

Lucas Vázquez Entorno Servidor

WWW



Teoria (Resumen)

Conjunto de recursos interconectados que conforman el conocimiento humano actual

(Hubs, repetidores, puentes)

(TCP,IP,HTTPS,FTP,SMTP)

(DNS)

Cliente servidor, considerada la arquitectura mas habitual:

Cliente: es un componente consumidor de servicios

Servidor: es un proceso proveedor de servicios

CAPAS:

Capa de Presentación: La que visualiza el usuario

Capa de Negocio: Gestiona las funcionalidades que esperamos del sistema o aplicacion web

Capa de Datos: Donde se guardan los datos y encargada de acceder dentro

Classificacion de modelos de programacion web:

Segun la medida de los componentes:

FAT Client: El mayor peso se ejecuta en el cliente

FAT Server: Funcionalidad asociada al cliente limitada a la presentacion de la informacion enviada para el servidor

Segun la naturaleza del servicio ofrecido: Servidores de ficheros, servidores de BDD, servidores web

Clasificacion de las aplicaciones web:

Aplicaciones web estáticas: Usuario recibe pagina y la interaccion de la pagina no comporta ninguna acción.

Aplicaciones web dinámicas: interaccion del cliente con el recurso recibido por parte del servidor (gmail)

Aplicaciones web interactivas: Dialogo entre cliente servidor

Lenguajes de Programación:

Es el codigo del bien sea como un objeto procompilado o bien como un codigo interpretado, es ejecutado por un programa especifico en el componente que actua como servidor.

Tipos:

- Lenguajes de Scripting: La interacción del usuario hace que se genere un dialogo entre el cliente y el servidor, EJ:

PHP, ASP, Perl, Python, JSP

- Aplicaciones CGI: Delegan la creación de paginas dinamicas a un programa externo (RFC: Unifican criterios en muchas de los ambitos de telecomunicaciones)// Estandar CGI// El programa externo no depende del codigo a generar//

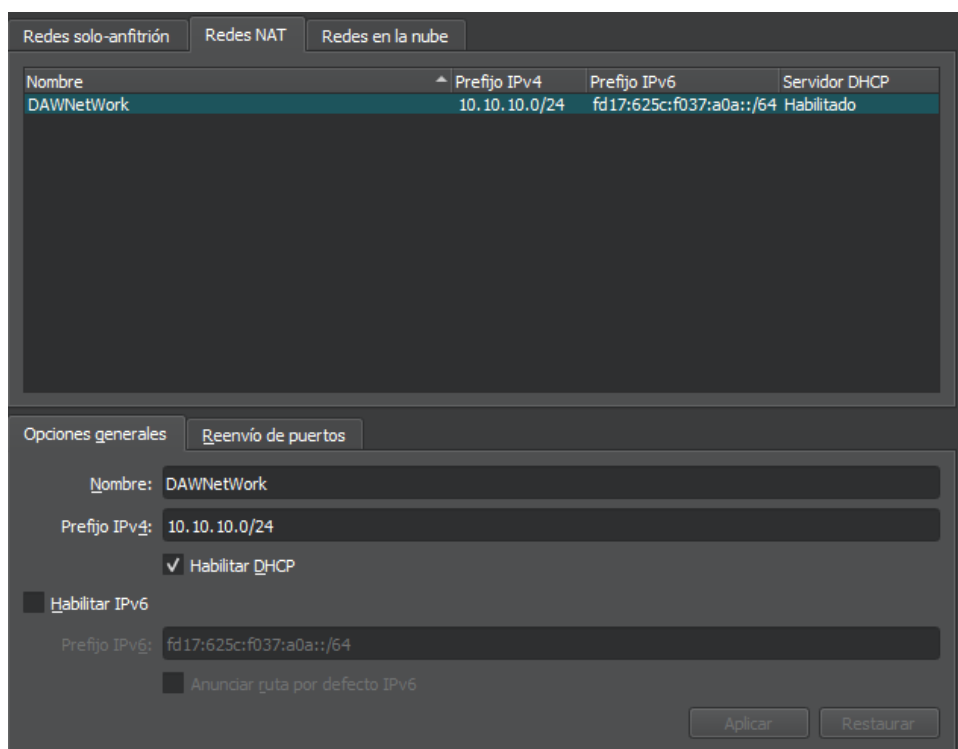
Desventajas: Escaso rendimiento a la hora de responder a multiples peticiones CGI simultaneamente

- Aplicaciones Híbridas de código repartido: Tecnología intermedia alternativa a los lenguajes de scripting y a las aplicaciones CGI y derivados

El proceso de desenvolvamiento no tiene porque realizarse en el mismo equipo en el qual finalmente se despliegue y ejecute la aplicacion web que se esta desenvolvando// Instrumentos: Navegadores, Editores de Documentos, Entornos de programación, BDD

Instalacion y Configuracion del Servidor

Usamos la IP 10.10.10.0/24 para que el servidor y el cliente tengan visibilidad entre ellas, la ip del servidor es 10.10.10.4/24.



Una vez teniendo todo el servidor preparado, procederíamos a instalar PHP.

Usando este comando **sudo apt install php libapache2-mod-php**, aparte de instalar el lenguaje PHP, tambien instala el modulo que permite a apache, que en este caso seria nuestro servidor web, poder procesar lenguaje PHP.

Después instalariamos paquetes opcionales, pero que utilizaremos mas adelante, asi como:

```
sudo apt install php-cli
sudo apt install php-cgi
sudo apt install php-mysql
sudo apt install php-pgsql
```

Ahora tocaria configurar PHP

Primero tendriamos que crear un fichero php en esta direccion /var/www/html:

```
alumnedaw@ubuntuserverdawcesur:/var/www/html$ ls
holamundo.php  index.html  phpinfo.php
alumnedaw@ubuntuserverdawcesur:/var/www/html$
```

Aqui cree el fichero phpinfo.php, que dentro contiene

```
GNU nano 7.2
<?php
phpinfo();
?>
```

Tambien creamos un fichero php donde pudimos ver que podemos utilizar lenguaje html, usando prints en cada linea de codigo.

```
GNU nano 7.2
<?php
print("<!DOCTYPE html>");
print("<html lang=en>");
print("<script>>window.alert('Hola Mundo')</script>");
    print("<h1>Hola Mundo</h1>");
print("</html>");
?>
```


Puesto en practica ya en el entorno cliente, usando la ip del servidor, en este caso 10.10.10.4 y escribiendo el nombre del fichero que queramos (10.10.10.4/phpinfo.php), comprobariamos que esta todo funcionando

Oct 1 09:09

¡Gracias por descargar We x PHP 8.3.6 - phpinfo()


10.10.10.4/phpinfo.php80%

PHP Version 8.3.6

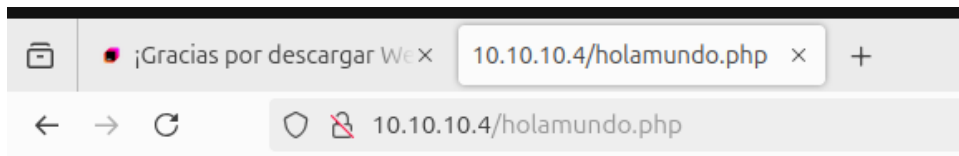


| | |
|---|--|
| System | Linux ubuntu:server:24.04-generic 6.8.0-45-generic #45-Ubuntu SMP PREEMPT_DYNAMIC Fri Aug 30 12:02:04 UTC 2024 x86_64 |
| Build Date | Jun 13 2024 15:23:20 |
| Build System | Linux |
| Server API | Apache 2.0 Handler |
| Virtual Directory Support | disabled |
| Configuration File (php.ini) Path | /etc/php/8.3/apache2 |
| Loaded Configuration File | /etc/php/8.3/apache2/php.ini |
| Scan this dir for additional .ini files | /etc/php/8.3/apache2/conf.d |
| Additional .ini files parsed | /etc/php/8.3/apache2/conf.d/10-mysqld.ini, /etc/php/8.3/apache2/conf.d/10-opcache.ini, /etc/php/8.3/apache2/conf.d/10-pdo.ini, /etc/php/8.3/apache2/conf.d/20-calendar.ini, /etc/php/8.3/apache2/conf.d/20-ctype.ini, /etc/php/8.3/apache2/conf.d/20-exif.ini, /etc/php/8.3/apache2/conf.d/20-fileinfo.ini, /etc/php/8.3/apache2/conf.d/20-ftp.ini, /etc/php/8.3/apache2/conf.d/20-gettext.ini, /etc/php/8.3/apache2/conf.d/20-iconv.ini, /etc/php/8.3/apache2/conf.d/20-mysqli.ini, /etc/php/8.3/apache2/conf.d/20-pdo_mysql.ini, /etc/php/8.3/apache2/conf.d/20-pdo_pgsql.ini, /etc/php/8.3/apache2/conf.d/20-pgsql.ini, /etc/php/8.3/apache2/conf.d/20-phar.ini, /etc/php/8.3/apache2/conf.d/20-posix.ini, /etc/php/8.3/apache2/conf.d/20-readline.ini, /etc/php/8.3/apache2/conf.d/20-shmop.ini, /etc/php/8.3/apache2/conf.d/20-sockets.ini, /etc/php/8.3/apache2/conf.d/20-sysmsg.ini, /etc/php/8.3/apache2/conf.d/20-sysvsem.ini, /etc/php/8.3/apache2/conf.d/20-sysvshm.ini, /etc/php/8.3/apache2/conf.d/20-tokenizer.ini |
| PHP API | 20230831 |
| PHP Extension | 20230831 |
| Zend Extension | 420230831 |
| Zend Extension Build | API420230831.NTS |
| PHP Extension Build | API20230831.NTS |
| Debug Build | no |
| Thread Safety | disabled |
| Zend Signal Handling | enabled |
| Zend Memory Manager | enabled |
| Zend Multibyte Support | disabled |
| Zend Max Execution Timers | disabled |
| IPv6 Support | enabled |
| DTrace Support | disabled |
| Registered PHP Streams | https, ftps, compress.zlib, php, file, glob, data, http, ftp, phar |
| Registered Stream Socket Transports | tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3 |
| Registered Stream Filters | zlib.*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, convert.iconv.* |

This program makes use of the Zend Scripting Language Engine:
Zend Engine v4.3.6, Copyright (c) Zend Technologies



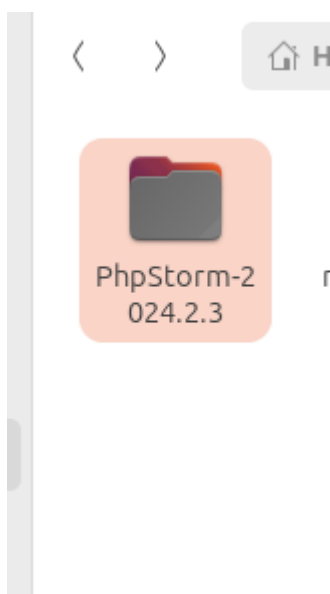
El otro fichero.



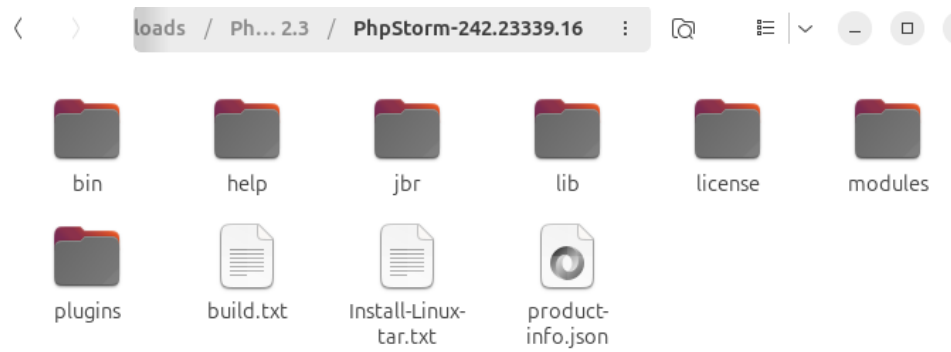
Hola Mundo

Instalacion de PhpStorm en el cliente.

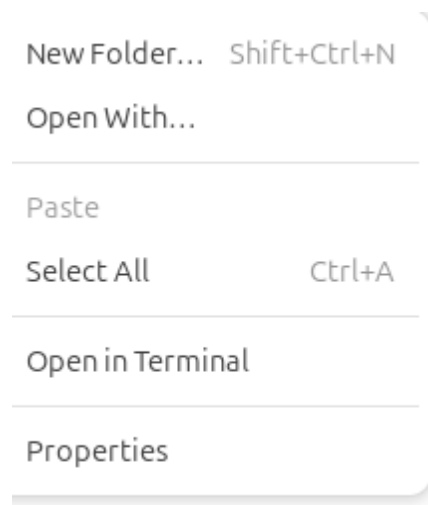
Primero simplemente nos bajamos el instalador desde el navegador



Entramos en bin



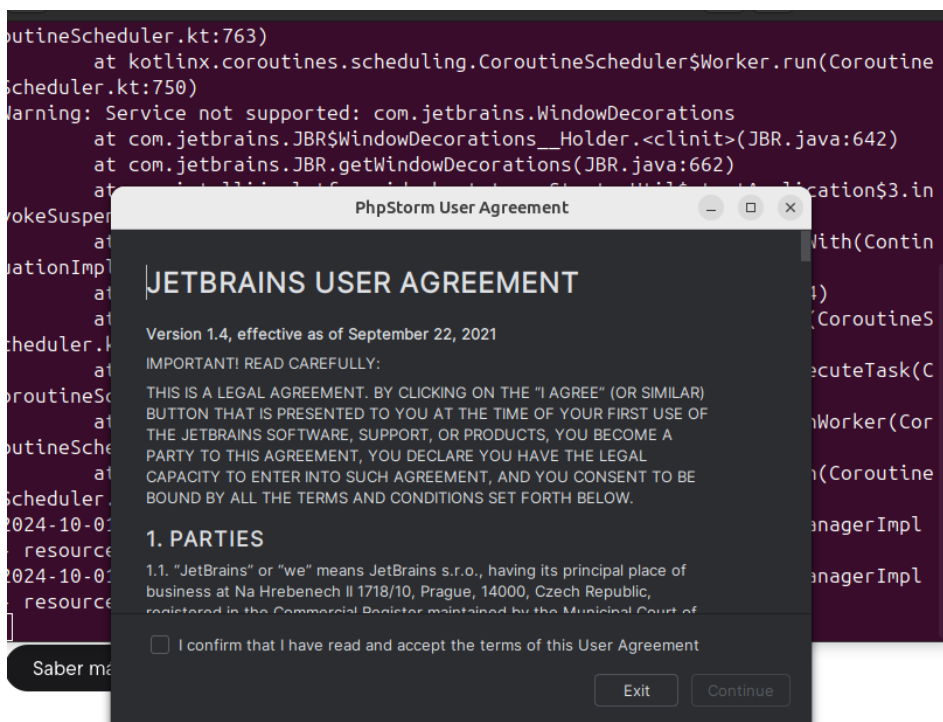
Hacemos click derecho y abrimos la terminal, para que estemos dentro ya del fichero de PhpStorm



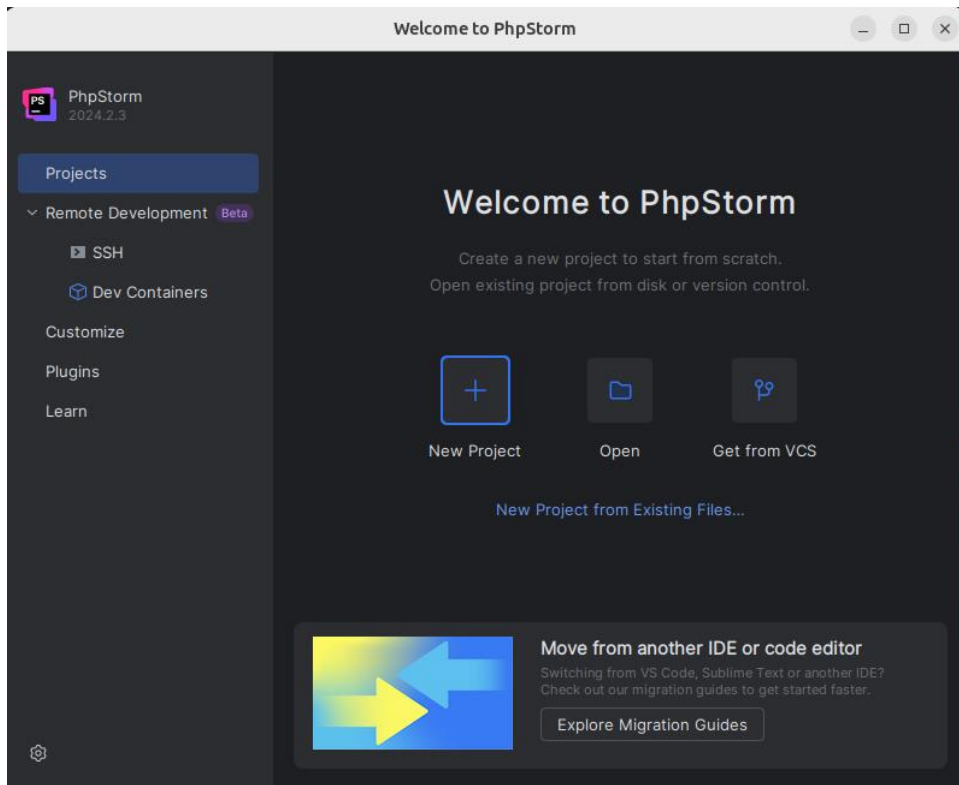
Escribimos `./phpstorm.sh`, para que empiece la instalacion

```
alumnedaw@alumnedaw-VirtualBox: ~/Downloads/PhpStorm-2024.2.3/PhpStorm-242.23339
6/bin$ ./phpstorm.sh
```

Aquí ya esta preparado para instalarse



Lo instalamos sin problema y activamos nuestra licencia



Instalar GIT

Usamos **sudo apt install GIT**

```
alumedaw@alumedaw-VirtualBox:~/Downloads/PhpStorm-2024.2.3/PhpStorm-242.2333  
16/bin$ git -v  
git version 2.43.0
```

Act Clase Analizar Códigos:

Analitzau aquests codis. A continuació comproveu el seu funcionament I expliqueu (a la vostra CA) el funcionament de tot plegat amb només de 5 línies (doc 2,5 cms marge esq I dreta, lletra arial 12).

```

1 <form action='login.php' method='post'>
2   <fieldset>
3     <legend>Login</legend>
4     <div><span class='error'><?php echo $error; ?></span></div>
5     <div class='fila'>
6       <label for='usuario'>Usuario:</label><br />
7       <input type='text' name='inputUsuario' id='usuario' maxlength="50" /><br />
8     </div>
9     <div class='fila'>
10      <label for='password'>Contraseña:</label><br />
11      <input type='password' name='inputPassword' id='password' maxlength="50" /><br />
12    </div>
13    <div class='fila'>
14      <input type='submit' name='enviar' value='Enviar' />
15    </div>
16  </fieldset>
17 </form>

```

Este código es un HTML que representa un formulario que está conectado con login.php, usando fieldset para agrupar los datos del formulario. La línea php representa por si hay algún error. Luego usamos div para ir agrupando lo que nos interesa dándole clases. El label es para pintar por máquina Usuario:, y el input de modo text con un máximo de 50 caracteres para que escriba su usuario, luego el otro input usamos submit para que le llegue al php.

```

1 <?php
2 // Comprobamos si ya se ha enviado el formulario
3 if (isset($_POST['enviar'])) {
4   $usuario = $_POST['inputUsuario'];
5   $password = $_POST['inputPassword'];
6
7   // validamos que recibimos ambos parámetros
8   if (empty($usuario) || empty($password)) {
9     $error = "Debes introducir un usuario y contraseña";
10    include "index.php";
11  } else {
12    if ($usuario == "admin" && $password == "admin") {
13      // almacenamos el usuario en la sesión
14      session_start();
15      $_SESSION['usuario'] = $usuario;
16      // cargamos la página principal
17      include "main.php";
18    } else {
19      // Si las credenciales no son válidas, se vuelven a pedir
20      $error = "Usuario o contraseña no válidos!";
21      include "index.php";
22    }
23  }
24 }

```

Lo que hace la primera línea es comprobar si desde el form se ha enviado, creamos 2 variables para usuario y contraseña, que las cogemos usando POST. Si Usuario o Contraseña están vacíos damos un error, si no, si el usuario y contraseña tienen el valor admin, usamos session_start para almacenar la sesión, y devolvemos la página principal que sería main.php, si no fuera el caso damos un error enviándolo al índice.

```

1  <?php
2      // Recuperamos la información de la sesión
3      if(!isset($_SESSION)) {
4          session_start();
5      }
6
7      // Y comprobamos que el usuario se haya autenticado
8      if (!isset($_SESSION['usuario'])) {
9          die("Error - debe <a href='index.php'>identificarse</a>.<br />");
10     }
11 >?
12 <!DOCTYPE html>
13 <html lang="es">
14 <head>
15     <meta charset="UTF-8">
16     <meta name="viewport" content="width=device-width, initial-scale=1.0">
17     <title>Listado de productos</title>
18 </head>
19 <body>
20     <h1>Bienvenido <?= $_SESSION['usuario'] ?></h1>
21     <p>Pulse <a href="logout.php">aquí</a> para salir</p>
22     <p>Volver al <a href="main.php">inicio</a></p>
23     <h2>Listado de productos</h2>
24     <ul>
25         <li>Producto 1</li>
26         <li>Producto 2</li>
27         <li>Producto 3</li>
28     </ul>
29 </body>
30 </html>

```

Lo primero miramos para devolver la información de la sesión, si el usuario no está registrado, devolvemos error devolviéndole a index.php. Creamos un html dándole la bienvenida al usuario , y un a href por si quiere volver a logout.php o al inicio. Luego creamos una lista de los productos.

```

1  <?php
2      // Recuperamos la información de la sesión
3      session_start();
4
5      // Y la destruimos
6      session_destroy();
7      header("Location: index.php");
8  ?>

```

Este Código sirve para borrar las sesiones

OBJETOS PHP:

Php sigue una estructura de objetos basada en clases. Existen diferentes tipos de clases:

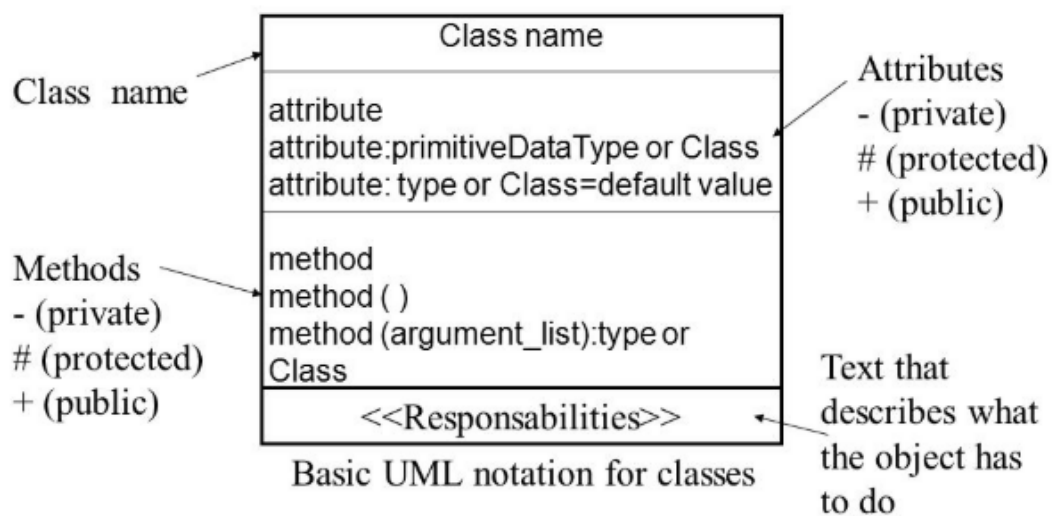
Privadas: solo puede acceder la propia clase

Publicas: Puede acceder cualquier clase

Protegidas: Solo puede acceder la clase o sus descendientes

Un ejemplo para crear una clase seria:

Class HolaClase {



Esto es una estructura basica para las clases, con sus atributos y metodos.

Para acceder a una propiedad de la misma clase se puede usar this.

```
$this->propiedad;
```

Tenemos Get y Set (metodos publicos):

Get lo que hace es permitir el obtener el valor de una propiedad de la clase y así poder utilizar dicho valor en diferentes métodos

Lo que hace Set es permitir introducir valores

Aqui tengo un ejemplo practico:

Funciones:

<?php

```
class Producte {  
    public $nom;  
    public $descripcio;  
    public $preu;  
    public $productos = [];  
  
    public function __constructor($nom, $descripcio, $preu)  
    {  
        $this->nom = $nom;  
        $this->descripcio = $descripcio;  
        $this->preu = $preu;  
    }  
}  
  
class Categoria {  
    public $nomCategoria;  
    public $descripcioCategoria;  
    public $categorias = [];
```

```
public function __constructor($nomCategoria, $desccipcioCategoria) {

    $this->nomCategoria = $nomCategoria;
    $this->descripcioCategoria = $desccipcioCategoria;

}
}

function crearProducte($nom, $descripcio, $preu, $productos) {

    $producte = new Producte($nom, $descripcio, $preu);
    $productos[] = $producte;
    return $producte;

}

function crearCategoria($nom, $descripcio, $categorias) {

    $categoria = new Categoria($nom, $descripcio);
    $categorias[] = $categoria;
    return $categoria;

}

function agregarCategoriaAProducte(Producte $producte, Categoria
$categoria) {
```

```
$producte->categoria = $categoria;
```

```
}
```

```
function obtenerProductsPorCategoria(Categoria $categoria) {
```

```
}
```

```
function mostrarProductos(array $productos) {
```

```
    echo "Nombre: " . $productos->nom . "\n";
```

```
    echo "Descripción: " . $productos->descripcio . "\n";
```

```
    echo "Precio: " . $productos->preu . "\n";
```

```
}
```

El índice:

```
<?php
```

```
require_once 'funcions.php';
```

```
class index {
```

```
    public function __constructor() {
```

```
        $producte1 = crearProducte('Samarreta', 'Samarreta de cotó', 19.99);
```

```
$producte2 = crearProducte('Pantalons', 'Pantalons vaquer', 39.99);

$categoria1 = crearCategoria('Roba', 'Secció de roba');

$categoria2 = crearCategoria('Home', 'Productes per a home');

agregarCategoriaAProducte($producte1, $categoria1);

agregarCategoriaAProducte($producte1, $categoria2);

agregarCategoriaAProducte($producte2, $categoria1);


$productesRoba = obtenirProductesPorCategoria($categoria1);

mostrarProductes($productesRoba);
}
}
```

Decodificar con JSON y insertar en una BDD:


```

<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $dades = json_decode(json: $_POST['dades'], associative: true);
    echo htmlspecialchars(string: $dades['nom']);
    echo htmlspecialchars(string: $dades['edat']);
    echo htmlspecialchars(string: $dades['dni']);
} else {
    echo "No dadaes";
}
?>

```

Decodificar (json_decode)

Insertar en mis base de datos:

```

<?php
$servername = "sql306.infinityfree.com";
$username = "if0_37670035";
$password = "La3F9lU0xuq0Sy";
$dbname = "if0_37670035_cvlucas";

$conectar = new mysqli($servername, $username, $password, $dbname);

if ($conectar->connect_error) {
    die("Conexion fallida: " . $conectar->connect_error);
}

$nom = "tofol";
$edat = 2;
$dni = "no en te";

$sql = "INSERT INTO Datos (nom, edat, dni) VALUES ('$nom', '$edat', '$dni')";

if ($conectar->query($sql) === TRUE) {
    echo "Inserción con éxito!";
} else {
    echo "Error en la insercción";
}
$conectar->close();
?>

```

Pongo mis datos de la bdd la conecto mediante msqli, si hay algun error cierro la conexion. Guardo los datos decodificados anteriormente en unas variables, en una variable guardo lo que es la insercion de estos datos. Si todo funciona doy un mensaje de exito y sino de error, cierro la conexion

Codigo de resolver una Ecuacion:

```
<?php
function ecuacion($a, $b) {
    if ($a == 0) {
        if ($b == 0) {
            return "La ecuacion tiene infinitas soluciones.";
        } else {
            return "La ecuacion no tiene solucion.";
        }
    } else {
        return -$b / $a;
    }
}

$a = 2;
$b = -4;
$solucion = ecuacion($a, $b);
print("La solución de la ecuacion es: " . $solucion);
?>
```

Codigo de resolver un volumen:

```
<?php
function volumenCono($radio, $altura) {
    $pi = 3.1415;
    return ($pi * pow($radio, 2) * $altura) / 3;
}

$radio = 5;
$altura = 10;
$volumen = volumenCono($radio, $altura);
```

```
echo("El volumen del cono és: " . $volumen);  
  
?>
```

Aqui tengo un formulario que dependiendo que elija devuelve si esta mal o bien la respuesta:

```
<!DOCTYPE html>  
<html lang="ca">  
<head>  
  <meta charset="UTF-8">  
  <title>Resultat del Qüestionari</title>  
</head>  
<body>  
  <fieldset>  
    <h1>Resultat del Qüestionari</h1>  
    <form method="post" >  
      <p>1. Qual no es un Lenguaje de Programacion?</p>  
      <input type="radio" name="question0" value="a"> HTML<br>  
      <input type="radio" name="question0" value="b"> Java<br>  
      <input type="radio" name="question0" value="c"> PHP<br>  
      <input type="radio" name="question0" value="d"> JavaScript<br>  
  
      <p>2. ¿Cuál es el lenguaje de programación más utilizado para  
desarrollo web?</p>  
      <input type="radio" name="question1" value="a"> Python<br>  
      <input type="radio" name="question1" value="b"> JavaScript<br>  
      <input type="radio" name="question1" value="c"> C++<br>  
      <input type="radio" name="question1" value="d"> Ruby<br>  
  
      <p>3. ¿Que significa CSS?</p>  
      <input type="radio" name="question2" value="a"> Cascading Style  
Sheets<br>  
      <input type="radio" name="question2" value="b"> Computer Style  
Sheets<br>  
      <input type="radio" name="question2" value="c"> Creative Style  
Sheets<br>  
      <input type="radio" name="question2" value="d"> Colorful Style  
Sheets<br>  
  
      <p>4. ¿Cuál es la funcion principal de una BDD?</p>
```

```
<input type="radio" name="question3" value="a"> Diseñar interfaces
web<br>
<input type="radio" name="question3" value="b"> Crear paginas web<br>
<input type="radio" name="question3" value="c"> Procesar Datos<br>
<input type="radio" name="question3" value="d"> Almacenar Datos<br>

<p>5. ¿Cuál es un sistema de gestion en BDD?</p>
<input type="radio" name="question4" value="a"> MYSQL<br>
<input type="radio" name="question4" value="b"> JavaScript<br>
<input type="radio" name="question4" value="c"> HTML<br>
<input type="radio" name="question4" value="d"> Apache<br>

<p>6. ¿Que es un servidor web?</p>
<input type="radio" name="question5" value="a"> Python<br>
<input type="radio" name="question5" value="b"> JavaScript<br>
<input type="radio" name="question5" value="c"> Un software que sirve
para paginas web a los usuarios<br>
<input type="radio" name="question5" value="d"> Ruby<br>

<p>7. ¿Que es Apache2?</p>
<input type="radio" name="question6" value="a"> Python<br>
<input type="radio" name="question6" value="b"> JavaScript<br>
<input type="radio" name="question6" value="c"> Es el servidor web que
mas se usa en el mundo<br>
<input type="radio" name="question6" value="d"> Ruby<br>

<p>8. ¿Para que se usa CSS?</p>
<input type="radio" name="question7" value="a"> Python<br>
<input type="radio" name="question7" value="b"> JavaScript<br>
<input type="radio" name="question7" value="c"> Para decorar una
pagina web<br>
<input type="radio" name="question7" value="d"> Ruby<br>

<p>9. ¿Que significa IA?</p>
<input type="radio" name="question8" value="a"> Inteligencia
Artificial<br>
<input type="radio" name="question8" value="b"> JavaScript<br>
<input type="radio" name="question8" value="c"> C++<br>
<input type="radio" name="question8" value="d"> Ruby<br>

<p>10. ¿Como me llamo?</p>
<input type="radio" name="question9" value="a"> Python<br>
<input type="radio" name="question9" value="b"> Lucas<br>
<input type="radio" name="question9" value="c"> C++<br>
<input type="radio" name="question9" value="d"> Ruby<br>
```

```
        <input type="submit" value="Enviar">

    </form>
</fieldset>
</body>
</html>
```

Este seria el html sencillo

Y aqui estaria el php:

```
<?php

if ($_POST) {
    $r1 = $_POST["r1"];
    $r2 = $_POST["r2"];
    $r3 = $_POST["r3"];
    $r4 = $_POST["r4"];
    $r5 = $_POST["r5"];
    $r6 = $_POST["r6"];
    $r7 = $_POST["r7"];
    $r8 = $_POST["r8"];
    $r9 = $_POST["r9"];
    $r10 = $_POST["r10"];
    $c = 0;
    $i = 0;

    if($r1=="a") {
        print"Pregunta 1 Correcta";
        $c+=1;
    } else {
        print"Pregunta 1 Mal";
    }
    if($r2=="b") {
        print"Pregunta 2 Correcta";
        $c+=1;
    } else {
        print"Pregunta 2 Mal";
    }
    if($r3=="a") {
        print"Pregunta 3 Correcta";
        $c+=1;
    } else {
```

```
        print"Pregunta 3 Mal";
    }
    if($r4=="d") {
        print"Pregunta 4 Correcta";
        $c+=1;
    } else {
        print"Pregunta 4 Mal";
    }
    if($r5=="a") {
        print"Pregunta 5 Correcta";
        $c+=1;
    } else {
        print"Pregunta 5 Mal";
    }
    if($r6=="c") {
        print"Pregunta 6 Correcta";
        $c+=1;
    } else {
        print"Pregunta 6 Mal";
    }
    if($r7=="c") {
        print"Pregunta 7 Correcta";
        $c+=1;
    } else {
        print"Pregunta 7 Mal";
    }
    if($r8=="c") {
        print"Pregunta 1 Correcta";
        $c+=1;
    } else {
        print"Pregunta 8 Mal";
    }
    if($r9=="a") {
        print"Pregunta 9 Correcta";
        $c+=1;
    } else {
        print"Pregunta 9 Mal";
    }
    if($r10=="b") {
        print"Pregunta 10 Correcta";
        $c+=1;
    } else {
        print"Pregunta 10 Mal";
    }
}
?>
```

Cojemos las preguntas y las guardamos en variables, hacemos si r1 por ejemplos es igual a “a”, pues se la damos por correcta sino mal, además hay un contador que suma las respuestas correctas para dar una nota

Aqui hay algunos ejercicios sobre COOKIES:

```
<?php
$cookieName = "comptador_visites";

if (isset($_GET['reinicialitzar'])) {

    setcookie($cookieName, 0, time() + (86400 * 30), "/");
    $missatge = "El teu comptador de visites ha estat reinicialitzat.";
    $visites = 0;
} else {

    if (isset($_COOKIE[$cookieName])) {

        $visites = $_COOKIE[$cookieName] + 1;
    } else {

        $visites = 1;
    }

    setcookie($cookieName, $visites, time() + (86400 * 30), "/");
    $missatge = "Has visitat aquesta pàgina $visites vegades.";
}
?>
<!DOCTYPE html>
<html lang="ca">
<head>
    <meta charset="UTF-8">
    <title>Comptador de Visites</title>
</head>
<body>

<h1><?php echo $missatge; ?></h1>
<form method="get" action="">
```

```

        <button type="submit" name="reinicialitzar" value="1">Reinicialitzar
Comptador</button>
</form>
</body>
</html>

```

```

<!DOCTYPE html>
<html lang="ca">
<head>
    <meta charset="UTF-8">
    <title>Selector de Color de Fons</title>
</head>
<body>
<?php
$cookieName = "color_fons";
$colorSeleccionat = "#FFFFFF";

if (isset($_POST['color'])) {
    $colorSeleccionat = $_POST['color'];

    setcookie($cookieName, $colorSeleccionat, time() + 86400, "/");
} elseif (isset($_COOKIE[$cookieName])) {

    $colorSeleccionat = $_COOKIE[$cookieName];
}
?>

<body style="background-color: <?php echo $colorSeleccionat; ?>;">

<h1>Selecciona un color de fons</h1>
<form method="post" action="">
    <select name="color" onchange="this.form.submit()">
        <option value="#FFFFFF" <?php if($colorSeleccionat == "#FFFFFF") echo
'selected'; ?>>Blanc</option>
        <option value="#FF0000" <?php if($colorSeleccionat == "#FF0000") echo
'selected'; ?>>Vermell</option>
        <option value="#00FF00" <?php if($colorSeleccionat == "#00FF00") echo
'selected'; ?>>Verd</option>
        <option value="#0000FF" <?php if($colorSeleccionat == "#0000FF") echo
'selected'; ?>>Blau</option>
        <option value="#FFFF00" <?php if($colorSeleccionat == "#FFFF00") echo
'selected'; ?>>Groc</option>

```



```

        <option value="#FF00FF" <?php if($colorSeleccionat == "#FF00FF") echo
'selected'; ?>>Magenta</option>
        <option value="#00FFFF" <?php if($colorSeleccionat == "#00FFFF") echo
'selected'; ?>>Cian</option>
    </select>
</form>
</body>
</html>

```

```

<!DOCTYPE html>
<html lang="ca">
<head>
    <meta charset="UTF-8">
    <title>Formulari 1</title>
</head>
<body>
<h1>Formulari 1: Dades Bàsiques</h1>
<form method="post" action="exercici127formulari2.php">
    <label for="nom">Nom i Cognoms:</label>
    <input type="text" id="nom" name="nom" placeholder="Escriu el teu nom"
required><br>

    <label for="email">Email:</label>
    <input type="email" id="email" name="email" placeholder="Escriu el teu
email " required><br>

    <label for="url">URL:</label>
    <input type="url" id="url" name="url" placeholder="Escriu una URL"
required><br>

    <label for="sexe">Sexe:</label>
    <select id="sexe" name="sexe" required>
        <option value="Home">Home</option>
        <option value="Dona">Dona</option>
        <option value="Altres">Altres</option>
    </select><br>

    <button type="submit">Enviar</button>
</form>
</body>
</html>

```

```

<?php
session_start();

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $_SESSION['nom'] = $_POST['nom'];
    $_SESSION['email'] = $_POST['email'];
    $_SESSION['url'] = $_POST['url'];
    $_SESSION['sexe'] = $_POST['sexe'];
}
?>

<!DOCTYPE html>
<html lang="ca">
<head>
    <meta charset="UTF-8">
    <title>Formulari 2</title>
</head>
<body>
<h1>Formulari 2: Dades Addicionals</h1>
<form method="post" action="exercici127formulari3.php">
    <label for="convivents">Nombre de convivents:</label>
    <input type="number" id="convivents" name="convivents" placeholder="Escriu
cuantes personas" required><br>

    <label for="aficions">Aficions:</label>
    <input type="text" id="aficions" name="aficions" placeholder="Escriu les
teves aficions" required><br>

    <label for="menu">Tipus de menú:</label>
    <select id="menu" name="menu" required>
        <option value="Vegetarià">Vegetarià</option>
        <option value="Omnívor">Omnívor</option>
        <option value="Vegà">Vegà</option>
    </select><br>

    <button type="submit">Enviar</button>
</form>
</body>
</html>

```

```

<?php
session_start();

```

```
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $_SESSION['convivents'] = $_POST['convivents'];
    $_SESSION['aficions'] = $_POST['aficions'];
    $_SESSION['menu'] = $_POST['menu'];
}
?>

<!DOCTYPE html>
<html lang="ca">
<head>
    <meta charset="UTF-8">
    <title>Resum de les Dades</title>
</head>
<body>
<h1>Resum de les Dades</h1>
<ul>
    <li><strong>Nom i Cognoms:</strong> <?php echo $_SESSION['nom']; ?></li>
    <li><strong>Email:</strong> <?php echo $_SESSION['email']; ?></li>
    <li><strong>URL:</strong> <?php echo $_SESSION['url']; ?></li>
    <li><strong>Sexe:</strong> <?php echo $_SESSION['sexe']; ?></li>
    <li><strong>Nombre de convivents:</strong> <?php echo
$_SESSION['convivents']; ?></li>
    <li><strong>Aficions:</strong> <?php echo $_SESSION['aficions']; ?></li>
    <li><strong>Tipus de menú:</strong> <?php echo $_SESSION['menu']; ?></li>
</ul>
<p><a href="exercici127formulari1.php">Tornar al Formulari 1</a></p>
<p><a href="buida_sessio.php">Buidar la sessió i tornar al Formulari 1</a></p>
</body>
</html>
```