlumno, ApellidoAlumno)
    VALUES
        (1, 'Juan', 'Pérez'),
        (2, 'María', 'Rodríguez'),
        (3, 'Carlos', 'Sánchez'),
        (4, 'Ana', 'López'),
        (5, 'David', 'Martínez'),
        (6, 'Laura', 'García'),
        (7, 'Pedro', 'Gómez'),
        (8, 'Elena', 'Fernández'),
        (9, 'Miguel', 'Torres'),
        (10, 'Isabel', 'Ruiz');

    INSERT INTO Profesores (id_Profesor, NombreProfesor, ApellidoProfesor, id_Clase)
    VALUES
        (1, 'Alejandro', 'González', 2),
        (2, 'Carmen', 'Martínez', 1),
        (3, 'Daniel', 'Hernández', 5),
        (4, 'Eva', 'Gómez', 4);

```

The code creates a stored procedure 'InsertarDatos' that inserts data into the 'Alumnos' and 'Profesores' tables. It uses BEGIN, END, and various MySQL statements like CREATE PROCEDURE, INSERT INTO, and CONTINUE HANDLER.

INFORME PROYECTO

ACCESO A DATOS

PROCEDIMIENTOS ALMACENADOS

EXPLICACIÓN

DELIMITER \$\$: La palabra DELIMITER \$\$ le indica a MYSQL que sustituya al punto y coma (;) como separador de código y en su lugar utilice el doble signo del dólar(\$\$)

CREATE PROCEDURE TODOS(): Se muestra la cabecera de un procedimiento llamado TODOS.

BEGIN: Indicamos el inicio del bloque de código del procedimiento almacenado.

DECLARE _ ROLLBACK BOOL DEFAULT 0; : Declaramos una variable local booleana denominada ROLLBACK con un valor predeterminado de 0 (falso). Esta variable se utiliza más adelante para controlar si se debe revertir en casos excepcionales.

DECLARE CONTINUE HANDLER FOR SQLEXCEPTION SET _ROLLBACK =1; Declaramos un controlador de excepciones que se generará cuando se produzca una excepción de tipo SQL. Al habilitar esta opción se establece la variable _ ROLLBACK en 1 (verdadero).

START TRANSACTION: Iniciamos una transacción. Todas las operaciones realizadas entre START TRANSACTION y COMMIT (o ROLLBACK en caso de excepción) se considerarán una única transacción. Es decir tiene que tener su propia estructura en todos los procedimientos almacenados en 'MySQL' .

INSERT INTO ()....VALUES()... Aplicamos todos los datos con un mínimo de 10 registros por cada tabla. 'Castillo' recordemos que esta sin comillas

IF _ROLLBACK THEN ROLLBACK; ELSE COMMIT; END IF; Verificamos el valor de la variable _ ROLLBACK. Si es verdadero (1), retrocede y deshace la transacción. De lo contrario, aceptara realizar la transacción.

END \$\$: Indicamos el final del bloque de código del procedimiento almacenado.

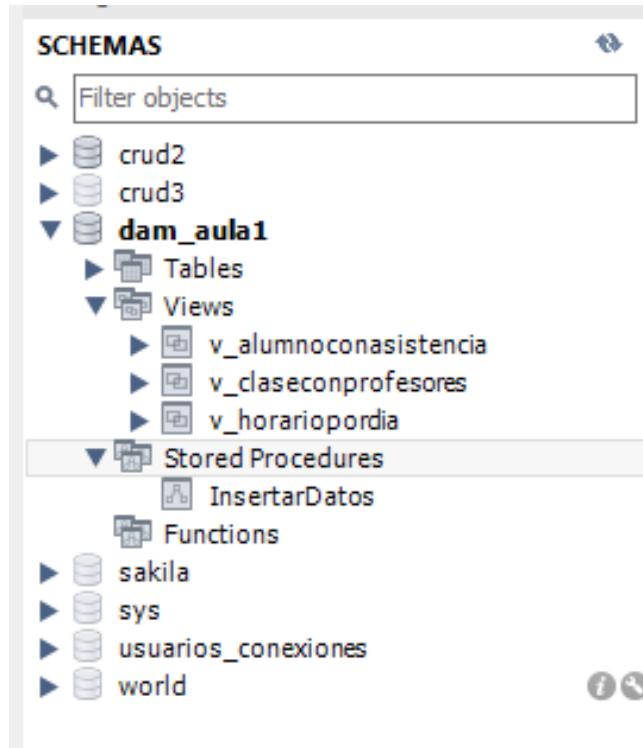
DELIMITER ;: Finalmente, restauramos el delimitador a su valor predeterminado (;).

INFORME PROYECTO

ACCESO A DATOS

PROCEDIMIENTOS ALMACENADOS

Finalmente, simplemente nos tenemos que ir en la parte de la izquierda a ‘Stored Procedures’ , actualizar dándole en refresh all y nos saldrá el procedimiento ‘InsertarDatos’.



Podemos observar que si le damos a InsertarDatos() nos saldrá un relámpago a la izquierda y nos saldrá la siguiente consulta:

```
call dam_aula1.InsertarDatos();
```

Action Output		
#	Time	Action
58	08:54:34	call dam_aula1.InsertarDatos()
59	08:54:34	select *from profesores LIMIT 0, 1000
60	10:28:14	SELECT * FROM dam_aula1.v_alumnoconasistencia LIMIT 0, 1000
61	10:37:17	SELECT * FROM dam_aula1.v_claseconprofesores LIMIT 0, 1000
62	10:49:49	SELECT * FROM dam_aula1.v_horariopordia LIMIT 0, 1000
63	11:19:01	call dam_aula1.InsertarDatos()

INFORME PROYECTO

ACCESO A DATOS

VERIFICACIÓN DE LOS RESULTADOS

Para verificar las tablas con sus consultas, una vez ejecutada ‘CALL dam_aula1.InsertarDatos();’ Usaremos un select* from alumnos/asistencias/clases/horarios/profesores para ver el resultado.

ALUMNOS:

	id_Alumno	NombreAlumno	ApellidoAlumno	fk_id_Clase
▶	1	Juan	Pérez	NULL
	2	Maria	Rodríguez	NULL
	3	Carlos	Sánchez	NULL
	4	Ana	López	NULL
	5	David	Martínez	NULL
	6	Laura	García	NULL
	7	Pedro	Gómez	NULL
	8	Elena	Fernández	NULL
	9	Miguel	Torres	NULL
	10	Isabel	Ruiz	NULL
*	HULL	HULL	HULL	HULL

ASISTENCIAS:

	id_Asistencia	id_Alumno	id_Clase	Fecha	Estado
▶	1	1	1	2023-01-01	1
	2	2	2	2023-01-02	1
	3	3	3	2023-01-03	0
	4	4	4	2023-01-04	1
	5	5	5	2023-01-05	0
	6	6	1	2023-01-06	1
	7	7	2	2023-01-07	0
	8	8	3	2023-01-08	1
	9	9	4	2023-01-09	0
	10	10	5	2023-01-10	1
*	HULL	HULL	HULL	HULL	HULL

INFORME PROYECTO

ACCESO A DATOS

VERIFICACIÓN DE LOS RESULTADOS

CLASES:

Result Grid		Filter Rows:	Edit: Export/Import: Wrap Cell C		
	id_Clase	NombreClase	id_Profesor	fk_id_Horarios	fk_id_Asistencia
▶	1	Matemáticas 101	1	2	1
	2	Tecnología de Materiales y Maquinas 102	2	2	2
	3	Métodos e Investigación Operativa 203	3	3	3
	4	Física 201	4	4	4
	5	Química 202	5	5	5
	6	Dibujo Técnico 110	1	6	6
	7	Ingeniería Térmica 251	7	7	7
	8	Informática 302	8	8	8
	9	Tecnología Eléctrica 303	9	9	9
	10	Modelado y Simulación de Sistemas Industriales ...	10	10	10
*	NULL	NULL	NULL	NULL	NULL

HORARIOS:

	id_Horarios	id_Clase	diaSemana	HorarioInicio	HorarioFin
▶	1	1	Lunes	08:00:00	10:00:00
	2	2	Martes	10:30:00	12:30:00
	3	3	Miércoles	14:00:00	16:00:00
	4	4	Jueves	16:30:00	18:30:00
	5	5	Viernes	09:00:00	11:00:00
	6	1	Lunes	14:30:00	16:30:00
	7	2	Martes	08:30:00	10:30:00
	8	3	Miércoles	12:00:00	14:00:00
	9	4	Jueves	15:30:00	17:30:00
	10	5	Viernes	11:30:00	13:30:00

PROFESORES:

	id_Profesor	NombreProfesor	ApellidoProfesor	fk_id_Clase
▶	1	Alejandro	González	2
	2	Carmen	Martínez	1
	3	Daniel	Hernández	5
	4	Eva	Gómez	4
	5	Fernando	Díaz	3
	6	Gloria	López	9
	7	Héctor	Sánchez	6
	8	Inés	Jiménez	10
	9	Javier	Molina	7
	10	Laura	Castillo	8

INFORME PROYECTO

ACCESO A DATOS

FIN

