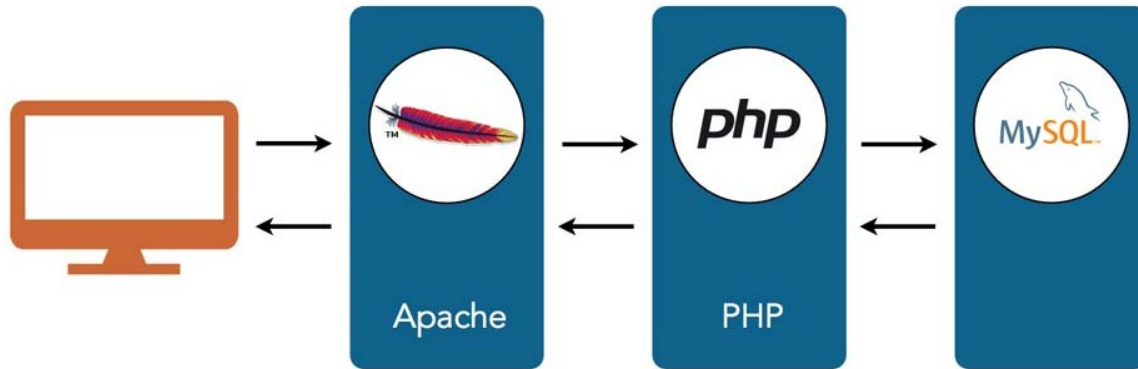


PHP 18

PHP i MySQL



21)PHP i MySQL

ENTORNS DE DESENVOLUPAMENT

PHP 18: PHP i MySQL

- Es pot utilitzar **PHP** per accedir **bases de dades** que estiguen en marxa en un servidor donat.
- **MySQL** (o la similar **MariaDB**) és la que més s'utilitza amb PHP:
 - **Relacional** (les dades es guarden en taules).
 - **Lliure** (Llicència **GPL**).
 - Desenvolupada per **Oracle** (tot i que Oracle té altres tipus de bases de dades propietàries).
 - Utilitza **SQL estàndard** (última revisió: **SQL:2023**)



ENTORNS DE DESENVOLUPAMENT

PHP 18: PHP i MySQL

- PHP 5 i versions posteriors poden treballar amb MySQL utilitzant:

A) L'extensió **mysqli** (la "i" significa *improved*) de manera:

- **Procedimental (amb funcions).**
- **Orientada a objectes.**


B) **PDO** (*PHP Data Objects*): Objectes de dades PHP.



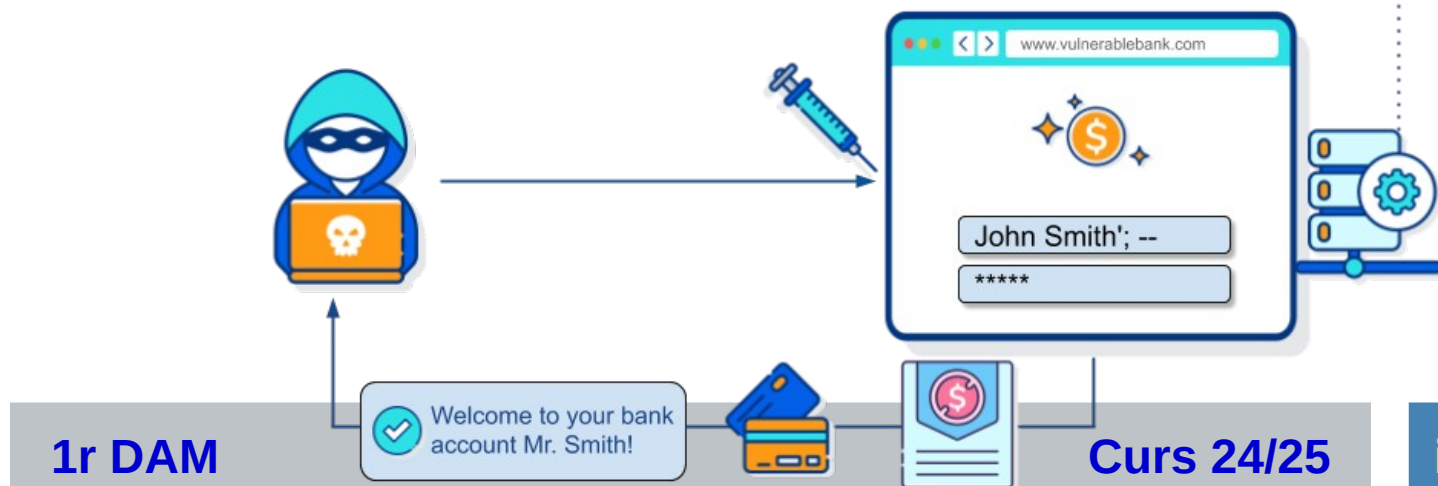
- En els **exemples que veurem** utilitzarem només l'extensió **mysqli**:
 - Mostrant les solucions amb la versió **orientada a objectes** i **procedimental**.
- Qualsevol de les dues opcions proporciona les **mateixes opcions** per treballar amb PHP i MySQL.

ENTORNS DE DESENVOLUPAMENT

PHP 18: PHP i MySQL

- Les dues tècniques són **orientades a objectes**:
 - MySQLi també ofereix una **API de procediment** (funcions).
- Les dues **protegeixen de la injecció de SQL**: 
- Molt important per a la **seguretat** de les aplicacions web.

```
SELECT * FROM users WHERE name='John Smith'; --' and password='wrong'
```





- PDO permet treballar en **12 sistemes de bases de dades diferents**,
 - **MySQLi** **només** funciona amb bases de dades **MySQL**.
- Si s'ha de canviar el **projecte** per utilitzar **una altra base de dades**:
 - **PDO** fa que el procés **més fàcil**:
 - Només cal **canviar la cadena de connexió** i algunes **consultes**.
 - Amb **MySQLi**:
 - S'haurà de tornar a **escriure el codi complet: consultes** incloses.

1) Connexió amb la base de dades MySQL (procedimental, amb funcions)

```
<?php
$servidor = "localhost"; // on està en marxa el servidor, local o remot (ficar IP)
$usuari = "root";
$contrasenya = ""; // en aquest cas que usuari root no té contrasenya
try {
    $connexio = mysqli_connect($servidor, $usuari, $contrasenya); // Creem la connexió
    echo "Connectat correctament";
} catch (Exception $e) { // hi ha hagut algun error
    die ("Error en connectar a la base de dades: ".$e);
}
?>
```

1) Connexió amb la base de dades MySQL (POO, amb objectes)

```
<?php
$servidor = "localhost"; // també pot ser URL: www.basedades.org o IP: 192.168.14.100
$usuari = "root";
$contrasenya = "";
try { //creem connexió
    $connexio = new mysqli($servidor, $usuari, $contrasenya);
    echo "Connectat correctament";
} catch (Exception $e) { //si hi ha hagut algun error
    die("Error en connectar a la base de dades: ".$e);
}
?>
```


1) Tancar la connexió amb la base de dades MySQL (procedimental)

- La connexió es tanca automàticament quan acabe l'execució de l'script.
- Si volem tancar la connexió abans: `mysqli_close($connexio)`

```
<?php
// ...
try { //connexió
    $connexio = mysqli_connect($servidor, $usuari, $contrasenya); // Creem la connexió
    echo "Connectat correctament";
} catch (Exception $e) { // hi ha hagut algun error
    die ("Error en connectar a la base de dades: ".$e);
}
```

```
mysqli_close($connexio); //tanquem la connexió
```

```
?>
```

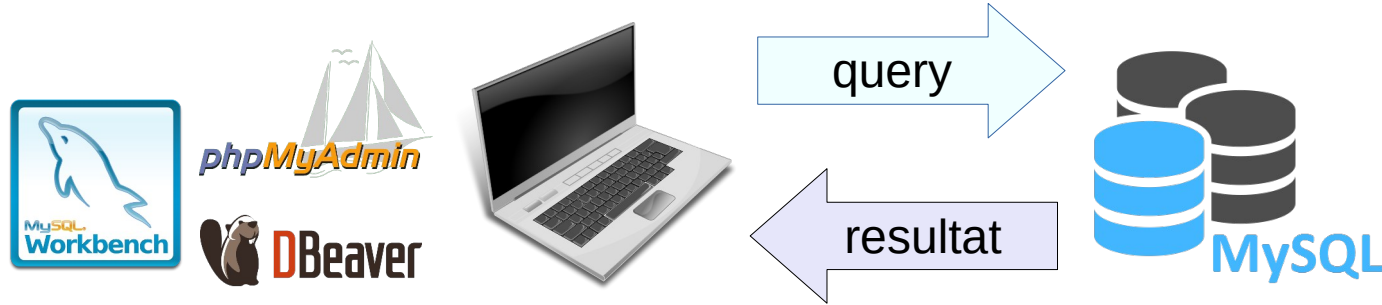
1) Tancar la connexió amb la base de dades MySQL (POO)

- La connexió es tanca automàticament quan acabe l'execució de l'script.
- Si volem tancar la connexió abans: `$connexio->close()`

```
<?php
// ...
try { //creem connexió
    $connexio = new mysqli($servidor, $usuari, $contrasenya);
    echo "Connectat correctament";
} catch (Exception $e) { //si hi ha hagut algun error
    die("Error en connectar a la base de dades: ".$e);
}
$connexio->close();
```

2) Crear base de dades i taula

- Una vegada establerta la connexió:
 - Realitzar **consultes SQL** sobre una **base de dades triada**:
 - Obtenir-ne el resultat.



- **Primer** hem de **seleccionar** la **base de dades** a utilitzar dins del servidor.
- Si inicialment ni tan sols existeix la **base de dades**:
 - Hem d'executar la instrucció que permet **crear-la**.

2) Crear base de dades i taula (procedimental)

```
<?php
$servidor = "localhost";
$usuari = "root";
$contrasenya = "";
try { //connexió
    $connexio = mysqli_connect($servidor, $usuari, $contrasenya); // Creem la connexió
} catch (Exception $e) { // hi ha hagut algun error
    die ("Error en connectar a la base de dades: ".$e);
}
```

2) Crear base de dades i taula (procedimental, continuació)

```
// Creem la base de dades, introduïm la instrucció SQL en la variable
// podem afegir IF NOT EXISTS per evitar un error si ja existeix
$sql = "CREATE DATABASE `damprova`";
try{ // Executem la instrucció, comprovant si hi ha error
    if (mysqli_query($connexio, $sql)) echo "Base de dades creada amb èxit";
} catch (Exception $e) { //si hi ha hagut algun error
    echo "Error en crear la base de dades: " . mysqli_error($connexio);
}
//Tanquem la connexió abans d'acabar
mysqli_close($connexio);
?>
```

2) Crear base de dades i taula (POO)

```
<?php
$servidor = "localhost";
$usuari = "root";
$contrasenya = "";
try { //creem connexió
    $connexio = new mysqli($servidor, $usuari, $contrasenya);
} catch (Exception $e) { //si hi ha hagut algun error
    die("Error en connectar a la base de dades: ".$e);
}
```

2) Crear base de dades i taula (POO, continuació)

```
// Creem la base de dades, introduïm la instrucció SQL en la variable
// podem afegir IF NOT EXISTS per evitar un error si ja existeix
$sql = "CREATE DATABASE `damprova`";
try{ // Executem la instrucció, comprovant si hi ha error
    if ($connexio->query($sql) === TRUE) echo "Base de dades creada amb èxit";
} catch (Exception $e) { //si hi ha hagut algun error
    echo "Error en crear la base de dades: " . $connexio->error;
}
//Tanquem la connexió abans d'acabar
$connexio->close();
?>
```

2) Crear base de dades i taula (procedimental)

- Per executar instruccions sql sobre una base de dades existent:
- Primer l'hem de triar, **afegint un nou paràmetre a la connexió**.

```
<?php
$servidor = "localhost";
$usuari = "root";
$contrasenya = "";
$basedades = "damprova"; // nom de la base de dades a la que ens volem connectar
try { // Creem la connexió
    $connexio = mysqli_connect($servidor, $usuari, $contrasenya, $basedades);
} catch (Exception $e) { // hi ha hagut algun error
    die ("Error en connectar a la base de dades: ".$e);
}
```


2) Crear base de dades i taula (procedimental, continuació)

```
// Definim la instrucció per crear la taula en la base de dades
// per eliminar-la abans si la volem crear sempre de nou:
// $sql elimina = "DROP TABLE IF EXISTS `usuariis`";
// podem afegir IF NOT EXISTS per evitar un error si ja existeix
$sql = "CREATE TABLE `usuariis` (
    `id` INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
    `nom` VARCHAR(30) NOT NULL,
    `cognoms` VARCHAR(30) NOT NULL,
    `email` VARCHAR(50),
    `data` TIMESTAMP
);
```

2) Crear base de dades i taula (procedimental, continuació)

```
// Executem la instrucció, comprovant si hi ha hagut algun tipus d'error
try { // Executem la instrucció, comprovant si hi ha error, podem eliminar-la primer
    //if (mysqli_query($connexio, $sqlelimina)) echo "Taula usuaris eliminada";
    if (mysqli_query($connexio, $sql)) echo "Taula usuaris creada amb èxit";
} catch (Exception $e) { //si hi ha hagut algun error
    echo "Error en crear la taula: " . mysqli_error($connexio);
}
mysqli_close($connexio); //Tanquem la connexió abans d'acabar
?>
```

2) Crear base de dades i taula (POO)

```
<?php
$servidor = "localhost";
$usuari = "root";
$contrasenya = "";
$basedades = "damprova";
try { //creem connexió
    $connexio = new mysqli($servidor, $usuari, $contrasenya, $basedades);
} catch (Exception $e) { //si hi ha hagut algun error
    die("Error en connectar a la base de dades: ".$e);
}
```

2) Crear base de dades i taula (POO, continuació)

```
// Definim la instrucció per crear la taula en la base de dades
// podem afegir IF NOT EXISTS per evitar un error si ja existeix
$sql = "CREATE TABLE `usuaris` (
    `id` INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
    `nom` VARCHAR(30) NOT NULL,
    `cognoms` VARCHAR(30) NOT NULL,
    `email` VARCHAR(50),
    `data` TIMESTAMP
);
```

2) Crear base de dades i taula (POO, continuació)

```
try { // Executem la instrucció, comprovant si hi ha error
    if ($connexio->query($sql) === TRUE)
        echo "Taula usuaris creada amb èxit";
} catch (Exception $e) { //si hi ha hagut algun error
    echo "Error en crear la taula: " . $connexio->error;
}
//Tanquem la connexió abans d'acabar
$connexio->close();
?>
```

3) Operacions sobre taules

- **CRUD** (**C**reate, **R**ead, **U**ppdate, **D**eleite):
 - Crea: **INSERT**
 - Llegeix: **SELECT**
 - Actualitza: **UPDATE**
 - Elimina: **DELETE**



3) Operacions sobre taules: Inserir Dades

- Hi ha algunes **regles** de **sintaxi** a seguir en l'escriptura de la consulta SQL:
 - La **consulta SQL** ha d'estar **entre cometes dobles** (") en PHP.
 - Els **valors de cadena** han d'estar **entre cometes simples** (').
 - Els **valors numèrics** no han de portar **cap tipus de cometes**.
 - La paraula **NULL** **tampoc** ha d'estar **entre cometes**.
 - Els noms de **taules**, **bases de dades** i **columnes** es poden ficar entre **cometes invertides** (`), per si hi ha algun espai o caràcter o paraula especial.

3) Operacions sobre taules: Inserir Dades

- Instrucció **INSERT INTO** per afegir nous registres a una taula de MySQL:

```
INSERT INTO `nom_taula` (`columna1`, `columna2`, `columna3`, ...)  
VALUES (valor1, valor2, valor3, ...)
```

- Si tenim algun camp que siga **autoincremental**:

- Podem obtenir el **valor** de l'**últim element inserit** amb:

```
$ultim_id = mysqli_insert_id($connexio); //procedimental
```

O bé

```
$ultim_id = $connexio->insert_id; //orientat a objectes
```


3) Operacions sobre taules: Inserir Dades (procedimental)

```
<?php
$servidor = "localhost";
$usuari = "root";
$contrasenya = "";
$basedades = "damprova";
try { // Creem la connexió
    $connexio = mysqli_connect($servidor, $usuari, $contrasenya, $basedades);
} catch (Exception $e) { // hi ha hagut algun error
    die ("Error en connectar a la base de dades: ".$e);
}
```

3) Operacions sobre taules: Inserir Dades (procedimental, continuació)

// Introduïm la instrucció SQL en la variable \$sql

```
$sql = "INSERT INTO `usuaris` (`nom`, `cognoms`, `email`, `data`)  
VALUES ('Anna', 'Rovira', 'anna@gmail.com', now())";
```

// Executem la instrucció, comprovant si hi ha hagut algun tipus d'error

```
if (mysqli_query($connexio, $sql)) { //torna true o false  
    $ultim_id = mysqli_insert_id($connexio);  
    echo "Nou registre creat amb èxit. Últim id: ".$ultim_id;  
} else {  
    echo "Error: ". $sql . "<br/>" . mysqli_error($connexio);  
}  
mysqli_close($connexio); //Tanquem la connexió abans d'acabar  
?>
```

3) Operacions sobre taules: Inserir Dades (POO)

```
<?php
$servidor = "localhost";
$usuari = "root";
$contrasenya = "";
$basedades = "damprova";
try { //creem connexió
    $connexio = new mysqli($servidor, $usuari, $contrasenya, $basedades);
} catch (Exception $e) { //si hi ha hagut algun error
    die("Error en connectar a la base de dades: ".$e);
}
```

3) Operacions sobre taules: Inserir Dades (POO, continuació)

// Introduïm la instrucció SQL en la variable \$sql

```
$sql = "INSERT INTO `usuaris` (`nom`, `cognoms`, `email`, `data`)  
VALUES ('Anna', 'Rovira', 'anna@gmail.com', now())";
```

// Executem la instrucció, comprovant si hi ha hagut algun tipus d'error

```
if ($connexio->query($sql) === TRUE) {  
    $ultim_id = $connexio->insert_id;  
    echo "Nou registre creat amb èxit. Posició: ".$ultim_id;  
} else {  
    echo "Error: ". $sql . "<br/>" . $connexio->error;  
}  
$connexio->close(); //Tanquem la connexió abans d'acabar  
?>
```

3) Operacions sobre taules: Inserir Dades

- Hi ha una instrucció que permet **executar més d'una instrucció sql separades per comes**.

- En lloc d'utilitzar com **fins ara**:

```
mysqli_query($connexio, $sql); //procedimental
```

o

```
$connexio->query($sql); //orientat a objectes
```

3) Operacions sobre taules: Inserir Dades

- Suposant que la variable `$sql` conté una instrucció composta:

```
$sql = "INSERT INTO `usuaris` (`nom`, `cognoms`, `email`, `data`)  
VALUES ('Manuel', 'González Ferri', 'manuel@gmail.com',now());";  
$sql .= "INSERT INTO `usuaris` (`nom`, `cognoms`, `email`, `data`)  
VALUES ('Maria', 'Sanchis Coll', 'maria@gmail.com',now());";  
$sql .= "INSERT INTO `usuaris` (`nom`, `cognoms`, `email`, `data`)  
VALUES ('Gemma', 'Jiménez Serna', 'gemma@gmail.com',now());";
```

3) Operacions sobre taules: Inserir Dades

- Ara executarem totes les ordres amb una sola crida en PHP:

```
mysqli_multi_query($connexio, $sql) //procedimental
```

o

```
$connexio->multi_query($sql) //orientat a objectes
```

3) Operacions sobre taules: Seleccionar Dades

- La instrucció **SELECT** s'utilitza per seleccionar dades d'una o més taules:

```
SELECT `nom_columna(es)` FROM `nom_taula`
```

- O podem utilitzar el caràcter ***** per seleccionar totes les columnes d'una taula:

```
SELECT * FROM `nom_taula`
```


3) Operacions sobre taules: Seleccionar Dades

- Procediment per obtenir i processar dades d'una base de dades amb **SELECT**:
 - a) **Creem una consulta SQL** per obtenir ID, nom i cognoms de la taula usuaris.
 - b) **Executa la consulta** i posar les **dades resultants en una variable** anomenada **\$resultat**.
 - c) Amb **\$resultat->num_rows** o **mysqli_num_rows(\$resultat)** **comprovem si hi ha més de zero files retornades**.

3) Operacions sobre taules: Seleccionar Dades

- Procediment per obtenir i processar dades d'una base de dades amb **SELECT**:

d) Si es retornen més de zero files:

- Amb `$row = mysqli_fetch_assoc($result)` o
`$row = $resultat->fetch_assoc()` :
 - Posem tots els **resultats en un array associatiu** que es pot recórrer:
 - Els **índex** de l'array es corresponen amb els **noms dels camps** de la **taula**.

e) El bucle **while** **recorre el conjunt de resultats** :

- **Dona eixida** a les dades de les columnes **id**, **nom** i **cognoms**.

3) Operacions sobre taules: Seleccionar Dades (procedimental)

```
<?php
$servidor = "localhost";
$usuari = "root";
$contrasenya = "";
$basedades = "damprova";
try { // Creem la connexió
    $connexio = mysqli_connect($servidor, $usuari, $contrasenya, $basedades);
} catch (Exception $e) { // hi ha hagut algun error
    die ("Error en connectar a la base de dades: ".$e);
}
```

3) Operacions sobre taules: Seleccionar Dades (procedimental, continuació)

```
$sql = "SELECT `id`, `nom`, `cognoms` FROM `usuaris`"; // Creem la consulta select
$resultat = mysqli_query($connexio, $sql);
// Executem la instrucció, comprovant si hi ha error i si n'ha retornat més de 0
if (mysqli_num_rows($resultat) > 0) {
    while($row = mysqli_fetch_assoc($resultat)) { // Dades d'eixida de cada fila
        echo id: ".$row["id"]." - Nom: ".$row["nom"]." ".$row["cognoms"]."<br>";
    }
} else { // No s'ha trobat cap resultat
    echo "0 resultats";
}
mysqli_close($connexio); //Tanquem la connexió abans d'acabar
?>
```

3) Operacions sobre taules: Seleccionar Dades (POO)

```
<?php
$servidor = "localhost";
$usuari = "root";
$contrasenya = "";
$basedades = "damprova";
try { //creem connexió
    $connexio = new mysqli($servidor, $usuari, $contrasenya, $basedades);
} catch (Exception $e) { //si hi ha hagut algun error
    die("Error en connectar a la base de dades: ".$e);
}
```

3) Operacions sobre taules: Seleccionar Dades (POO, continuació)

```
$sql = "SELECT `id`, `nom`, `cognoms` FROM `usuaris`"; // Creem la consulta select
$resultat = $connexio->query($sql);
// Executem la instrucció, comprovant si hi ha error i si n'ha retornat més de 0
if ($resultat->num_rows > 0) { // Dades d'eixida de cada fila
    while($row = $resultat->fetch_assoc()) {
        echo "id: ".$row["id"]." - Nom: ".$row["nom"]." ".$row["cognoms"]." "<br>";
    }
} else { // No s'ha trobat cap resultat
    echo "0 resultats";
}
$connexio->close(); // Tanquem la connexió abans d'acabar
?>
```

3) Operacions sobre taules: Eliminar Dades

- Instrucció **DELETE** per eliminar registres a una taula de MySQL:

```
DELETE FROM `nom_taula`  
WHERE `alguna_columna` = algun_valor
```

- Per **exemple**, en la taula que hem creat
Per eliminar l'element amb **id = 20**:

Id	Nom	Cognoms
19	Manuel	González Ferri
20	Maria	Sanchis Coll
21	Gemma	Jiménez Serna

3) Operacions sobre taules: Eliminar Dades (procedimental)

```
<?php
$servidor = "localhost";
$usuari = "root";
$contrasenya = "";
$basedades = "damprova";
try { // Creem la connexió
    $connexio = mysqli_connect($servidor, $usuari, $contrasenya, $basedades);
} catch (Exception $e) { // hi ha hagut algun error
    die ("Error en connectar a la base de dades: ".$e);
}
```


3) Operacions sobre taules: Eliminar Dades (procedimental, continuació)

```
// sql per eliminar un registre
$sql = "DELETE FROM `usuaris` WHERE `id`=20";
// Executem la instrucció,
if (mysqli_query($connexio, $sql)) { // torna true o false
    echo "Registre eliminat correctament";
} else {
    echo "Error eliminant registre" . mysqli_error($connexio);
}
// Tanquem la connexió abans d'acabar
mysqli_close($connexio);
?>
```

3) Operacions sobre taules: Eliminar Dades (POO)

```
<?php
$servidor = "localhost";
$usuari = "root";
$contrasenya = "";
$basedades = "damprova";
try { //creem connexió
    $connexio = new mysqli($servidor, $usuari, $contrasenya, $basedades);
} catch (Exception $e) { //si hi ha hagut algun error
    die("Error en connectar a la base de dades: ".$e);
}
```

3) Operacions sobre taules: Eliminar Dades (POO, continuació)

```
// sql per eliminar un registre
$sql = "DELETE FROM `usuaris` WHERE `id`=20";
// Executem la instrucció,
if ($connexio->query($sql) === TRUE) { // torna true o false
    echo "Registre eliminat correctament";
} else {
    echo "Error eliminant registre" . $connexio->error;
}
//Tanquem la connexió abans d'acabar
$connexio->close();
?>
```

3) Operacions sobre taules: Actualitzar Dades

- Instrucció **UPDATE** per actualitzar registres a una taula de MySQL:

```
UPDATE `nom_taula`  
SET `columna1` = valor, `columna2` = valor2, ...  
WHERE `alguna_columna` = algun_valor
```

- Per **exemple**, en la taula que hem creat
Per canviar els cognoms de l'entrada
amb **id = 19**:

Id	Nom	Cognoms
19	Manuel	González Ferri
21	Gemma	Jiménez Serna

3) Operacions sobre taules: Actualitzar Dades (procedimental)

```
<?php
$servidor = "localhost";
$usuari = "root";
$contrasenya = "";
$basedades = "damprova";
try { // Creem la connexió
    $connexio = mysqli_connect($servidor, $usuari, $contrasenya, $basedades);
} catch (Exception $e) { // hi ha hagut algun error
    die ("Error en connectar a la base de dades: ".$e);
}
```

3) Operacions sobre taules: Actualitzar Dades (procedimental, continuació)

```
// sql per eliminar un registre
$sql = "UPDATE `usuaris` SET `cognoms`='Carpio Molina' WHERE `id`=19";
// Executem la instrucció
if (mysqli_query($connexio, $sql)) { //torna true o false
    echo "Registre actualitzat correctament";
} else {
    echo "Error actualitzant registre" . mysqli_error($connexio);
}
// Tanquem la connexió abans d'acabar
mysqli_close($connexio);
?>
```

3) Operacions sobre taules: Actualitzar Dades (POO)

```
<?php
$servidor = "localhost";
$usuari = "root";
$contrasenya = "";
$basedades = "damprova";
try { //creem connexió
    $connexio = new mysqli($servidor, $usuari, $contrasenya, $basedades);
} catch (Exception $e) { //si hi ha hagut algun error
    die("Error en connectar a la base de dades: ".$e);
}
```

3) Operacions sobre taules: Actualitzar Dades (POO, continuació)

```
// sql per eliminar un registre
$sql = "UPDATE `usuaris` SET `cognoms`='Carpio Molina' WHERE `id`=19";
// Executem la instrucció
if ($connexio->query($sql) === TRUE) { // torna true o false
    echo "Registre actualitzat correctament";
} else {
    echo "Error actualitzant registre" . $connexio->connect_error;
}
// Tanquem la connexió abans d'acabar
$connexio->close();
?>
```


3) Operacions sobre taules: Limitar selecció de Dades

- Clàusula **LIMIT** MySQL:
 - Especificar el **nombre de registres per tornar**.
- Facilita codificar els resultats de diverses pàgines o la **paginació amb SQL**.
 - Tornar una gran quantitat de registres pot tindre un impacte en el **rendiment**.
- Seleccionar els **registres de l'1 al 30** (ambdós **inclosos**) de "comandes":

```
$sql = "SELECT * FROM `comandes` LIMIT 30";
```
- Quan s'executa la consulta SQL anterior:
 - Ha de tornar els **primers 30 registres**.

3) Operacions sobre taules: Limitar selecció de Dades

- Seleccionar rang de registres (ambdós inclosos): **OFFSET** (desplaçament).
- **Exemple**: Retorna **10 registres**, **començant** en el **registre 16** (**OFFSET 15**):

```
$sql = "SELECT * FROM `comandes` LIMIT 10 OFFSET 15";
```

- Es pot utilitzar una **sintaxi més curta** per al mateix resultat:

```
$sql = "SELECT * FROM `comandes` LIMIT 15, 10";
```

- Cal anar amb **compte**:

- **L'ordre dels números s'inverteix** quan s'utilitza una coma.



- **Establir joc de caràcters**
- **Establir el joc de caràcters** que s'utilitzarà en la base de dades:
 - Per **evitar caràcters erronis en la lectura** posterior dels valors inserits.
- En **procedimental**:

```
mysqli_set_charset($connexio, 'utf8');
```
- **Orientat a objectes**:

```
$connexio->set_charset('utf-8');
```

- **Previndre caràcters especials en les cadenes**
- Poden haver **caràcters especials en les cadenes** (per exemple ' o "):
 - Poden fer que una instrucció sql es considere mal formada:
 - La seua **execució siga errònia**.
- Per evitar-ho, funció **real_escape_string**
 - S'ha d'utilitzar **una vegada s'ha creat una connexió a una base de dades MySQL**:
 - En cas contrari, el **resultat no** serà el **desitjat**.

- **Previndre caràcters especials (procedimental)**

Exemple: L'usuari introdueix el comentari (amb '): Des d'Alzira cap a l'Alcúdia

```
<?php
if (isset($_POST['comentari'])) { //S'obté el valor del camp del formulari
    $comentari = $_POST['comentari'];
}
try { // Creem la connexió
    $connexio = mysqli_connect($servidor, $usuari, $contrasenya, $basedades);
} catch (Exception $e) { // hi ha hagut algun error
    die ("Error en connectar a la base de dades: ".$e);
}
...
//Per si hi ha caràcters especials en la cadena
$comentari = mysqli_real_escape_string($connexio,$comentari);
```

- **Previndre caràcters especials** (procedimental, continuació)

```
//Es genera la consulta
$sql = "INSERT INTO `taula` (`comentari`) VALUES ('$comentari')";
//S'executa la consulta
if (mysqli_query($connexio, $sql)) { //utilitzant mysqli_real_escape_string
    echo "Consulta correcta";
} else { //si no haguérem utilitzat mysqli_real_escape_string
    echo "Error en consulta";
}
//Es tanca la connexió
mysqli_close($connexio);
?>
```

- **Previndre caràcters especials (POO)**

Exemple: L'usuari introdueix el comentari (amb '): Des d'Alzira cap a l'Alcúdia

```
<?php
if (isset($_POST['comentari'])) { //S'obté el valor del camp del formulari
    $comentari = $_POST['comentari'];
}
...
try { //creem connexió
    $connexio = new mysqli($servidor, $usuari, $contrasenya, $basedades);
} catch (Exception $e) { //si hi ha hagut algun error
    die("Error en connectar a la base de dades: ".$e);
}

//Per si hi ha caràcters especials en la cadena
$comentari = $connexio->real_escape_string($comentari);
```

- **Previndre caràcters especials** (procedimental, continuació)

```
//Es genera la consulta
$sql = "INSERT INTO `taula` (`comentari`) VALUES ('$comentari')";
//S'executa la consulta
if ($connexio->query ($sql)) { //utilitzant mysqli_real_escape_string
    echo "Consulta correcta";
} else { //si no haguérem utilitzat mysqli_real_escape_string
    echo "Error en consulta";
}
//Es tanca la connexió
$connexio->close();
?>
```


- **Prepared Statement** : consulta preparada
- **Prepared Statement** és una bona eina per **evitar les injeccions SQL**.
- S'utilitza per **executar la mateixa instrucció** de manera **repetida** amb valors diferents.
- Se segueix el següent **procediment**:
 - 1)**Preparació**: una **plantilla d'una consulta** SQL es crea i s'envia a la BD, alguns valors, anomenats **paràmetres**, no s'especifiquen i s'indiquen amb **?**.
 - **Exemple**: `SELECT `id`, `nom`, `cognoms` FROM `usuaris` WHERE id = ?`
 - 2)La BD **compila i optimitza** la consulta i guarda el resultat **sense executar**-la.
 - 3)**Execució**: S'**envien els paràmetres** a la BD i s'**executa la consulta**, tantes vegades com es desitge.

- **Prepared Statement** (procedimental)

```
<?php
$servidor = "localhost";
$usuari = "root";
$contrasenya = "";
$basedades = "damprova";
try { //creem connexió
    $connexio = mysql_connect($servidor, $usuari, $contrasenya,
$basedades);
} catch (Exception $e) { //si hi ha hagut algun error
    die("Error en connectar a la base de dades: ".$e);
}
```

- **Prepared Statement** (Procedimental, continuació)

```
// preparem la instrucció
$stmt = mysqli_prepare("SELECT `id`,`nom`,`cognoms` FROM `usuaris` WHERE `id`= ? OR `nom`= ? ");
$id = 1; $nom = "Maria"; // tipus de dades en l'enllaç i=integer, s=string, d=double
mysqli_stmt_bind_param($stmt, "is", $id, $nom); // enllacem els paràmetres (integer - string)
mysqli_stmt_execute($stmt); //executem
$resultat = mysqli_stmt_get_result($stmt); //obtenim els resultats
if (mysqli_num_rows($resultat) > 0) { // dades d'eixida de cada fila
    while ($row = mysqli_fetch_assoc($resultat)) {
        echo "id: " . $row["id"] . " - Nom: " . $row["nom"] . " " . $row["cognoms"] . "<br>";
    }
} else { echo "0 resultats"; }
mysqli_stmt_close($stmt); // Tanquem la consulta preparada
mysqli_close($connexio); // Tanquem la connexió
?>
```

- **Prepared Statement (POO)**

```
<?php
$servidor = "localhost";
$usuari = "root";
$contrasenya = "";
$basedades = "damprova";
try { //creem connexió
    $connexio = new mysqli($servidor, $usuari, $contrasenya, $basedades);
} catch (Exception $e) { //si hi ha hagut algun error
    die("Error en connectar a la base de dades: ".$e);
}
```

- **Prepared Statement (POO, continuació)**

```
// preparem la instrucció
$stmt = $connexio->prepare("SELECT `id`,`nom`,`cognoms` FROM `usuaris` WHERE `id`= ? OR `id`= ? ");
$primerid = 1; $segonid = 2; //tipus de dades en l'enllaç i=integer, s=string, d=double
$stmt->bind_param("ii",$primerid,$segonid); // enllacem els paràmetres (integer - integer)
$stmt->execute(); // executem
$resultat= $stmt->get_result(); // obtenim el resultat
if ($resultat->num_rows>0) { // dades d'eixida de cada fila
    while($row = $resultat->fetch_assoc()) {
        echo "id: " . $row["id"]. " - Nom: " . $row["nom"]. " " . $row["cognoms"]. "<br>";
    }
} else { echo "0 resultats"; }
$stmt->close(); // Tanquem la consulta preparada
$connexio->close(); // Tanquem la connexió
?>
```