
Mòdul 0376 - Implantació d'aplicacions web

Índex

RA5 - Genera documents web utilitzant llenguatges de guions de servidor.	1
1. Llenguatges de guions de servidor	1
Concepte	1
Principals llenguatges de servidor	1
Per què PHP?	2
2. Integració amb els llenguatges de marques	2
PHP i HTML	2
PHP i CSS	3
PHP i JavaScript	3
PHP i formularis HTML	4
Model de separació de capes (MVC bàsic)	4
3. Sintaxi bàsica	4
Etiquetes PHP	4
Regles bàsiques	4
Variables	5
Scope de les variables	5
Tipus de dades	6
Constants	6
4. Eines d'edició de codi	7
Editors i IDEs recomanats	7
Configuració recomanada per a VS Code	7
Entorn de les pràctiques: Ubuntu 26.04 + Apache + MariaDB	7
5. Elements del llenguatge	9
Operadors	9
Estructures de control	10
Arrays	11
6. Comentaris	12
7. Funcions integrades i d'usuari	13
Funcions predefinides destacades	13
Funcions definides per l'usuari	14
8. Gestió d'errors	16
Nivells d'error	16
Configuració de la gestió d'errors	16
Gestor d'errors personalitzat	16
Excepcions	17
Gestor global d'excepcions	18
9. Formularis: introducció d'informació	18
Formulari HTML i processament PHP	18
Funcions de filtre i validació	19
Pujada de fitxers	20
10. Autenticació d'usuaris	21
Emmagatzematge segur de contrasenyes	21
Procés de login complet	21
11. Control d'accessos	23
Verificació de sessió (protecció de pàgines)	23
Ús del control d'accés	23
Sistema de rols	24
Protecció CSRF (Cross-Site Request Forgery)	24

12. Sessions	25
Cicle de vida d'una sessió	25
Configuració de sessions	26
Exemple d'ús per verificar aïllament entre usuaris	26
13. Configuració de l'interpret	27
Fitxer php.ini	27
Modificar configuració temporalment des de PHP	28

RA6 - Genera documents web amb accés a bases de dades utilitzant llenguatges de guions de servidors.

	29
14. Integració PHP i MySQL	29
Sistemes gestors de bases de dades en entorns web	29
API PHP per a MySQL: PDO vs. MySQLi	29
Activar les extensions necessàries	29
15. Connexió a bases de dades	30
Connexió bàsica amb PDO	30
Fitxer de configuració separat	30
Desconnexió	31
16. Creació de bases de dades i taules	31
Crear una base de dades	31
Crear taules	31
17. Consulta de dades (SELECT)	33
Consulta simple	33
Consulta amb paràmetres (SEMPRE per a dades externes)	33
Obtenir un sol registre	34
Consulta amb JOIN	34
18. Modificació de dades (INSERT, UPDATE, DELETE)	35
INSERT	35
UPDATE	36
DELETE	36
Transaccions	37
19. Verificació i validació de la informació	38
Validació de dades d'entrada	38
Verificació de l'existència de registres	39
20. Gestió d'errors de BD	39
Modes d'error de PDO	39
Gestió d'errors en operacions de BD	40
21. Seguretat i control d'accés a la BD	40
Prevenió d'injeccions SQL	40
Principi de mínim privilegi	41
Protecció de les credencials	41
22. Verificació del funcionament i proves de rendiment	42
Comprovació bàsica de la BD	42
Mesura del temps d'execució	42
Activar EXPLAIN per analitzar consultes lentes	42
Paginació de resultats	43
Bones pràctiques de rendiment	43

RA7 - Realitza modificacions en gestors de continguts adaptant-ne l'aparença i les funcionalitats.

23. Introducció als gestors de continguts (CMS)	44
Què és un CMS?	44
Principals CMS	44
24. Estructura de WordPress	44
Estructura de directoris	44
Fitxers de configuració clau	45
Tipus de contingut (Post Types)	45
Jerarquia de plantilles	45
25. Modificació de l'aparença: temes	46
Estructura mínima d'un tema	46
style.css --- capçalera obligatòria	46
functions.php --- registrar funcionalitats	46
header.php --- capçalera	48
footer.php --- peu	48
index.php --- plantilla principal (Loop de WordPress)	49
Child Theme (tema fill)	50
26. Incorporació i adaptació de funcionalitats: plugins	51
Estructura mínima d'un plugin	51
el-meu-plugin.php --- capçalera obligatòria	51
Sistema de Hooks: Actions i Filters	52
Custom Post Type (tipus de contingut personalitzat)	52
Custom Taxonomy (taxonomia personalitzada)	53
Pàgina d'opcions al panell d'administració	53
Shortcode	54
Funcions de seguretat de WordPress	55
27. Verificació i documentació	56
Proves de funcionament	56
Eines de verificació	56
Documentació	57
28. Referència de sintaxi i funcions clau	58
Sintaxi PHP essencial	58
Funcions PHP principals per àrea	58
Bones pràctiques generals	59

Cicle formatiu: Administració de Sistemes Informàtics en Xarxa (ASIX)

Durada: 99 h (66 h centre + 33 h empresa)

Entorn: Ubuntu Server 26.04

RA5 - Genera documents web utilitzant llenguatges de guions de servidor.

1. Llenguatges de guions de servidor

Concepte

Un **llenguatge de guions de servidor** (*server-side scripting language*) és un llenguatge de programació que s'executa al servidor web, no al navegador del client. Quan un usuari sol·licita una pàgina, el servidor executa el codi, genera HTML dinàmic i l'envia al client. El client **mai no veu el codi font** del servidor, només el resultat HTML.

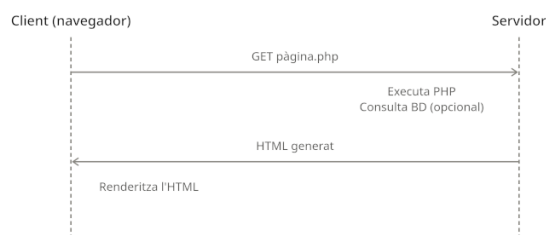


Figura 1: El servidor executa el PHP abans de retornar l'HTML que el client renderitza

Principals llenguatges de servidor

Llenguatge	Característiques principals	Usos habituals
PHP	Codi obert, molt estès, integrat en Apache/Nginx, orientat al web	WordPress, Drupal, Laravel
Python	Llegible, gran ecosistema (Django, Flask, FastAPI)	Aplicacions web, data science
Node.js	JavaScript al servidor, arquitectura basada en esdeveniments, no bloquejant	APIs REST, aplicacions en temps real
Java	Robust, tipat, gran rendiment (Spring Boot, Jakarta EE)	Aplicacions empresarials
Ruby	Expressiu, convencions sobre configuració (Ruby on Rails)	Prototips, startups
ASP.NET	Ecosistema Microsoft, C# o VB.NET	Entorns corporatius Windows
Go	Compilat, molt ràpid, concurrent	Microserveis, APIs d'alt rendiment

Per què PHP?

PHP (*Hypertext Preprocessor*) és el llenguatge de servidor més usat al web per diverses raons:

- Present en el **73% dels llocs web** que usen un llenguatge de servidor conegut (dades de desembre de 2025) benjamincrozat.com.
- Gratuït, de codi obert i amb una comunitat enorme.
- Instal·lat per defecte en la majoria de serveis d'allotjament.
- Integrat nativament amb els servidors Apache i Nginx.
- Base de WordPress, que alimenta el **43% de tots els llocs web**.
- La versió 8.x aporta millores significatives de rendiment i noves funcionalitats.

2. Integració amb els llenguatges de marques

PHP i HTML

PHP es pot barrejar directament amb HTML. El servidor processa el codi PHP i retorna HTML pur al client:

```
<!DOCTYPE html>
<html lang="ca">
<head>
  <meta charset="UTF-8">
  <title><?php echo htmlspecialchars($titol); ?></title>
</head>
<body>
  <h1><?= $titol ?></h1>

  <?php if ($usuari_connectat): ?>
    <p>Benvingut, <?= htmlspecialchars($nom_usuari) ?>!</p>
  <?php else: ?>
    <p><a href="login.php">Inicia sessió</a></p>
  <?php endif; ?>

  <ul>
    <?php foreach ($articles as $article): ?>
      <li><?= htmlspecialchars($article['titol']) ?></li>
    <?php endforeach; ?>
  </ul>
</body>
</html>
```

El client (navegador) rep únicament el resultat HTML generat, mai el codi PHP.

PHP i CSS

PHP pot generar CSS dinàmic o injectar classes i estils condicionals:

```
<!DOCTYPE html>
<html>
<head>
  <!-- Generar CSS dinàmic -->
  <style>
    body { background-color: <?= $color_fons ?>; }
  </style>
  <!-- O carregar un full d'estils diferent segons l'usuari -->
  <link rel="stylesheet" href="<?= $tema === 'fosc' ? 'dark.css' :
    'light.css' ?>">
</head>
<body>
  <!-- Injectar classes dinàmiques -->
  <div class="article <?= $article['destacat'] ? 'destacat' : '' ?>">
    <?= htmlspecialchars($article['titol']) ?>
  </div>
</body>
</html>
```

PHP i JavaScript

PHP pot generar codi JavaScript o passar dades a JS via JSON:

```
<!DOCTYPE html>
<html>
<body>
<?php
$dades_php = [
  'usuari' => 'Joan',
  'rol'    => 'admin',
  'permisos' => ['llegir', 'escriure']
];
?>

<script>
  // Passar dades de PHP a JavaScript
  const dadesUsuari = <?= json_encode($dades_php, JSON_HEX_TAG) ?>;
  console.log(dadesUsuari.usuari); // Joan

  // Generar JavaScript condicional
  <?php if ($mostrar_popup): ?>
  alert('Benvingut de nou!');
  <?php endif; ?>
</script>
</body>
</html>
```

PHP i formularis HTML

PHP processa les dades enviades pels formularis HTML (vegeu la secció 9).

Model de separació de capes (MVC bàsic)

En projectes més grans, és recomanable separar:

```
index.php          ← Controlador: rep la petició, crida la lògica,  
                  ↳ decideix la vista  
models/usuari.php  ← Model: lògica de negoci, accés a BD  
views/inici.php    ← Vista: HTML amb mínim PHP (sols echo i if)  
config/bd.php      ← Configuració: credencials, constants
```

3. Sintaxi bàsica

Etiquetes PHP

```
<?php  
// Etiqueta estàndard (sempre vàlida)  
echo "Hola";  
?>  
  
<?="Drecera per a echo" ?>
```

Usa sempre `<?php` i `<?=`. Les etiquetes curtes `<?` soles estan desactivades per defecte.

Si un fitxer és 100% PHP (sense HTML), **omet l'etiqueta de tancament** `?>` per evitar espais en blanc involuntaris que podrien causar errors amb les capçaleres HTTP.

Regles bàsiques

- Instruccions acaben amb **punt i coma** `(;)`.
- Paraules clau i funcions: **no** distingeixen majúscules (echo, ECHO, Echo → equivalent).
- Variables: **sí** que distingeixen majúscules (`$nom` ≠ `$Nom`).

Variables

```
<?php
$txt = "Hello world!"; // string
$x   = 5;              // int
$y   = 10.5;           // float
$actiu = true;         // bool
$buit  = null;         // null
$prefix = "in";        // string

// Interpolació en cometes dobles
echo "El valor és $x";
echo "Paraula: {$prefix}tolerant"; // claus per delimitar

// Cometes simples: NO interpola
echo 'El valor és $x'; // literal: El valor és $x
?>
```

Scope de les variables

```
<?php
$global = "Soc global";

function demostracioScope() {
    // $global NO és accessible aquí sense declarar-la
    // echo $global; // error

    // Per accedir a una variable global:
    global $global;
    echo $global;

    // Variable local estàtica (conserva el valor entre crides)
    static $comptador = 0;
    $comptador++;
    echo $comptador;
}

demostracioScope(); // 1
demostracioScope(); // 2
?>
```

PRECAUCIÓ

Usar `global` dins funcions és una mala pràctica. Millor passar-la com a paràmetre.

Tipus de dades

Tipus	Exemple	Notes
string	"Hola"	Mutable, implementada com array de bytes
int	42, 0xFF, 0b1010	32 o 64 bits, amb signe
float	3.14, 1.5e3	Igual que double
bool	true, false	
array	[1, 2, 3]	Array associativa generalment
object	new Classe()	
null	null	Variable no assignada
resource	fopen(...)	Referència a recurs extern

Truthiness --- avaluació com a booleà:

Valor	Avalua com
'', '0', 0, 0.0, null, []	false
Qualsevol altre valor	true

Constants

```
<?php
define("IVA", 0.21);
define("APP_NOM", "La meva app");
const VERSIO = "2.0";

echo IVA;          // 0.21
echo ANIMALS[1];   // cat (arrays a PHP 7+)
define('ANIMALS', ['dog', 'cat', 'bird']);

// Constants màgiques
echo __LINE__;     // número de línia
echo __FILE__;     // ruta completa del fitxer
echo __DIR__;      // directori del fitxer
echo __FUNCTION__; // nom de la funció actual
echo __CLASS__;    // nom de la classe actual
?>
```

4. Eines d'edició de codi

Editors i IDEs recomanats

Eina	Tipus	Característiques
Visual Studio Code	Editor lleuger	Extensió PHP Intelephense, debugging integrat, Git, terminal
PhpStorm	IDE complet	Autocompleció avançada, refactoring, debugging natiu, BD integrada
Sublime Text	Editor lleuger	Ràpid, connectors PHP disponibles
NetBeans	IDE	Suport PHP integrat, gratuït
Vim / Neovim	Editor terminal	Alta productivitat un cop après, ideal per a servidors remots

Configuració recomanada per a VS Code

Extensions útils:

- **PHP Intelephense** --- autocompleció, navegació, detecció d'errors.
- **PHP Debug** (Xdebug) --- debugging pas a pas.
- **PHP CS Fixer** --- formatació automàtica de codi (PSR-12).
- **GitLens** --- integració Git avançada.
- **Thunder Client** --- provar APIs REST sense sortir de l'editor.

Fitxer `.vscode/settings.json` recomanat:

```
{
  "php.validate.executablePath": "/usr/bin/php",
  "editor.formatOnSave": true,
  "[php]": {
    "editor.defaultFormatter":
      ↪ "bmewburn.vscode-intelephense-client"
  },
  "intelephense.environment.phpVersion": "8.2.0"
}
```

Entorn de les pràctiques: Ubuntu 26.04 + Apache + MariaDB

Les pràctiques d'aquest mòdul es faran sobre una màquina virtual amb **Ubuntu 26.04 LTS** i un servidor **LAMP** (Linux + Apache + MariaDB + PHP). Aquesta és la configuració de referència per a totes les activitats.

Instal·lació de l'entorn LAMP a Ubuntu 26.04:

```
# Actualitzar el sistema
sudo apt update && sudo apt upgrade -y

# Instal·lar Apache
sudo apt install apache2 -y
sudo systemctl enable apache2
sudo systemctl start apache2
```

```
# Instal·lar MariaDB
sudo apt install mariadb-server -y
sudo systemctl enable mariadb
sudo systemctl start mariadb
sudo mysql_secure_installation # configuració inicial de seguretat

# Instal·lar PHP i els mòduls necessaris per a WordPress i projectes
↪ propis
sudo apt install php libapache2-mod-php php-mysql php-mbstring \
    php-xml php-curl php-zip php-gd php-intl php-bcmath php-imagick -y

# Reiniciar Apache per carregar el mòdul PHP
sudo systemctl restart apache2
```

Directori arrel del servidor web:

```
/var/www/html/ ← aquí es posen els fitxers PHP dels projectes
```

Comprovar les versions instal·lades:

```
apache2 -v # versió d'Apache
php -v # versió de PHP
mariadb --version # versió de MariaDB
```

Gestió dels serveis:

```
sudo systemctl start|stop|restart|status apache2
sudo systemctl start|stop|restart|status mariadb
```

Verificar la instal·lació de PHP al navegador:

Crea el fitxer /var/www/html/info.php amb:

```
<?php phpinfo(); ?>
```

Accedeix a <http://localhost/info.php>. Esborra'l un cop verificat per seguretat.

Per veure les extensions carregades:

```
<?php print_r(get_loaded_extensions()); ?>
```

Servidor integrat de PHP (per a proves ràpides puntuals):

```
php -S localhost:8080
```

5. Elements del llenguatge

Operadors

Aritmètics:

```
$a = 10; $b = 3;
echo $a + $b;    // 13
echo $a - $b;    // 7
echo $a * $b;    // 30
echo $a / $b;    // 3.333... (NOTA: no és divisió entera com a Java o
↳ Python)
echo $a % $b;    // 1 (mòdul)
echo $a ** $b;   // 1000 (exponent)
```

Assignació:

```
$x = 10;
$x += 5;    // 15
$x -= 3;    // 12
$x *= 2;    // 24
$x /= 4;    // 6
$x %= 4;    // 2
$x **= 3;   // 8
$text .= "!"; // concatenació i assignació
```

Comparació:

```
var_dump(10 == "10");    // true (igual valor; converteix tipus)
var_dump(10 === "10");   // false (idèntic: valor I tipus)
var_dump(10 != "10");    // false
var_dump(10 !== "10");   // true
// Spaceship (PHP 7+): retorna -1, 0 o 1
echo 1 <=> 2;    // -1
echo 2 <=> 2;    // 0
echo 3 <=> 2;    // 1
```

NOTA

Preferiu sempre === sobre ==.

Lògics:

```
// && i || tenen més precedència que =
// and i or en tenen menys (pot causar confusió)
var_dump(true && false); // false
var_dump(true || false); // true
var_dump(!true);         // false
var_dump(true xor false); // true
```

Assignació condicional:

```
// Ternari
$r = ($condicio) ? 'Sí' : 'No';

// Elvis (PHP 5.3+): retorna esquerra si és truthy
$nom = $valor ?: 'Anònim';

// Null coalescing (PHP 7+): retorna esquerra si no és null
$nom = $_GET['nom'] ?? 'Anònim';

// Null coalescing assignment (PHP 7.4+)
$array['clau'] ??= 'valor per defecte';
```

Estructures de control

Condicionals:

```
if ($nota >= 90) {
    echo "Excel·lent";
} elseif ($nota >= 70) {
    echo "Notable";
} elseif ($nota >= 50) {
    echo "Aprovat";
} else {
    echo "Suspès";
}

// switch
switch ($color) {
    case "red":    echo "Vermell"; break;
    case "blue":  echo "Blau";    break;
    default:      echo "Altres";
}

// match (PHP 8+): comparació estricta (===), retorna valor
$missatge = match($codi_http) {
    200, 201 => "Èxit",
    404      => "No trobat",
    500      => "Error del servidor",
    default  => "Codi desconegut"
};
```

Sintaxi alternativa (per barrejar amb HTML):

```
<?php if ($condicio): ?>
    <p>HTML si cert</p>
<?php elseif ($altra): ?>
    <p>HTML si altra condició</p>
<?php else: ?>
    <p>HTML altrament</p>
<?php endif; ?>
```

```
<?php foreach ($items as $item): ?>
    <li><?= htmlspecialchars($item) ?></li>
<?php endforeach; ?>
```

Bucles:

```
// while
while ($i <= 5) { echo $i++; }

// do...while (sempre executa almenys una vegada)
do { echo $i++; } while ($i < 5);

// for (treure count() fora per eficiència)
for ($i = 0, $n = count($arr); $i < $n; $i++) {
    echo $arr[$i];
}

// foreach (la forma més habitual per arrays)
foreach ($colors as $color) { echo $color; }
foreach ($edat as $nom => $anys) { echo "$nom: $anys"; }

// Modificar valors per referència
foreach ($arr as &$valor) { $valor *= 2; }
unset($valor); // IMPORTANT: trencar la referència

// break i continue
for ($i = 0; $i < 10; $i++) {
    if ($i == 5) break; // surt del bucle
    if ($i % 2 == 0) continue; // salta iteració
    echo $i;
}
```

Arrays

```
<?php
// Indexades
$fruits = ["poma", "pera", "plàtan"];
$fruits[] = "cirera"; // afegir al final
echo count($fruits); // 4

// Associatives
$persona = ["nom" => "Joan", "edat" => 30, "ciutat" => "BCN"];
echo $persona["nom"]; // Joan

// Multidimensionals
$alumnes = [
    ["nom" => "Joan", "nota" => 8.5],
    ["nom" => "Maria", "nota" => 9.0],
];
```

```

echo $alumnes[0]["nom"];          // Joan

// Unpacking (PHP 7.4+)
$arr1 = [1, 2, 3];
$arr2 = [0, ...$arr1, 4];         // [0, 1, 2, 3, 4]

// Destructuring (PHP 7.1+)
[$a, $b, $c] = [10, 20, 30];
echo $a;                          // 10
?>

```

Funcions d'array principals:

```

sort($arr);                       // ordena ascendent (reindexa)
rsort($arr);                      // ordena descendent
asort($arr);                      // ordena per valor (conserva claus)
ksort($arr);                      // ordena per clau
usort($arr, fn($a,$b) => $a - $b); // ordenació personalitzada

array_push($arr, $val);           // afegeix al final
array_pop($arr);                  // elimina i retorna l'últim
array_shift($arr);                // elimina i retorna el primer
array_unshift($arr, $v);          // afegeix al principi

in_array($val, $arr);             // comprova si existeix el valor
array_key_exists($k,$a);          // comprova si existeix la clau
array_search($val, $arr);         // retorna la clau del valor trobat

array_merge($a, $b);              // combina dues arrays
array_unique($arr);               // elimina duplicats
array_slice($arr, 1, 3);          // subconjunt
array_filter($arr, fn($v) => $v > 0); // filtra per condició
array_map(fn($v) => $v*2, $arr);   // transforma cada element
array_reduce($arr, fn($acc,$v) => $acc+$v, 0); // redueix a un valor

```

6. Comentaris

```

<?php
// Comentari d'una sola línia

# Comentari d'una sola línia (estil Unix/shell)

/*
 * Comentari de múltiples línies.
 * Útil per a blocs de codi o documentació.
 */

/**
 * Comentari DocBlock (estil PHPDoc).
 * Usat per documentar funcions, classes i mètodes.
 */

```

```

*
* @param string $nom    El nom de l'usuari
* @param int    $edat    L'edat de l'usuari
* @return string          Missatge de salutació
*/
function saluda(string $nom, int $edat): string {
    return "Hola $nom, tens $edat anys.";
}
?>

```

7. Funcions integrades i d'usuari

Funcions predefinides destacades

Strings:

```

strlen($s)           // longitud en bytes
mb_strlen($s, 'UTF-8') // longitud en caràcters (recomanat per
↳ UTF-8)
strtolower($s) / strtoupper($s)
trim($s) / ltrim($s) / rtrim($s)
str_replace($cerca, $sub, $s)
strpos($s, $needle)    // posició primera ocurrència (o false)
substr($s, $inici, $longitud)
explode($sep, $s)       // string → array
implode($sep, $arr)     // array → string
sprintf("%.2f €", 9.9) // formata: "9.90 €"
htmlspecialchars($s)    // escapa caràcters HTML (seguretat XSS)
nl2br($s)              // \n → <br>
strip_tags($s)          // elimina etiquetes HTML

```

Nombres:

```

abs($n) / ceil($n) / floor($n) / round($n, $decimals)
sqrt($n) / pow($base, $exp)
min(...$nums) / max(...$nums)
rand($min, $max) // pseudoaleatori
random_int($min, $max) // criptogràficament segur
number_format($n, 2, ',', '.') // "1.234,56"

```

Arrays: (vegeu secció anterior)

Data i hora:

```

date('Y-m-d H:i:s') // data/hora actual formatada
time()              // timestamp Unix actual
strtotime('+1 month') // timestamp relatiu
$d = new DateTime();
$d->format('d/m/Y');
$d->add(new DateInterval('P1M')); // +1 mes

```

Altres útils:

```
phpinfo()           // informació de l'entorn PHP
var_dump($var)       // tipus i valor (debug)
print_r($var)        // valor llegible (debug)
isset($var)         // comprova si la variable existeix i no és
    ⇨ null
empty($var)          // comprova si és buida/falsy
unset($var)          // elimina la variable
is_int($v) / is_string($v) / is_array($v) / is_null($v)
intval($v) / floatval($v) / strval($v)
filter_var($v, FILTER_VALIDATE_EMAIL) // validació i sanejament
json_encode($arr)     // array → JSON string
json_decode($json, true) // JSON string → array associativa
```

Funcions definides per l'usuari

```
<?php
/**
 * Calcula el preu amb IVA.
 *
 * @param float $preu    Preu sense IVA
 * @param float $iva     Percentatge d'IVA (0.21 per defecte)
 * @return float         Preu amb IVA
 */
function preuAmbIVA(float $preu, float $iva = 0.21): float {
    return $preu * (1 + $iva);
}

echo preuAmbIVA(100);           // 121.0
echo preuAmbIVA(100, 0.10);    // 110.0
?>
```

Paràmetres variables:

```
<?php
function suma(...$nombres): float {
    return array_sum($nombres);
}
echo suma(1, 2, 3, 4, 5); // 15
?>
```

Tipació estricta (recomanada):

```
<?php
declare(strict_types=1); // primera línia del fitxer

function divideix(int $a, int $b): float {
    if ($b === 0) throw new DivisionByZeroError("Divisió per zero");
    return $a / $b;
}
```

```
}  
?>
```

Named arguments (PHP 8+):

```
<?php  
// array_fill(start_index, count, value)  
array_fill(start_index: 0, count: 100, value: 50);  
// Podem canviar l'ordre i saltar-nos paràmetres amb valor per defecte  
?>
```

Passar per referència:

```
<?php  
function incrementa(int &$valor): void {  
    $valor++;  
}  
$x = 5;  
incrementa($x);  
echo $x; // 6  
?>
```

Funcions anònimes i funcions fletxa:

```
<?php  
// Closure (funció anònima)  
$doble = function($n) { return $n * 2; };  
  
// Captura de variables externes  
$factor = 3;  
$multiplica = function($n) use ($factor) { return $n * $factor; };  
  
// Funció fletxa (PHP 7.4+): captura automàtica de l'entorn  
$multiplica = fn($n) => $n * $factor;  
  
// Ús habitual en array_map, array_filter  
$quadrats = array_map(fn($n) => $n ** 2, [1, 2, 3, 4]);  
$parells = array_filter([1,2,3,4,5,6], fn($n) => $n % 2 === 0);  
?>
```

Variables variables:

```
<?php  
$camp = 'títol';  
$$camp = 'El meu títol';  
echo $títol; // "El meu títol"  
?>
```

8. Gestió d'errors

Nivells d'error

Constant	Descripció
E_ERROR	Errors fatals que aturen l'execució
E_WARNING	Avisos que no aturen l'execució
E_NOTICE	Avisos lleugers (ús de variables no definides, etc.)
E_PARSE	Errors de sintaxi
E_ALL	Tots els errors i avisos

Configuració de la gestió d'errors

```
<?php
// En fitxer php (sobreescriu php.ini temporalment)
error_reporting(E_ALL);           // mostra tots els errors
ini_set('display_errors', '1');   // mostra errors per pantalla
    ⇨ (NOMÉS dev)
ini_set('log_errors', '1');       // desa errors en un fitxer
    ⇨ de log
ini_set('error_log', '/var/log/php-errors.log');
?>
```

PRECAUCIÓ

Mai activis `display_errors` en un servidor de producció: podria revelar rutes, usuaris i contrasenyes.

Gestor d'errors personalitzat

```
<?php
function gestorErrors(int $errno, string $errstr, string $errfile, int
    ⇨ $errline): bool {
    $missatge = "[" . date('Y-m-d H:i:s') . "] Error $errno: $errstr a
        ⇨ $errfile línia $errline";
    error_log($missatge);

    if ($errno === E_USER_ERROR) {
        echo "<p>S'ha produït un error. Si us plau, contacteu
            ⇨ l'administrador.</p>";
        exit(1);
    }
    return true; // no executar el gestor intern de PHP
}

set_error_handler("gestorErrors");

// Generar errors personalitzats
```

```
trigger_error("Valor invàlid", E_USER_WARNING);
trigger_error("Error crític", E_USER_ERROR);
?>
```

Excepcions

```
<?php
declare(strict_types=1);

// Excepció personalitzada
class ValidacióException extends RuntimeException {
    private array $errors;

    public function __construct(array $errors) {
        parent::__construct(implode(', ', $errors));
        $this->errors = $errors;
    }

    public function getErrors(): array { return $this->errors; }
}

// Ús
function validaNom(string $nom): string {
    if (empty($nom)) throw new ValidacióException(["El nom és
        ↳ obligatori"]);
    if (strlen($nom) < 2) throw new ValidacióException(["El nom és
        ↳ massa curt"]);
    return trim($nom);
}

try {
    $nom = validaNom("");
    echo "Nom vàlid: $nom";
} catch (ValidacióException $e) {
    foreach ($e->getErrors() as $error) {
        echo "- $error\n";
    }
} catch (Exception $e) {
    echo "Error inesperat: " . $e->getMessage();
} finally {
    echo "Sempre s'executa";
}
?>
```

Gestor global d'excepcions

```
<?php
set_exception_handler(function(Throwable $e) {
    error_log("[EXCEPCIÓ] " . $e->getMessage() . " a " . $e->getFile()
        . ":" . $e->getLine());
    http_response_code(500);
    echo "<h1>Error intern del servidor</h1>";
    // En desenvolupament podries mostrar $e->getMessage()
});
?>
```

9. Formularis: introducció d'informació

Formulari HTML i processament PHP

```
<!-- formulari.html -->
<form method="POST" action="processar.php">
    <label>Nom: <input type="text" name="nom" required></label>
    <label>Email: <input type="email" name="email"></label>
    <label>Edat: <input type="number" name="edat" min="0"
        ↪ max="120"></label>
    <label>
        Gènere:
        <select name="genere">
            <option value="h">Home</option>
            <option value="d">Dona</option>
            <option value="a">Altres</option>
        </select>
    </label>
    <label>
        <input type="checkbox" name="accepta_condicions" value="1">
        Accepto les condicions
    </label>
    <button type="submit">Enviar</button>
</form>
```

```
<?php
// processar.php
declare(strict_types=1);

if ($_SERVER['REQUEST_METHOD'] !== 'POST') {
    header('Location: formulari.html');
    exit;
}

// 1. Recollir i netejar
$nom = trim(htmlspecialchars($_POST['nom'] ?? ''));
$email = trim($_POST['email'] ?? '');
```

```

$edat = (int) ($_POST['edat'] ?? 0);
$accepta = isset($_POST['accepta_condicions']);

// 2. Validar
$errors = [];

if (empty($nom)) {
    $errors[] = "El nom és obligatori";
} elseif (mb_strlen($nom) < 2) {
    $errors[] = "El nom ha de tenir almenys 2 caràcters";
}

if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
    $errors[] = "L'adreça de correu no és vàlida";
}

if ($edat < 0 || $edat > 120) {
    $errors[] = "L'edat ha de ser entre 0 i 120";
}

if (!$accepta) {
    $errors[] = "Cal acceptar les condicions";
}

// 3. Mostrar resultat
if (empty($errors)) {
    echo "<p>Dades rebudes correctament: $nom, $email, $edat
    ↪ anys</p>";
    // Aquí vindria la lògica: desar a BD, enviar email, etc.
} else {
    echo "<ul class='errors'>";
    foreach ($errors as $error) {
        echo "<li>" . htmlspecialchars($error) . "</li>";
    }
    echo "</ul>";
}
?>

```

Funcions de filtre i validació

```

<?php
// Validació
filter_var($email, FILTER_VALIDATE_EMAIL); // email
filter_var($url, FILTER_VALIDATE_URL); // URL
filter_var($ip, FILTER_VALIDATE_IP); // IP
filter_var($n, FILTER_VALIDATE_INT, ['options' => ['min_range' =>
    ↪ 0, 'max_range' => 120]]);
filter_var($n, FILTER_VALIDATE_FLOAT);
filter_var($b, FILTER_VALIDATE_BOOLEAN);

// Sanejament (elimina caràcters no desitjats)

```

```

filter_var($input, FILTER_SANITIZE_NUMBER_INT);
filter_var($input, FILTER_SANITIZE_EMAIL);
filter_var($input, FILTER_SANITIZE_URL);

// Equivalent manual per a HTML
htmlspecialchars($input, ENT_QUOTES, 'UTF-8');
strip_tags($input);
trim($input);
?>

```

Pujada de fitxers

```

<form action="pujar.php" method="POST" enctype="multipart/form-data">
    <input type="file" name="imatge" accept="image/*">
    <button type="submit">Pujar</button>
</form>

```

```

<?php
// pujar.php
$tipus_permesos = ['image/jpeg', 'image/png', 'image/gif',
    ↪ 'image/webp'];
$mida_maxima    = 2 * 1024 * 1024; // 2 MB
$directori_dest = __DIR__ . '/uploads/';

if (!isset($_FILES['imatge']) || $_FILES['imatge']['error'] !==
    ↪ UPLOAD_ERR_OK) {
    die("Error en la pujada del fitxer");
}

$fitxer = $_FILES['imatge'];

// Verificar tipus MIME real (no el que diu el navegador)
$finfo = new finfo(FILEINFO_MIME_TYPE);
$tipus_real = $finfo->file($fitxer['tmp_name']);

if (!in_array($tipus_real, $tipus_permesos)) {
    die("Tipus de fitxer no permès: $tipus_real");
}

if ($fitxer['size'] > $mida_maxima) {
    die("El fitxer és massa gran");
}

// Generar nom únic per evitar sobrescriure fitxers existents
$extensio = pathinfo($fitxer['name'], PATHINFO_EXTENSION);
$nom_segur = uniqid('img_', true) . '.' . strtolower($extensio);
$ruta_dest = $directori_dest . $nom_segur;

if (move_uploaded_file($fitxer['tmp_name'], $ruta_dest)) {

```

```

        echo "Fitxer pujat correctament: $nom_segur";
    } else {
        die("No s'ha pogut moure el fitxer");
    }
?>

```

10. Autenticació d'usuaris

Emmagatzematge segur de contrasenyes

```

<?php
// GENERAR HASH (en registrar l'usuari)
$clau_text_pla = $_POST['clau'];
$hash = password_hash($clau_text_pla, PASSWORD_DEFAULT);
// Desa $hash a la BD (camp VARCHAR(255))
// Exemple: $2y$10$6z7GKa9kpDN7KC3ICW1Hi.f0/7Y/...

// VERIFICAR (en fer login)
if (password_verify($clau_introduïda, $hash_de_bd)) {
    echo "Contrasenya correcta";
}

// Actualitzar l'algoritme si cal (cost computacional ha augmentat)
if (password_needs_rehash($hash, PASSWORD_DEFAULT)) {
    $hash_nou = password_hash($clau_text_pla, PASSWORD_DEFAULT);
    // Actualitza el hash a la BD
}
?>

```

PASSWORD_DEFAULT usa bcrypt. PHP fa evolucionar l'algoritme amb el temps. No usar SHA1, MD5 ni SHA256 directament: són massa ràpids i vulnerables a atacs de força bruta.

Procés de login complet

```

<?php
// login.php
declare(strict_types=1);
session_start();

// Si ja hi ha sessió, redirigir
if (isset($_SESSION['user_id'])) {
    header('Location: dashboard.php');
    exit;
}

$error = '';

```

```

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $usuari = trim($_POST['usuari'] ?? '');
    $clau = $_POST['clau'] ?? '';

    if (empty($usuari) || empty($clau)) {
        $error = "Omple tots els camps";
    } else {
        try {
            $pdo = new PDO('mysql:host=localhost;dbname=app;charset=
                ↪ utf8mb4', 'user', 'pass',
                [PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION]);
            $stmt = $pdo->prepare("SELECT id, nom, clau_hash FROM
                ↪ usuarios WHERE email = ? LIMIT 1");
            $stmt->execute([$usuari]);
            $user = $stmt->fetch(PDO::FETCH_ASSOC);

            if ($user && password_verify($clau, $user['clau_hash'])) {
                // Login correcta
                session_regenerate_id(true); // evita fixació de sessió
                $_SESSION['user_id'] = $user['id'];
                $_SESSION['nom'] = $user['nom'];
                $_SESSION['rol'] = $user['rol'] ?? 'usuari';

                $dest = $_GET['backto'] ?? 'dashboard.php';
                header('Location: ' . $dest);
                exit;
            } else {
                $error = "Credencials incorrectes";
                // Registrar intent fallat (per a fail2ban o similar)
                error_log("Login fallat per a l'usuari: $usuari des de
                    ↪ " . $_SERVER['REMOTE_ADDR']);
            }
        } catch (PDOException $e) {
            error_log("Error BD en login: " . $e->getMessage());
            $error = "Error intern. Torna-ho a provar.";
        }
    }
}

?>
<!DOCTYPE html>
<html lang="ca">
<body>
<?php if ($error): ?>
    <p class="error"><?= htmlspecialchars($error) ?></p>
<?php endif; ?>
<form method="POST">
    <input type="email" name="usuari" placeholder="Email" required>
    <input type="password" name="clau" placeholder="Contrasenya"
        ↪ required>
    <button type="submit">Entrar</button>
</form>
</body>
</html>

```

11. Control d'accessos

Verificació de sessió (protecció de pàgines)

```
<?php
// auth.php --- incloure a l'inici de cada pàgina protegida
declare(strict_types=1);

function requereixLogin(string $redirigir_a = 'login.php'): void {
    if (session_status() === PHP_SESSION_NONE) {
        session_start();
    }
    if (!isset($_SESSION['user_id'])) {
        $backto = urlencode($_SERVER['REQUEST_URI']);
        header("Location: $redirigir_a?backto=$backto");
        exit;
    }
}

function requereixRol(string ...$rols_permesos): void {
    requereixLogin();
    $rol_usuari = $_SESSION['rol'] ?? 'usuari';
    if (!in_array($rol_usuari, $rols_permesos, true)) {
        http_response_code(403);
        echo "<h1>403 --- Accés denegat</h1>";
        echo "<p>No tens permís per accedir a aquesta pàgina.</p>";
        exit;
    }
}
?>
```

Ús del control d'accés

```
<?php
// dashboard.php --- pàgina accessible per a qualsevol usuari
↳ autenticat
require_once 'auth.php';
requereixLogin();

echo "Benvingut, " . htmlspecialchars($_SESSION['nom']);
?>

<?php
// admin.php --- pàgina accessible només per a administradors
require_once 'auth.php';
requereixRol('admin');

echo "Panell d'administrador";
?>

<?php
```

```
// editor.php --- pàgina accessible per a editors i administradors
require_once 'auth.php';
requereixRol('editor', 'admin');
?>
```

Sistema de rols

```
<?php
// Definir permisos per rol
const PERMISOS = [
    'usuari'    => ['llegir'],
    'editor'    => ['llegir', 'escriure', 'editar'],
    'admin'     => ['llegir', 'escriure', 'editar', 'esborrar',
        ↪ 'gestionar_usuaris'],
    'superadmin'=> ['*'], // tots els permisos
];

function tePermis(string $permis): bool {
    $rol = $_SESSION['rol'] ?? 'usuari';
    $permisos_rol = PERMISOS[$rol] ?? [];
    return in_array('*', $permisos_rol) || in_array($permis,
        ↪ $permisos_rol);
}

// Ús a les vistes
if (tePermis('esborrar')): ?>
    <button class="btn-danger">Esborrar article</button>
<?php endif; ?>
```

Protecció CSRF (Cross-Site Request Forgery)

```
<?php
// Generar token CSRF
function generaTokenCSRF(): string {
    if (empty($_SESSION['csrf_token'])) {
        $_SESSION['csrf_token'] = bin2hex(random_bytes(32));
    }
    return $_SESSION['csrf_token'];
}

// Validar token CSRF
function validaTokenCSRF(): bool {
    $token_rebut = $_POST['csrf_token'] ?? '';
    $token_sessio = $_SESSION['csrf_token'] ?? '';
    return hash_equals($token_sessio, $token_rebut);
}
?>
```

```

<!-- Al formulari -->
<form method="POST" action="processar.php">
    <input type="hidden" name="csrf_token" value="<?=
        ↪ generaTokenCSRF() ?>">
    <!-- resta del formulari -->
</form>

<?php
// En processar
if (!validaTokenCSRF()) {
    http_response_code(403);
    die("Token CSRF invàlid");
}
?>

```

12. Sessions

Cicle de vida d'una sessió

```

<?php
// 1. Iniciar (a l'inici de CADA pàgina que usa sessions, abans de
    ↪ qualsevol sortida)
session_start();

// 2. Emmagatzemar dades
$_SESSION['user_id'] = 42;
$_SESSION['nom']      = 'Joan';
$_SESSION['rol']      = 'admin';
$_SESSION['darrer_accés'] = time();

// 3. Llegir dades
echo $_SESSION['nom']; // Joan
echo isset($_SESSION['nom']) ? 'sí' : 'no';

// 4. Eliminar una variable de sessió
unset($_SESSION['darrer_accés']);

// 5. Tancar la sessió completament (logout)
$_SESSION = []; // buidar totes les variables
if (ini_get("session.use_cookies")) { // destruir la cookie de sessió
    $params = session_get_cookie_params();
    setcookie(
        session_name(), '', time() - 42000,
        $params['path'], $params['domain'],
        $params['secure'], $params['httponly']
    );
}
session_destroy();
?>

```

Configuració de sessions

```
<?php
// Modificar opcions en cridar session_start()
session_start([
    'cookie_lifetime' => 3600,           // la cookie expira en 1 hora
    'cookie_secure'   => true,           // HTTPS obligatori
    'cookie_httponly' => true,           // inaccessible per JavaScript
    'cookie_samesite' => 'Strict',       // prevenció CSRF
    'gc_maxlifetime'  => 3600,           // dades de sessió al
    // servidor: 1 hora
]);

// Regenerar ID per evitar fixació de sessió (fer-ho en el login)
session_regenerate_id(true);
?>
```

Exemple d'ús per verificar aïllament entre usuaris

```
<?php
// pàgina_personal.php
session_start();

if (!isset($_SESSION['user_id'])) {
    header('Location: login.php');
    exit;
}

// Cada usuari veu NOMÉS les seves dades
$user_id = (int) $_SESSION['user_id']; // cast a int per seguretat

$pdo = connexioBD();
$stmt = $pdo->prepare("SELECT * FROM comandes WHERE usuari_id = ?");
$stmt->execute([$user_id]);
$comandes = $stmt->fetchAll();

// Cada usuari veu sols les seves comandes
// L'aïllament el garanteix la sessió + la consulta SQL amb WHERE
//   ↳ usuari_id = ?
?>
```

13. Configuració de l'interpret

Fitxer php.ini

El fitxer principal de configuració de PHP. A Ubuntu 26.04 amb Apache, la ubicació és:

Sistema	Ubicació
Ubuntu 26.04 + Apache	/etc/php/8.x/apache2/php.ini
Ubuntu 26.04 (CLI)	/etc/php/8.x/cli/php.ini

Per saber la versió exacta de PHP i el fitxer que s'usa:

```
php -v                # mostra la versió (ex. 8.3.x)
php --ini              # mostra el php.ini que usa la CLI
apache2ctl -t -D DUMP_MODULES # comprova que el mòdul PHP és actiu
```

Opcions clau:

```
; ===== ERRORS (DEV) =====
display_errors = On
error_reporting = E_ALL
log_errors = On
error_log = /var/log/php_errors.log

; ===== ERRORS (PRODUCCIÓ) =====
display_errors = Off
display_startup_errors = Off
error_reporting = E_ALL & ~E_DEPRECATED & ~E_STRICT
log_errors = On
error_log = /var/log/php_errors.log

; ===== FITXERS PUJATS (recomanat per a WordPress) =====
file_uploads = On
upload_max_filesize = 64M
post_max_size = 68M           ; ha de ser > upload_max_filesize
max_file_uploads = 20

; ===== RENDIMENT =====
memory_limit = 256M           ; WordPress recomana 256M
max_execution_time = 60
max_input_time = 60
max_input_vars = 3000         ; necessari per a temes/plugins complexos

; ===== EXTENSIONS (ja actives si s'ha seguit la instal·lació) =====
; Les extensions s'activen com a paquets APT a Ubuntu,
; no editant php.ini directament. Comprova-les amb phpinfo().

; ===== SESSIONS =====
session.cookie_httponly = 1
session.cookie_secure = 1     ; requereix HTTPS
session.gc_maxlifetime = 1440
session.use_strict_mode = 1
```

```
; ===== OPCACHE (rendiment) =====  
opcache.enable = 1  
opcache.memory_consumption = 128  
opcache.max_accelerated_files = 4000
```

Després de modificar `php.ini`, cal reiniciar Apache:

```
sudo systemctl restart apache2
```

Modificar configuració temporalment des de PHP

```
<?php  
ini_set('display_errors', '1');  
ini_set('memory_limit', '256M');  
ini_set('max_execution_time', '120');  
  
echo ini_get('memory_limit'); // 256M  
?>
```

RA6 - Genera documents web amb accés a bases de dades utilitzant llenguatges de guions de servidors.

14. Integració PHP i MySQL

Sistemes gestors de bases de dades en entorns web

SGBD	Característiques	Usos habituals
MySQL / MariaDB	El més usat en web, ràpid, codi obert	WordPress, LAMP
PostgreSQL	Funcionalitats avançades, ACID complet	Aplicacions empresarials, geolocalització
SQLite	Fitxer únic, sense servidor	Apps mòbils, prototips, tests
MongoDB	NoSQL orientat a documents	APIs, big data, dades no estructurades
Redis	Clau-valor en memòria	Caching, sessions, cues de missatges

API PHP per a MySQL: PDO vs. MySQLi

PHP ofereix dues API per connectar-se a MySQL:

	PDO	MySQLi
Portabilitat	Suporta 12 SGBD	Només MySQL/MariaDB
Prepared statements	Sí (nomenats i posicionals)	Sí (només posicionals)
Estil	OO	OO i procedural
Recomanació	Preferida	Per a funcionalitats MySQL específiques

En aquest document usem **PDO** per portabilitat.

Activar les extensions necessàries

A Ubuntu 26.04, les extensions de PHP s'instal·len com a paquets APT, no editant `php.ini` directament:

```
# Ubuntu 26.04 --- instal·lar extensions per a MariaDB/MySQL
sudo apt install php-mysql php-mbstring php-xml php-curl \
    php-zip php-gd php-intl php-bcmath -y
sudo systemctl restart apache2
```

Per verificar que les extensions estan actives:

```
php -m | grep -E "pdo_mysql|mysqli|mbstring|gd|curl"
```

O bé consultant `phpinfo()` al navegador.

15. Connexió a bases de dades

Connexió bàsica amb PDO

```
<?php
declare(strict_types=1);

function connexioBD(): PDO {
    $dsn = 'mysql:host=localhost;dbname=la_meva_bd;charset=utf8mb4';
    $usuari = 'nom_usuari';
    $clau = 'contrasenya';

    $opcions = [
        PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION, //
        ↪ llença excepcions
        PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC, //
        ↪ arrays associatives
        PDO::ATTR_EMULATE_PREPARES => false, //
        ↪ prepared statements reals
    ];

    try {
        return new PDO($dsn, $usuari, $clau, $opcions);
    } catch (PDOException $e) {
        // No mostrar detalls de connexió per pantalla en producció
        error_log("Error de connexió BD: " . $e->getMessage());
        throw new RuntimeException("No s'ha pogut connectar a la base
            ↪ de dades");
    }
}

// Ús
$pdo = connexioBD();
```

Fitxer de configuració separat

```
<?php
// config/database.php
return [
    'host' => 'localhost',
    'dbname' => 'la_meva_bd',
    'charset' => 'utf8mb4',
    'username' => 'nom_usuari',
    'password' => 'contrasenya',
];
```

```
<?php
// Carregar configuració
$config = require __DIR__ . '/config/database.php';
```

```
$dsn = "mysql:host={$config['host']};dbname={$config['dbname']};char_
↳ set={$config['charset']}";
$pdo = new PDO($dsn, $config['username'], $config['password']);
```

Desconnexió

```
<?php
$pdo = null; // allibera la connexió
?>
```

16. Creació de bases de dades i taules

Crear una base de dades

```
<?php
// Connexió sense especificar dbname per poder crear la BD
$pdo = new PDO('mysql:host=localhost;charset=utf8mb4', 'root',
↳ 'rootpass',
    [PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION]);

$pdo->exec("CREATE DATABASE IF NOT EXISTS botiga
        CHARACTER SET utf8mb4
        COLLATE utf8mb4_unicode_ci");

echo "Base de dades creada";

// Seleccionar-la
$pdo->exec("USE botiga");
?>
```

Crear taules

```
<?php
$pdo = connexioBD();

// Taula d'usuaris
$pdo->exec("
    CREATE TABLE IF NOT EXISTS usuaris (
        id            INT AUTO_INCREMENT PRIMARY KEY,
        nom            VARCHAR(100)        NOT NULL,
        email          VARCHAR(255)        NOT NULL UNIQUE,
        clau_hash      VARCHAR(255)        NOT NULL,
        rol            ENUM('usuari','editor','admin') DEFAULT 'usuari',
        actiu          TINYINT(1)          DEFAULT 1,
        creat_el       TIMESTAMP            DEFAULT CURRENT_TIMESTAMP,
```

```

        modificat_el TIMESTAMP          DEFAULT CURRENT_TIMESTAMP ON
        ↪ UPDATE CURRENT_TIMESTAMP
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci
");

// Taula de productes
$pdo->exec("
    CREATE TABLE IF NOT EXISTS productes (
        id            INT AUTO_INCREMENT PRIMARY KEY,
        nom            VARCHAR(200)      NOT NULL,
        descripcio    TEXT,
        preu           DECIMAL(10,2)     NOT NULL,
        estoc          INT                DEFAULT 0,
        actiu          TINYINT(1)        DEFAULT 1,
        creat_el       TIMESTAMP          DEFAULT CURRENT_TIMESTAMP
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci
");

// Taula de comandes (clau forana)
$pdo->exec("
    CREATE TABLE IF NOT EXISTS comandes (
        id            INT AUTO_INCREMENT PRIMARY KEY,
        usuari_id     INT                NOT NULL,
        total         DECIMAL(10,2)     NOT NULL,
        estat         ENUM('pendent','pagada','enviada','cancel·lada')
        ↪ DEFAULT 'pendent',
        creat_el       TIMESTAMP          DEFAULT CURRENT_TIMESTAMP,
        FOREIGN KEY (usuari_id) REFERENCES usuaris(id) ON DELETE
        ↪ CASCADE
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci
");

echo "Taules creades correctament";
?>

```

17. Consulta de dades (SELECT)

Consulta simple

```
<?php
$pdo = connexioBD();

// Tots els productes actius
$stmt = $pdo->query("SELECT * FROM productes WHERE actiu = 1 ORDER BY
↳ nom");
$productes = $stmt->fetchAll(); // array de tots els registres

foreach ($productes as $p) {
    echo "{$p['nom']}: {$p['preu']} €<br>";
}

// Nombre de files
echo "Total: " . $stmt->rowCount();
?>
```

Consulta amb paràmetres (SEMPRE per a dades externes)

```
<?php
$pdo = connexioBD();

// Paràmetres nomenats (:nom)
$sql = "SELECT * FROM productes
    WHERE preu < :preu_max AND actiu = 1
    ORDER BY preu ASC
    LIMIT :limit";

$stmt = $pdo->prepare($sql);
$stmt->bindValue(':preu_max', 50.0, PDO::PARAM_STR); // float usa
↳ PARAM_STR
$stmt->bindValue(':limit', 10, PDO::PARAM_INT);
$stmt->execute();
$productes = $stmt->fetchAll();

// O amb execute() directament:
$stmt = $pdo->prepare("SELECT * FROM productes WHERE preu < ? AND
↳ actiu = 1");
$stmt->execute([50.0]);
$productes = $stmt->fetchAll();
?>
```

Obtenir un sol registre

```
<?php
$pdo = connexionBD();

$stmt = $pdo->prepare("SELECT * FROM usuaris WHERE id = ?");
$stmt->execute([$id]);
$usuari = $stmt->fetch(); // un sol registre o false si no existeix

if ($usuari) {
    echo "Nom: {$usuari['nom']}";
} else {
    echo "Usuari no trobat";
}

// Obtenir un sol valor escalar
$stmt = $pdo->prepare("SELECT COUNT(*) FROM productes WHERE actiu =
↳ 1");
$stmt->execute();
$total = $stmt->fetchColumn();
echo "Total actius: $total";
?>
```

Consulta amb JOIN

```
<?php
$sql = "
    SELECT c.id, c.total, c.estat, u.nom AS nom_usuari, u.email
    FROM comandes c
    JOIN usuaris u ON c.usuari_id = u.id
    WHERE c.estat = ?
    ORDER BY c.creat_el DESC
";

$stmt = $pdo->prepare($sql);
$stmt->execute(['pendent']);
$comandes = $stmt->fetchAll();
?>
```

18. Modificació de dades (INSERT, UPDATE, DELETE)

INSERT

```
<?php
$pdo = connexioBD();

// Inserció simple
$stmt = $pdo->prepare("
    INSERT INTO productes (nom, descripcio, preu, estoc)
    VALUES (:nom, :descripcio, :preu, :estoc)
");

$stmt->execute([
    ':nom'          => 'Teclat mecànic',
    ':descripcio'   => 'Teclat mecànic retroil·luminat',
    ':preu'         => 79.99,
    ':estoc'        => 50,
]);

$id_nou = (int) $pdo->lastInsertId();
echo "Producte inserit amb ID: $id_nou";

// Inserció múltiple (transacció per eficiència)
$productes = [
    ['Ratolí ergonòmic', 29.99, 100],
    ['Monitor 24"',      199.00, 20],
    ['Auriculars BT',    49.99, 75],
];

$stmt = $pdo->prepare("INSERT INTO productes (nom, preu, estoc) VALUES
    ↪  (?, ?, ?)");

$pdo->beginTransaction();
try {
    foreach ($productes as $p) {
        $stmt->execute($p);
    }
    $pdo->commit();
    echo "Inserció múltiple completada";
} catch (PDOException $e) {
    $pdo->rollBack();
    error_log("Error en inserció múltiple: " . $e->getMessage());
    throw $e;
}
?>
```

UPDATE

```
<?php
$pdo = connexioBD();

// Actualitzar un registre
$stmt = $pdo->prepare("
    UPDATE productes
    SET preu = :preu, estoc = :estoc, modificat_el = NOW()
    WHERE id = :id AND actiu = 1
");

$stmt->execute([
    ':preu' => 89.99,
    ':estoc' => 45,
    ':id'    => 1,
]);

echo "Files actualitzades: " . $stmt->rowCount();

// Actualitzar múltiples registres
$stmt = $pdo->prepare("UPDATE productes SET actiu = 0 WHERE estoc =
    ↪ 0");
$stmt->execute();
echo "Productes desactivats: " . $stmt->rowCount();
?>
```

DELETE

```
<?php
$pdo = connexioBD();

// Esborrat lògic (recomanat: no elimina el registre)
$stmt = $pdo->prepare("UPDATE productes SET actiu = 0 WHERE id = ?");
$stmt->execute([$id_producte]);

// Esborrat físic
$stmt = $pdo->prepare("DELETE FROM productes WHERE id = ?");
$stmt->execute([$id_producte]);
echo "Files esborrades: " . $stmt->rowCount();
?>
```

Transaccions

```
<?php
$pdo = connexioBD();

try {
    $pdo->beginTransaction();

    // Operació 1: crear comanda
    $stmt = $pdo->prepare("INSERT INTO comandes (usuari_id, total)
        ↳ VALUES (?, ?)");
    $stmt->execute([$usuari_id, $total]);
    $comanda_id = (int) $pdo->lastInsertId();

    // Operació 2: reduir estoc
    $stmt = $pdo->prepare("UPDATE productes SET estoc = estoc - ?
        ↳ WHERE id = ? AND estoc >= ?");
    $stmt->execute([$quantitat, $producte_id, $quantitat]);

    if ($stmt->rowCount() === 0) {
        throw new RuntimeException("Estoc insuficient");
    }

    $pdo->commit();
    echo "Comanda $comanda_id creada correctament";

} catch (Exception $e) {
    $pdo->rollBack();
    error_log("Error en transacció: " . $e->getMessage());
    echo "Error: " . $e->getMessage();
}
?>
```

19. Verificació i validació de la informació

Validació de dades d'entrada

```
<?php
declare(strict_types=1);

/**
 * Valida i neteja les dades d'un producte.
 * @throws InvalidArgumentException si les dades no són vàlides
 */
function validaProducte(array $dades): array {
    $errors = [];

    $nom = trim($dades['nom'] ?? '');
    if (empty($nom)) {
        $errors[] = "El nom és obligatori";
    } elseif (mb_strlen($nom) > 200) {
        $errors[] = "El nom no pot superar 200 caràcters";
    }

    $preu = filter_var($dades['preu'] ?? '', FILTER_VALIDATE_FLOAT);
    if ($preu === false || $preu < 0) {
        $errors[] = "El preu ha de ser un número positiu";
    }

    $estoc = filter_var($dades['estoc'] ?? '', FILTER_VALIDATE_INT,
        ['options' => ['min_range' => 0]]);
    if ($estoc === false) {
        $errors[] = "L'estoc ha de ser un enter no negatiu";
    }

    if (!empty($errors)) {
        throw new InvalidArgumentException(implode('; ', $errors));
    }

    return [
        'nom' => htmlspecialchars($nom, ENT_QUOTES, 'UTF-8'),
        'preu' => (float) $preu,
        'estoc' => (int) $estoc,
    ];
}

// Ús
try {
    $dades_validades = validaProducte($_POST);
    // Inserir a la BD...
} catch (InvalidArgumentException $e) {
    echo "Error de validació: " . $e->getMessage();
}
?>
```

Verificació de l'existència de registres

```
<?php
function existeixEmail(PDO $pdo, string $email, ?int $excloure_id =
    ↪ null): bool {
    $sql = "SELECT COUNT(*) FROM usuaris WHERE email = ?";
    $params = [$email];

    if ($excloure_id !== null) {
        $sql .= " AND id != ?";
        $params[] = $excloure_id;
    }

    $stmt = $pdo->prepare($sql);
    $stmt->execute($params);
    return (int) $stmt->fetchColumn() > 0;
}

if (existeixEmail($pdo, $_POST['email'])) {
    $errors[] = "Ja existeix un compte amb aquest email";
}
?>
```

20. Gestió d'errors de BD

Modes d'error de PDO

```
<?php
$options = [
    // ERRMODE_SILENT (defecte): errors silenciosos, cal comprovar
    ↪ manualment
    PDO::ATTR_ERRMODE => PDO::ERRMODE_SILENT,

    // ERRMODE_WARNING: genera un E_WARNING (no atura l'execució)
    PDO::ATTR_ERRMODE => PDO::ERRMODE_WARNING,

    // ERRMODE_EXCEPTION (recomanat): llença PDOException
    PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION,
];
?>
```

Gestió d'errors en operacions de BD

```
<?php
function insereixProducte(PDO $pdo, array $dades): int {
    try {
        $stmt = $pdo->prepare("
            INSERT INTO productes (nom, preu, estoc)
            VALUES (:nom, :preu, :estoc)
        ");
        $stmt->execute($dades);
        return (int) $pdo->lastInsertId();

    } catch (PDOException $e) {
        $codi = (int) $e->getCode();

        // Error de clau duplicada (SQLSTATE 23000)
        if ($codi === 23000) {
            throw new RuntimeException("Ja existeix un producte amb
                ↪ aquest nom");
        }

        // Altres errors: registrar i relllençar
        error_log("Error BD insereixProducte: [{ $e->getCode() }]
            ↪ { $e->getMessage() }");
        throw new RuntimeException("Error en desar el producte.
            ↪ Torna-ho a provar.");
    }
}
?>
```

21. Seguretat i control d'accés a la BD

Prevenició d'injeccions SQL

```
<?php
// VULNERABLE --- MAI fer-ho així
$id = $_GET['id'];
$sql = "SELECT * FROM usuaris WHERE id = $id";
// Si id = "1 OR 1=1", retorna tots els usuaris

// VULNERABLE --- escapament manual insuficient
$id = mysql_real_escape_string($_GET['id']); // funció eliminada a PHP
↪ 7

// CORRECTE --- prepared statements
$stmt = $pdo->prepare("SELECT * FROM usuaris WHERE id = ?");
$stmt->execute([(int) $_GET['id']]);
$usuari = $stmt->fetch();
?>
```

Principi de mínim privilegi

```
-- Crear un usuari de BD amb NOMÉS els permisos necessaris
CREATE USER 'app_user'@'localhost' IDENTIFIED BY 'contrasenya_segura';

-- Permisos mínims per a l'aplicació web (no DROP, no CREATE, no GRANT)
GRANT SELECT, INSERT, UPDATE, DELETE ON botiga.* TO
    ↪ 'app_user'@'localhost';
FLUSH PRIVILEGES;
```

Protecció de les credencials

```
<?php
// MAL: credencials al codi
$pdo = new PDO('mysql:host=localhost;dbname=bd', 'root',
    ↪ 'password123');

// BÉ: credencials en un fitxer fora de l'arrel web
// Estructura del projecte:
// /var/www/html/    ↪ arrel web (accessible)
//   index.php
//   config/         ↪ cal protegir amb .htaccess o moure fora de l'arrel
// /etc/app/         ↪ fora de l'arrel web (no accessible)
//   database.php

// database.php (fora de l'arrel web)
return [
    'host' => getenv('DB_HOST') ?: 'localhost',
    'name' => getenv('DB_NAME') ?: 'botiga',
    'user' => getenv('DB_USER') ?: 'app_user',
    'pass' => getenv('DB_PASS') ?: '',
];
?>
```

```
# .htaccess per protegir el directori config/
<Directory /var/www/html/config>
    Require all denied
</Directory>
```

22. Verificació del funcionament i proves de rendiment

Comprovació bàsica de la BD

```
<?php
// Verificar la connexió
try {
    $pdo = connexioBD();
    $stmt = $pdo->query("SELECT 1");
    echo "Connexió a la BD: OK";
} catch (Exception $e) {
    echo "Connexió a la BD: FALLIDA --- " . $e->getMessage();
}

// Informació del servidor MySQL
$stmt = $pdo->query("SELECT VERSION() AS versio, NOW() AS ara");
$info = $stmt->fetch();
echo "MySQL versió: {$info['versio']}, Hora servidor: {$info['ara']}";
?>
```

Mesura del temps d'execució

```
<?php
$inici = microtime(true);

// Consulta a mesurar
$stmt = $pdo->query("SELECT * FROM productes WHERE actiu = 1");
$productes = $stmt->fetchAll();

$temps = (microtime(true) - $inici) * 1000;
echo sprintf("Consulta: %.2f ms, %d registres", $temps,
    ↪ count($productes));
?>
```

Activar EXPLAIN per analitzar consultes lentes

```
<?php
$sql = "SELECT * FROM productes WHERE nom LIKE '%teclat%'";
$stmt = $pdo->query("EXPLAIN $sql");
$pla = $stmt->fetchAll();
print_r($pla);
// Mostra si s'usen índexs i el cost estimat de la consulta
?>
```

Paginació de resultats

```
<?php
$pagina          = max(1, (int) ($_GET['pagina'] ?? 1));
$resultats_pagina = 20;
$offset          = ($pagina - 1) * $resultats_pagina;

// Comptar total
$stmt = $pdo->query("SELECT COUNT(*) FROM productes WHERE actiu =
    ↳ 1");
$total = (int) $stmt->fetchColumn();
$total_pagines = (int) ceil($total / $resultats_pagina);

// Obtenir la pàgina actual
$stmt = $pdo->prepare("SELECT * FROM productes WHERE actiu = 1 LIMIT ?
    ↳ OFFSET ?");
$stmt->bindValue(1, $resultats_pagina, PDO::PARAM_INT);
$stmt->bindValue(2, $offset, PDO::PARAM_INT);
$stmt->execute();
$productes = $stmt->fetchAll();
?>
<p>Pàgina <?= $pagina ?> de <?= $total_pagines ?> (<?= $total ?>
    ↳ resultats)</p>
```

Bones pràctiques de rendiment

```
<?php
// 1. Usar índexs a les columnes que s'usen al WHERE, JOIN i ORDER BY
// (configuració a MySQL, no a PHP)

// 2. Seleccionar SOLS les columnes necessàries
$stmt = $pdo->query("SELECT id, nom, preu FROM productes"); // no
    ↳ SELECT *

// 3. Reutilitzar prepared statements en bucles
$stmt = $pdo->prepare("INSERT INTO log (event, dades) VALUES (?, ?)");
foreach ($events as $e) {
    $stmt->execute([$e['tipus'], json_encode($e['dades'])]);
}

// 4. Usar transaccions per a insercions múltiples (molt més ràpid)
$pdo->beginTransaction();
// ... múltiples inserts ...
$pdo->commit();

// 5. Limitar sempre els resultats
$stmt = $pdo->prepare("SELECT * FROM logs ORDER BY creat_el DESC LIMIT
    ↳ 100");
?>
```

RA7 - Realitza modificacions en gestors de continguts adaptant-ne l'aparença i les funcionalitats.

23. Introducció als gestors de continguts (CMS)

Què és un CMS?

Un **gestor de continguts** (*Content Management System*, CMS) és una aplicació web que permet crear, gestionar i publicar contingut digital sense necessitat de programar cada vegada. Separa el contingut de la presentació i ofereix una interfície d'administració gràfica.

Principals CMS

CMS	Quota mercat	Tecnologia	Usos típics
WordPress	~43% web mundial	PHP + MySQL	Blogs, webs corporatives, e-commerce
Drupal	~2%	PHP + MySQL	Portals complexos, grans institucions
Joomla	~2%	PHP + MySQL	Webs de tot mena
Magento/Adobe Commerce	---	PHP + MySQL	E-commerce
Shopify	---	Ruby	E-commerce (SaaS)
Ghost	---	Node.js	Blogs professionals

Dades de desembre de 2025 [Sketchweb](#) [Microblog](#)

WordPress és el CMS dominant i el que estudiem en detall per la seva penetració i arquitectura representativa.

24. Estructura de WordPress

Estructura de directoris

```
/var/www/html/wordpress/  
├─ wp-admin/           ← Panell d'administració (no modificar)  
├─ wp-includes/        ← Nucli de WordPress (no modificar)  
├─ wp-content/         ← Tot el contingut personalitzable  
│   ├── themes/        ← Temes (aparença)  
│   │   ├── twentytwentyseven/ ← Tema per defecte de WordPress 7  
│   │   └─ el-meu-tema/    ← Tema personalitzat  
│   ├── plugins/        ← Plugins (funcionalitats)  
│   │   ├── woocommerce/  
│   │   └─ el-meu-plugin/  ← Plugin personalitzat  
│   ├── uploads/        ← Fitxers pujats pels usuaris  
│   └─ languages/       ← Traduccions
```

Fitxers de configuració clau

```
// wp-config.php --- configuració principal (a l'arrel)
define('DB_NAME',      'nom_base_dades');
define('DB_USER',      'usuari_bd');
define('DB_PASSWORD', 'contrasenya_bd');
define('DB_HOST',      'localhost');
define('DB_CHARSET',   'utf8mb4');

// Claus de seguretat (generar a
↳ https://api.wordpress.org/secret-key/1.1/salt/)
define('AUTH_KEY', 'posar clau única aquí');
// ...

// Mode de debug (activar en desenvolupament)
define('WP_DEBUG', true);
define('WP_DEBUG_LOG', true); // desa errors a wp-content/debug.log
define('WP_DEBUG_DISPLAY', false); // no mostrar per pantalla en
↳ producció
```

Tipus de contingut (Post Types)

WordPress gestiona el contingut com a **posts** de diferent tipus:

Post Type	Descripció
post	Entrades del blog
page	Pàgines estàtiques
attachment	Fitxers multimèdia
custom	Qualsevol tipus personalitzat

Jerarquia de plantilles

WordPress segueix una jerarquia per escollir quina plantilla renderitza cada URL:

```
Pàgina concreta → page-{slug}.php → page-{id}.php → page.php
Entrada blog    → single-{type}.php → single.php → singular.php
Arxiu           → archive-{type}.php → archive.php
Cerca           → search.php
Error 404       → 404.php
Cap dels anteriors → index.php (obligatori)
```

25. Modificació de l'aparença: temes

Estructura mínima d'un tema

```
wp-content/themes/el-meu-tema/  
├── style.css           ← obligatori (capçalera amb metadades)  
├── index.php          ← obligatori (plantilla principal)  
├── functions.php      ← funcions del tema i hooks  
├── header.php         ← capçalera reutilitzable  
├── footer.php         ← peu reutilitzable  
├── sidebar.php        ← barra lateral reutilitzable  
├── single.php         ← plantilla per a entrades  
├── page.php           ← plantilla per a pàgines  
├── archive.php        ← plantilla per a arxius  
├── 404.php            ← pàgina d'error  
└── screenshot.png     ← previsualització al panell (1200×900 px)
```

style.css --- capçalera obligatòria

```
/*  
Theme Name:      El Meu Tema  
Theme URI:       https://exemple.cat/el-meu-tema  
Description:     Tema personalitzat per al mòdul 0376.  
Author:         Ramon López  
Author URI:      https://proferamon.com  
Version:         1.0.0  
License:         GNU General Public License v2 or later  
License URI:     https://www.gnu.org/licenses/gpl-2.0.html  
Text Domain:     el-meu-tema  
*/  
  
/* Aquí els estils CSS del tema */  
body {  
    font-family: 'Segoe UI', sans-serif;  
    line-height: 1.6;  
    color: #333;  
}
```

functions.php --- registrar funcionalitats

```
<?php  
// Activar funcionalitats del tema  
function el_meu_tema_setup(): void {  
    // Suport per a imatge destacada  
    add_theme_support('post-thumbnails');  
  
    // Suport per a títol dinàmic a <title>  
    add_theme_support('title-tag');
```

```

// Suport per a HTML5
add_theme_support('html5', ['search-form', 'comment-form',
    ↪ 'gallery']);

// Registrar menús de navegació
register_nav_menus([
    'primary' => __('Menú principal', 'el-meu-tema'),
    'footer'  => __('Menú del peu', 'el-meu-tema'),
]);
}
add_action('after_setup_theme', 'el_meu_tema_setup');

// Encolar CSS i JS (mai incloure'ls directament amb <link> o
    ↪ <script>)
function el_meu_tema_scripts(): void {
    wp_enqueue_style(
        'el-meu-tema-style',
        get_stylesheet_uri(),
        [],
        '1.0.0'
    );
    wp_enqueue_style(
        'el-meu-tema-fonts',
        'https://fonts.googleapis.com/css2?family=Inter:wght@400;600;700',
        ↪ '700',
        [],
        null
    );
    wp_enqueue_script(
        'el-meu-tema-script',
        get_template_directory_uri() . '/js/main.js',
        ['jquery'], // dependències
        '1.0.0',
        true // carregar al footer
    );

    // Passar variables PHP a JavaScript
    wp_localize_script('el-meu-tema-script', 'temaVars', [
        'ajaxUrl' => admin_url('admin-ajax.php'),
        'nonce'   => wp_create_nonce('el-meu-tema-nonce'),
    ]);
}
add_action('wp_enqueue_scripts', 'el_meu_tema_scripts');

// Registrar zona de widgets
function el_meu_tema_widgets(): void {
    register_sidebar([
        'name'          => __('Barra lateral', 'el-meu-tema'),
        'id'            => 'sidebar-1',
        'before_widget' => '<section class="widget">',
        'after_widget'  => '</section>',
        'before_title'  => '<h2 class="widget-title">',
    ]);
}

```

```

        'after_title'    => '</h2>',
    ));
}
add_action('widgets_init', 'el_meu_tema_widgets');
?>

```

header.php --- capçalera

```

<!DOCTYPE html>
<html <?php language_attributes(); ?>>
<head>
    <meta charset="<?php bloginfo('charset'); ?>">
    <meta name="viewport" content="width=device-width,
        ↪ initial-scale=1">
    <?php wp_head(); // OBLIGATORI: encola CSS, JS i metadades ?>
</head>
<body <?php body_class(); ?>>

<?php wp_body_open(); ?>

<header class="site-header">
    <div class="container">
        <a href="<?php echo esc_url(home_url('/')); ?>"
            ↪ class="site-title">
            <?php bloginfo('name'); ?>
        </a>
        <?php
        wp_nav_menu([
            'theme_location' => 'primary',
            'menu_class'     => 'nav-menu',
            'container'      => 'nav',
        ]);
        ?>
    </div>
</header>

```

footer.php --- peu

```

<footer class="site-footer">
    <div class="container">
        <p>&copy; <?php echo date('Y'); ?> <?php bloginfo('name'); ?>
        ↪ ?></p>
    </div>
</footer>

<?php wp_footer(); // OBLIGATORI: encola JavaScript del peu ?>
</body>
</html>

```

index.php --- plantilla principal (Loop de WordPress)

```
<?php get_header(); ?>

<main class="site-main">
    <div class="container">

        <?php if (have_posts()): ?>

            <?php while (have_posts()): the_post(); ?>

                <article id="post-<?php the_ID(); ?>" <?php post_class();
                ↪ ?>>
                    <header>
                        <h2><a href="<?php the_permalink(); ?>"><?php
                        ↪ the_title(); ?></a></h2>
                        <p>
                            <time datetime="<?php echo get_the_date('c');
                            ↪ ?>">
                                <?php echo get_the_date(); ?>
                            </time>
                            per <?php the_author(); ?>
                        </p>
                    </header>

                    <?php if (has_post_thumbnail()): ?>
                        <a href="<?php the_permalink(); ?>">
                            <?php the_post_thumbnail('medium'); ?>
                        </a>
                    <?php endif; ?>

                    <div class="entry-content">
                        <?php the_excerpt(); ?>
                    </div>

                    <a href="<?php the_permalink(); ?>">Llegir més →</a>
                </article>

            <?php endwhile; ?>

            <?php the_posts_navigation(); ?>

        <?php else: ?>
            <p>No s'ha trobat contingut.</p>
        <?php endif; ?>

    </div>
</main>

<?php get_sidebar(); ?>
<?php get_footer(); ?>
```

Child Theme (tema fill)

Per modificar un tema existent sense perdre els canvis en actualitzar:

```
wp-content/themes/  
├─ twentytwentyseven/ ← Tema pare per defecte de WordPress 7 (no  
  ↳ tocar)  
├─ twentytwentyseven-fill/ ← Tema fill (les nostres modificacions)  
  ├─ style.css  
  └─ functions.php
```

```
/* style.css del tema fill */  
/*  
  Theme Name:    Twenty Twenty-Seven Fill  
  Template:      twentytwentyseven  
  Version:       1.0.0  
*/
```

```
/* Sobreescriure estils del pare */  
.entry-title { color: #005f73; }
```

```
<?php  
// functions.php del tema fill  
function fill_enqueue_parent_styles(): void {  
    wp_enqueue_style(  
        'parent-style',  
        get_template_directory_uri() . '/style.css'  
    );  
}  
add_action('wp_enqueue_scripts', 'fill_enqueue_parent_styles');  
?>
```

26. Incorporació i adaptació de funcionalitats: plugins

Estructura mínima d'un plugin

```
wp-content/plugins/el-meu-plugin/  
├─ el-meu-plugin.php      ← fitxer principal (obligatori, mateix nom  
   ↳ que la carpeta)  
├─ readme.txt             ← documentació (format estàndard WP)  
├─ includes/  
│   ├── class-plugin.php  ← classe principal  
│   └─ helpers.php        ← funcions auxiliars  
└─ assets/  
    ├── css/  
    └─ js/
```

el-meu-plugin.php --- capçalera obligatòria

```
<?php  
/**  
 * Plugin Name:      El Meu Plugin  
 * Plugin URI:       https://exemple.cat/el-meu-plugin  
 * Description:      Plugin d'exemple per al mòdul 0376.  
 * Version:          1.0.0  
 * Requires at least: 7.0  
 * Requires PHP:      8.2  
 * Author:           Ramon López  
 * Author URI:       https://proferamon.com  
 * License:          GPL v2 or later  
 * Text Domain:      el-meu-plugin  
 */  
  
// Evitar accés directe al fitxer  
if (!defined('ABSPATH')) {  
    exit;  
}  
  
// Constants del plugin  
define('EMP_VERSION', '1.0.0');  
define('EMP_DIR', plugin_dir_path(__FILE__));  
define('EMP_URL', plugin_dir_url(__FILE__));  
  
// Carregar fitxers  
require_once EMP_DIR . 'includes/class-plugin.php';  
  
// Inicialitzar el plugin  
add_action('plugins_loaded', function() {  
    ElMeuPlugin::getInstance();  
});  
?>
```

Sistema de Hooks: Actions i Filters

WordPress usa un sistema de hooks per permetre que plugins i temes s'enganxin al cicle d'execució:

```
<?php
// ACTION: s'executa en un punt concret (sense valor de retorn)
// add_action($nom_hook, $funció, $prioritat, $num_args)

add_action('wp_head', function() {
    echo '<meta name="generator" content="El meu plugin 1.0">';
});

add_action('save_post', function(int $post_id, WP_Post $post) {
    if ($post->post_type !== 'post') return;
    error_log("Post desat: {$post->post_title}");
}, 10, 2); // prioritat 10, accepta 2 arguments

// FILTER: intercepta i modifica un valor
// add_filter($nom_hook, $funció, $prioritat, $num_args)

add_filter('the_title', function(string $titol): string {
    return strtoupper($titol); // tots els títols en majúscules
});

add_filter('the_content', function(string $contingut): string {
    if (!is_single()) return $contingut;
    return $contingut . '<div class="signatura">Escrit per ' .
        ↪ get_the_author() . '</div>';
});
?>
```

Custom Post Type (tipus de contingut personalitzat)

```
<?php
add_action('init', function() {
    register_post_type('producte', [
        'labels' => [
            'name' => 'Productes',
            'singular_name' => 'Producte',
            'add_new_item' => 'Afegir producte',
            'edit_item' => 'Editar producte',
        ],
        'public' => true,
        'has_archive' => true,
        'menu_icon' => 'dashicons-cart',
        'supports' => ['title', 'editor', 'thumbnail',
            ↪ 'custom-fields'],
        'rewrite' => ['slug' => 'productes'],
        'show_in_rest' => true, // necessari per a l'editor Gutenberg
    ]
);
?>
```

```
    ]);  
});  
?>
```

Custom Taxonomy (taxonomia personalitzada)

```
<?php  
add_action('init', function() {  
    register_taxonomy('categoria_producte', 'producte', [  
        'labels' => [  
            'name' => 'Categories de producte',  
            'singular_name' => 'Categoria de producte',  
        ],  
        'hierarchical' => true,    // com les categories (false =  
        ⇨ etiquetes)  
        'show_in_rest' => true,  
        'rewrite' => ['slug' => 'categoria-producte'],  
    ]);  
});  
?>
```

Pàgina d'opcions al panell d'administració

```
<?php  
// Registrar la pàgina de configuració  
add_action('admin_menu', function() {  
    add_options_page(  
        'Configuració El Meu Plugin',  
        'El Meu Plugin',  
        'manage_options', // capacitat requerida  
        'el-meu-plugin', // slug  
        'emp_renderitza_opcions'  
    );  
});  
  
// Registrar les opcions  
add_action('admin_init', function() {  
    register_setting('emp_opcions_grup', 'emp_color_accent');  
    register_setting('emp_opcions_grup', 'emp_mostrar_data',  
        ['sanitize_callback' => 'absint']); // sanititza  
});  
  
// Renderitzar la pàgina  
function emp_renderitza_opcions(): void {  
    if (!current_user_can('manage_options')) {  
        wp_die(__('No tens permís per accedir a aquesta pàgina.'));  
    }  
}  
?>
```

```

<div class="wrap">
    <h1>Configuració El Meu Plugin</h1>
    <form method="POST" action="options.php">
        <?php
            settings_fields('emp_opcions_grup');
            do_settings_sections('el-meu-plugin');
            ?>
            <table class="form-table">
                <tr>
                    <th>Color accent</th>
                    <td>
                        <input type="color" name="emp_color_accent"
                            value="<?= esc_attr(get_option('emp_col_
                                ↪ or_accent', '#005f73')) ?>">
                    </td>
                </tr>
                <tr>
                    <th>Mostrar data</th>
                    <td>
                        <input type="checkbox" name="emp_mostrar_data"
                            ↪ value="1"
                            <?php checked(1,
                                ↪ get_option('emp_mostrar_data'),
                                ↪ true); ?>
                    </td>
                </tr>
            </table>
            <?php submit_button(); ?>
        </form>
    </div>
    <?php
}
?>

```

Shortcode

```

<?php
// Registrar shortcode [contacte_caixa titol="Escriu-nos"]
add_shortcode('contacte_caixa', function(array $atribs, string
    ↪ $contingut = ''): string {
    $atribs = shortcode_atts([
        'titol' => 'Contacta\''ns',
        'color' => '#005f73',
    ], $atribs, 'contacte_caixa');

    ob_start();
    ?>
    <div class="caixa-contacte" style="border-color: <?=
        ↪ esc_attr($atribs['color']) ?>">
        <h3><?= esc_html($atribs['titol']) ?></h3>

```

```

        <?= wpkses_post($contingut) ?>
        <a href="<?=
            ↪ esc_url(get_permalink(get_page_by_path('contacte')))) ?>">
            Anar al formulari ↪
        </a>
    </div>
    <?php
        return ob_get_clean();
    });
    ?>

```

Funcions de seguretat de WordPress

```

<?php
// Escapament de sortida (sempre usar la funció adequada al context)
esc_html($text);           // text pla dins HTML
esc_attr($valor);          // dins d'un atribut HTML (value="...")
esc_url($url);             // URLs (href, src)
esc_js($valor);            // dins de JavaScript
wp_kses_post($html);       // HTML restringit (permet etiquetes
    ↪ segures)
absint($numero);           // enter positiu o zero

// Comprovació de nonce (protecció CSRF)
// Generar
wp_nonce_field('accio_meva', 'nonce_field'); // camp ocult al
    ↪ formulari
wp_create_nonce('accio_meva');                // string del nonce

// Verificar
check_admin_referer('accio_meva', 'nonce_field'); // atura si invàlid
if (!wp_verify_nonce($_POST['nonce'], 'accio_ajax')) {
    wp_die('Petició no autoritzada');
}

// Comprovar permisos
if (!current_user_can('manage_options')) {
    wp_die('Accés denegat');
}
if (!current_user_can('edit_post', $post_id)) {
    wp_die('No pots editar aquest post');
}

// Sanitització d'entrada
sanitize_text_field($_POST['nom']); // text pla
sanitize_email($_POST['email']);    // email
sanitize_url($_POST['url']);        // URL
absint($_POST['quantitat']);        // enter positiu
wp_strip_all_tags($_POST['contingut']); // elimina etiquetes HTML
?>

```

27. Verificació i documentació

Proves de funcionament

```
<?php
// Activar WP_DEBUG per veure errors durant el desenvolupament
// wp-config.php
define('WP_DEBUG', true);
define('WP_DEBUG_LOG', true);
define('WP_DEBUG_DISPLAY', false); // false en producció

// Comprovar errors de PHP al log
// Ubicació: wp-content/debug.log
?>
```

Llista de verificació per a un tema o plugin:

- ☐ El tema s'activa sense errors al panell
- ☐ Totes les pàgines renderitzen correctament (inici, entrada, pàgina, arxiu, 404)
- ☐ El menú de navegació funciona
- ☐ La imatge destacada es mostra
- ☐ El formulari de cerca funciona
- ☐ El tema és responsiu (mòbil, tauleta, escriptori)
- ☐ S'ha provat en els navegadors principals (Chrome, Firefox, Safari)
- ☐ El plugin s'activa i desactiva sense errors
- ☐ Les opcions es desen correctament
- ☐ Els shortcodes renderitzen l'HTML esperat
- ☐ No hi ha errors PHP ni warnings al log
- ☐ S'ha passat el plugin pel Plugin Check de WordPress.org

Eines de verificació

```
# WP-CLI: interfície de línia d'ordres per a WordPress
wp plugin list                # llistar plugins
wp plugin activate el-meu-plugin # activar
wp theme check el-meu-tema     # comprovar el tema

# Theme Check plugin (panell WordPress)
# Verifica que el tema compleix els estàndards

# Query Monitor plugin
# Analitza les consultes a la BD, hooks usats i temps de càrrega
```

Documentació

Una bona documentació inclou:

Al codi:

```
<?php
/**
 * Registra un nou usuari al sistema.
 *
 * @since 1.0.0
 * @param string $nom    Nom complet de l'usuari
 * @param string $email  Adreça de correu electrònic
 * @param string $clau   Contrasenya en text pla (es xifrarà
 *                       ↪ internament)
 * @return int|WP_Error  ID del nou usuari o WP_Error en cas d'error
 */
function registra_usuari(string $nom, string $email, string $clau):
    ↪ int|WP_Error {
    // ...
}
?>
```

Fitxer readme.txt (format WordPress):

```
=== El Meu Plugin ===
Contributors: ramonlopez
Tags: exemple, asix
Requires at least: 7.0
Tested up to: 7.0
Requires PHP: 8.2
Stable tag: 1.0.0
License: GPLv2 or later

== Descripció ==
Plugin d'exemple creat per al mòdul d'Implantació d'aplicacions web.

== Instal·lació ==
1. Puja el directori `el-meu-plugin` a `/wp-content/plugins/`
2. Activa el plugin des del panell d'administració

== Registre de canvis ==
= 1.0.0 =
* Primera versió
```

28. Referència de sintaxi i funcions clau

Sintaxi PHP essencial

```
<?php
// Variables i tipus
$var = valor;          var_dump($var); print_r($var);
(int) $v; (float) $v; (string) $v; (bool) $v;

// Condicionals
if ($c): elseif ($c2): else: endif;
match($v) { cas => resultat, default => ... };

// Bucles
while ($c) { }
do { } while ($c);
for ($i=0; $i<$n; $i++) { }
foreach ($arr as $k => $v) { }

// Funcions
function nom(tipus $param = defecte): tipusRetorn { return $val; }
fn($n) => expressió; // funció fletxa

// Classes
class Nom extends Pare implements Interfície {
    public function __construct(private int $val) {}
    public function mètode(): tipus { return $this->val; }
    public static function estàtic(): void { self::$prop; }
}

// Errors
try { } catch (TipusException $e) { } finally { }
throw new Exception("missatge");
?>
```

Funcions PHP principals per àrea

Àrea	Funcions
Strings	strlen, mb_strlen, strpos, substr, str_replace, trim, strtolower, strtoupper, explode, implode, sprintf, htmlspecialchars, nl2br, strip_tags
Nombre	abs, ceil, floor, round, sqrt, pow, min, max, rand, random_int, number_format, is_numeric
Arrays	count, in_array, array_key_exists, array_push, array_pop, array_merge, array_slice, array_unique, array_filter, array_map, array_reduce, sort, usort, array_search, array_combine
Fitxers	file_get_contents, file_put_contents, fopen, fread, fwrite, fclose, file_exists, filesize, basename, pathinfo, scandir, is_dir, mkdir
BD (PDO)	new PDO(), prepare, execute, fetch, fetchAll, fetchColumn, rowCount, lastInsertId, beginTransaction, commit, rollBack
Seguretat	htmlspecialchars, filter_var, password_hash, password_verify, hash_equals, random_bytes, bin2hex
Sessió	session_start, session_regenerate_id, session_destroy, \$_SESSION
Data	date, time, strtotime, new DateTime, DateInterval, date_default_timezone_set

JSON	<code>json_encode</code> , <code>json_decode</code> , <code>json_last_error</code>
Debug	<code>var_dump</code> , <code>print_r</code> , <code>get_defined_vars</code> , <code>debug_backtrace</code> , <code>error_log</code>

Bones pràctiques generals

Àmbit	Pràctica
Codi	<code>declare(strict_types=1)</code> · Tipat de paràmetres i retorn · PHPDoc
Noms	Classes: StudlyCaps · Mètodes/variables: camelCase · Constants: MAJÚSCULES
Fixers	UTF-8 sense BOM · Una classe per fixer · Ometre <code>?></code> en fixers 100% PHP
Seguretat	Sempre prepared statements · <code>htmlspecialchars()</code> en sortides · <code>password_hash()</code>
Errors	<code>display_errors=Off</code> en producció · <code>log_errors=On</code> · Capturar excepcions
Rendiment	OPcache activat · Seleccionar columnes concretes · Paginació · Transaccions
WordPress	Sempre <code>esc_*()</code> per a sortides · Nonce per a formularis · <code>wp_enqueue_*()</code> per a assets

Índex de figures

1	El servidor executa el PHP abans de retornar l'HTML que el client renderitza	1
---	--	---

Versions d'aquest document

- [HTML](#)
- [PDF](#)
- [ODT](#)
- [MD](#)

[Domini Públic \(CC0\)](#)