PHP

**SINTAXIS**

Para escribir código php utilizamos la siguiente sintaxis: **<?php ?>**

**Escribir texto:** con la instrucción **echo**.

**Imprimir cualquier valor:** con la instrucción **print\_r()**.

**Imprimir un valor:** con la instrucción **print()**.

TODO DEBE CERRARSE CON PUNTO Y COMA ( ; ).

<?php

echo "Hola Mundo";

print\_r("Hola Joaquín");

print("Hello world");

?>

**MÉTODO POST**

**Concatenar:** utilizamos el punto (.).

**Variables:** con el signo peso + el nombre de la variable ($nombreDeLaVariable)

<?php

    if($\_POST){

    //Recibir información del formulario HTML (Método POST)

    $nombre = $\_POST["txtNombre"];

    echo "Hola ".$nombre;

    }

?>

<body>

    <form action="ejercicio2.php" method="post">

        <label for="">Nombre:</label>

        <br>

        <input type="text" name="txtNombre" id="">

        <br>

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

    </form>

</body>

**MÉTODO GET**

Estamos haciendo dos archivos, un html y un php. El formulariod el html va a viajar por GET, recibiendo esta información del archivo php.

**En el html:**

<body>

    <form action="ejercicio3.php" method="get">

        <label for="">Nombre:</label>

        <br>

        <input type="text" name="nombre" id="">

        <br>

        <button type="submit">Enviar</button>

    </form>

</body>

**En el php:**

<?php

    if ($\_GET) {

        $nombre = $\_GET["nombre"];

        echo "Hola ".$nombre;

    }

?>

**STRING Y CONCATENACIÓN**

Utilizamos el punto (.) para concatenar

<?php

    if($\_POST){

    $nombre = $\_POST["txtNombre"];

    $apellido = $\_POST["txtApellido"];

    echo "Hola ".$nombre." ".$apellido;

    }

?>

<form action="ejercicio5.php" method="post">

        <label for="">Nombre:</label>

        <input type="text" name="txtNombre" id="">

        <br>

        <label for="">Apellido:</label>

        <input type="text" name="txtApellido" id="">

        <br>

        <button type="submit">Enviar</button>

    </form>

**TIPOS DE DATOS**

<?php

    $Edad = 24;

    $Nombre = "Joaquín";

    $Altura = 1.77;

    $Programador = null;

    echo "Hola soy ".$Nombre.", mi edad es de ".$Edad.". Tengo una altura de ".$Altura." y mi puesto es ".$Programador;

?>

**CONSTANTES EN PHP**

Para hacer constantes utilizamos la instrucción **define(**nombre, valor**);**

<?php

    $edad = 32; //variable

    echo $edad."<br/>";

    $edad = 40; //cambiar valor de la variable

    echo $edad."<br/>";

    //uso de constante

    define("NOMBRE", "Joaquin");

    //Imprimir la constante

    echo NOMBRE

?>

**OPERADORES ARITMÉTICOS**

<?php

    if ($\_POST) {

        $valorA = $\_POST["valorA"];

        $valorB = $\_POST["valorB"];

        $suma = $valorA + $valorB;

        $resta = $valorA - $valorB;

        $division = $valorA / $valorB;

        $multiplicacion = $valorA \* $valorB;

        echo $suma."<br/>";

        echo $resta."<br/>";

        echo $division."<br/>";

        echo $multiplicacion;

    }

?>

<form action="ejercicio8.php" method="post">

        <label for="">Valor A:</label>

        <input type="text" name="valorA" id="">

        <br>

        <label for="">Valor B:</label>

        <input type="text" name="valorB" id="">

        <br>

        <button type="submit">Enviar</button>

    </form>

**OPERADORES RELACIONALES**

Se escriben igual que en Javascript. < - > - == - !=

<?php

    if ($\_POST) {

        $valorA = $\_POST["valorA"];

        $valorB = $\_POST["valorB"];

        if ($valorA > $valorB) {

            echo "El valor de A es mayor que el valor de B";

        }

    }

?>

<form action="ejercicio9.php" method="post">

        <label for="">Valor A:</label>

        <input type="text" name="valorA" id="">

        <br>

        <label for="">Valor B:</label>

        <input type="text" name="valorB" id="">

        <br>

        <button type="submit">Enviar</button>

    </form>

**OPERADORES LÓGICOS**

Como en JS también son los mismos: &&, ||

<?php

    if ($\_POST) {

        $valorA = $\_POST["valorA"];

        $valorB = $\_POST["valorB"];

        if (($valorA != $valorB) && ($valorA > $valorB)) {

            echo "El valor de A es diferente y mayor que el de B";

        }

    }

?>

**IF ANIDADO**

<?php

    if ($\_POST) {

        $valorA = $\_POST["valorA"];

        $valorB = $\_POST["valorB"];

        $suma = $valorA + $valorB;

        $resta = $valorA - $valorB;

        $division = $valorA / $valorB;

        $multiplicacion = $valorA \* $valorB;

        echo $suma."<br/>";

        echo $resta."<br/>";

        echo $division."<br/>";

        echo $multiplicacion."<br/>";

        if ($valorA == $valorB) {

            echo "El valor de A es igual que el valor de B <br/>";

            if ($valorA == 4) {

                echo "El valor es igual a 4";

            }

        }

    }

?>

<form action="ejercicio11.php" method="post">

        <label for="">Valor A:</label>

        <input type="text" name="valorA" id="">

        <br>

        <label for="">Valor B:</label>

        <input type="text" name="valorB" id="">

        <br>

        <button type="submit">Enviar</button>

    </form>

**IF ELSE**

<?php

    if ($\_POST) {

        $valorA = $\_POST["valorA"];

        $valorB = $\_POST["valorB"];

        if ($valorA > $valorB) {

            echo "El valor de A es mayor que el valor de B";

        } else {

            echo "El valor de B es mayor que el valor de A";

        }

    }

?>

<form action="ejercicio12.php" method="post">

        <label for="">Valor A:</label>

        <input type="text" name="valorA" id="">

        <br>

        <label for="">Valor B:</label>

        <input type="text" name="valorB" id="">

        <br>

        <button type="submit">Enviar</button>

    </form>

**SWITCH**

<?php

    if($\_POST){

        $button = $\_POST["btn"];

        switch($button){

            case 1:

                echo "Presionó el botón 1";

            break;

            case 2:

                echo "Presionó el botón 2";

            break;

            case 3:

                echo "Presionó el botón 3";

            break;

        }

    }

?>

<body>

    <form action="ejercicio13.php" method="post">

        <input type="submit" name="btn" value="1">

        <input type="submit" name="btn" value="2">

        <input type="submit" name="btn" value="3">

    </form>

</body>

**CICLO FOR**

<?php

    for ($i = 0; $i <= 10; $i++) {

        echo "Número ".$i."<br/>";

    }

?>

**CICLO DO WHILE**

<?php

    $i= 0;

    do{

        echo "Número ".$i."<br/>";

        $i++;

    } while ($i <= 10)

?>

**CICLO WHILE**

<?php

    $i= 0;

    while ($i <= 10) {

        echo "Número ".$i."<br/>";

        $i++;

    }

?>

**FUNCIONES EN PHP**

<?php

    function imprimirNombre ($nombre, $apellido ="") {

        echo "Hola ".$nombre." ".$apellido."<br/>";

    }

    imprimirNombre("Joaquín");

    imprimirNombre("Franco", "Di Stefano");

    imprimirNombre("Matías", "Gómez");

?>

**FUNCIONES MATEMÁTICAS**

Podemos buscar funciones establecidas por PHP en la documentación oficial, como lo es la función **rand()**.

<?php

    $numeroAleatorio = rand(1, 10);

    echo $numeroAleatorio;

?>

**FUNCIONES PARA STRINGS**

<?php

    $nombre = "JoAquiN SErra";

    $nombreMayuscula = strtoupper($nombre);

    echo $nombreMayuscula;

?>

**FUNCIONES DE FECHA**

<?php

    $hoy = date("Y-m-d");

    echo $hoy;

?>

**ARRAYS NUMÉRICOS**

<?php

    $frutas = array("Fresa", "Pera", "Manzana");

    print\_r($frutas)

?>

**ARRAYS ASOCIATIVOS**

Podemos asignarles el índice a los elementos en PHP de la siguiente manera: “como queremos que se llame el índice” => elemento.

<?php

    $frutas = array("f" => "Fresa", "p" => "Pera", "m" => "Manzana");

    print\_r($frutas);

?>

**LEER ARRAYS**

<?php

    $frutas = array("Fresa", "Pera", "Manzana");

    print\_r($frutas);

    echo "<br/>".$frutas[1]."<br/>";

    for ($i=0; $i < 3; $i++) {

        echo $frutas[$i].", ";

    }

?>

<?php

    $frutas = array("f" => "Fresa", "p" => "Pera", "m" => "Manzana");

    print\_r($frutas);

    echo "<br/>".$frutas["m"]."<br/>";

    foreach ($frutas as $i => $value) {

        echo "El valor ".$value.", tiene el índice: ".$i."<br/>";

    }

?>

**FUNCIONES PARA ARRAYS**

Leer la documentación para ver más funciones.

<?php

    $frutas = array("Fresa", "Pera", "Manzana");

    array\_push($frutas, "Uva");

    print\_r($frutas)

?>

**CLASES EN PHP**

Crearemos un objeto a través de un **class**.

<?php

    class persona {

        public $nombre;

        public function asignarNombre ($nuevoNombre) {

            $this -> nombre = $nuevoNombre."<br/>";

        }

        public function imprimirNombre (){

            echo "Hola mi nombre es ".$this->nombre;

        }

    }

    $objAlumno = new persona();

    $objAlumno -> asignarNombre("Joaquín");

    $objAlumno -> imprimirNombre();

    echo $objAlumno -> nombre;

?>

**VISIBILIDAD DE DATOS**

Si queremos acceder a una variable que tiene la palabra **private**, la única forma de acceder a ella es a través una function que este **public**.

<?php

    class persona {

        public $nombre;

        private $edad;

        public function asignarNombre ($nuevoNombre) {

            $this -> nombre = $nuevoNombre."<br/>";

        }

        public function imprimirNombre (){

            echo "Hola mi nombre es ".$this->nombre;

        }

        public function mostrarEdad () {

            $this->edad = 24;

            return $this->edad;

        }

    }

    $objAlumno = new persona();

    $objAlumno -> asignarNombre("Joaquín");

    $objAlumno -> imprimirNombre();

    echo $objAlumno -> nombre;

    echo $objAlumno -> mostrarEdad();

?>

**HERENCIA EN PHP**

<?php

    class persona {

        public $nombre;

        private $edad;

        public function asignarNombre ($nuevoNombre) {

            $this -> nombre = $nuevoNombre;

        }

        public function imprimirNombre (){

            echo "Hola mi nombre es ".$this->nombre;

        }

        public function mostrarEdad () {

            $this->edad = 24;

            return $this->edad;

        }

    }

    class trabajador extends persona {

        public $puesto;

        public function presentarseComoTrabajador () {

            echo "Hola soy ".$this->nombre." y soy ".$this->puesto;

        }

    }

    $objTrabajador = new trabajador();

    $objTrabajador -> asignarNombre("Franco");

    $objTrabajador -> puesto = "Estudiante";

    $objTrabajador -> presentarseComoTrabajador();

?>

**CONSTRUCTOR EN PHP**

Con el **\_\_construct** predefinimos que acciones queremos que se inicien al llamar al objeto **persona**.

<?php

    class persona {

        public $nombre;

        private $edad;

        function \_\_construct($nuevoNombre){

            $this -> nombre = $nuevoNombre;

        }

        public function asignarNombre ($nuevoNombre) {

            $this -> nombre = $nuevoNombre;

        }

        public function imprimirNombre (){

            echo "Hola mi nombre es ".$this->nombre;

        }

        public function mostrarEdad () {

            $this->edad = 24;

            return $this->edad;

        }

    }

    $objAlumno = new persona("Joaquín");

    $objAlumno -> imprimirNombre();

?>

**MÉTODOS ESTÁTICOS**

<?php

    class unaClase {

        public static function unMetodo () {

            echo "Hola soy un método estático";

        }

    }

    unaClase::unMetodo();

?>

**CONEXIÓN PHP CON MySQL**

**PDO** nos permite conectarnos a la base de datos.

<?php

    $servidor = "localhost"; //127.0.0.1

    $usuario = "root";

    $password = "";

    try {

        $conexion = new PDO("mysql:host=$servidor;dbname=album", $usuario, $password);

        $conexion->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

        echo "Conexión establecida";

    } catch (PDOException $error) {

        echo "ERROR en la conexión".$error;

    }

?>

**INSERTAR DATOS en MySQL con PHP**

Primero en phpMyAdmin debemos crear una tabla con sus filas. Luego insertamos datos con **exec** desde el archivo **php**.

<?php

    $servidor = “localhost”; //127.0.0.1

    $usuario = “root”;

    $password = “”;

    try {

        $conexion = new PDO(“mysql:host=$servidor;dbname=lbum”, $usuario, $password);

        $conexion->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

        $sql=”INSERT INTO `fotos` (`id`, `name`, `path`) VALUES (NULL, ‘Jugando con la programación’, ‘foto.jpg’);”;

        $conexion->exec($sql);

        echo “Conexión establecida”;

    } catch (PDOException $error) {

        echo “ERROR en la conexión”.$error;

    }

¿>

**LEER DATOS DE MySQL**

<?php

    $servidor = "localhost"; //127.0.0.1

    $usuario = "root";

    $password = "";

    try {

        $conexion = new PDO("mysql:host=$servidor;dbname=album", $usuario, $password);

        $conexion->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

        $sql="SELECT \* from `fotos`";

        // Preparo la selección

        $sentencia=$conexion->prepare($sql);

        // Ejecuto la seleccion

        $sentencia->execute();

        // Obtengo los resultados

        $resultado=$sentencia->fetchAll();

        // Y aca lo leo con el print o el foreach

        // print\_r($resultado);

        // foreach

        foreach ($resultado as $foto) {

            echo $foto["name"]."<br/>";

        }

        echo "Conexión establecida";

    } catch (PDOException $error) {

        echo "ERROR en la conexión".$error;

    }

?>

**VARIABLES DE SESSION**

Las variables de sesión, las puedo usar en cualquier otro archivo de PHP una vez inicializadas. Solo debo poner en el archivo:

**sesión\_start();**

**$\_SESSION[“**nombreDeLaSession**”];**

<?php

    session\_start();

    $\_SESSION["usuario"]="joaquincaggiano";

    $\_SESSION["status"]="loggeado";

    echo "Sesión iniciada".":<br/>";

    echo "Usuario: ".$\_SESSION["usuario"].". Status: ".$\_SESSION["status"];

?>

**DESTRUIR VARIABLES DE SESSION**

<?php

    session\_start();

    session\_destroy();

    echo "Se destruyo la sesión"

?>

Si volvemos al ejercicio 30\_1 nos va a decir que tenemos un error porque ahora nosotros borramos la sesión. Entonces debemos hacer un condicional:

<?php

    session\_start();

    // La function isset sirve para saber si hay algo vacío

    if (isset($\_SESSION["usuario"])) {

        echo "Usuario: ".$\_SESSION["usuario"].". Status: ".$\_SESSION["status"];

    } else {

        echo "No hay datos";

    }

?>

**VALORES DEL INPUT TEXT**

Puedo usar las etiquetas **php** dentro del HTML para poner código dinámico como en EJS.

<?php

    $txtNombre="";

    if($\_POST){

        $txtNombre= (isset($\_POST["txtNombre"])) ? $\_POST["txtNombre"] : "";

    }

?>

<body>

    <?php if($\_POST) { ?>

    <strong>Hola: </strong><?php echo $txtNombre; ?>

    <?php } else { ?>

        <?php echo "<strong>Hola Usuario</strong>" ?>

    <?php } ?>

    <form action="ejercicio31.php" method="post">

        <input value="<?php echo $txtNombre; ?>" type="text" name="txtNombre" id="">

        <button type="submit">Enviar</button>

    </form>

</body>

**VALORES DEL INPUT RADIO**

<?php

    $rdgLenguaje="";

    if($\_POST){

     $rdgLenguaje= (isset($\_POST["lenguaje"])) ? $\_POST["lenguaje"] : "";

        // print\_r($\_POST); //esto es para ver que me llego como en un console.log

        // print\_r($rdgLenguaje);

    }

?>

<body>

    <?php if($\_POST) { ?>

    <strong>Tu lenguaje es: </strong><?php echo $rdgLenguaje; ?>

    <?php } ?>

    <form action="ejercicio31.php" method="post">

        <label for="">¿Te gusta?</label>

        <br>

        <label for="">PHP:</label><input type="radio" <?php echo ($rdgLenguaje == "php" ? "checked":"") ?> name="lenguaje" value="php" id="">

        <br>

        <label for="">HTML:</label><input type="radio" <?php echo ($rdgLenguaje == "html" ? "checked":"") ?> name="lenguaje" value="html" id="">

        <br>

        <label for="">CSS:</label><input type="radio" <?php echo ($rdgLenguaje == "css" ? "checked":"") ?> name="lenguaje" value="css" id="">

        <br>

        <button type="submit">Enviar</button>

    </form>

</body>

**VALORES INPUT CHECKBOX**

<?php

    $chkphp="";

    $chkhtml="";

    $chkcss="";

    if($\_POST){

        $chkphp=( isset($\_POST["chkphp"]) == "si" ) ? "checked": "";

        $chkhtml=( isset($\_POST["chkhtml"]) == "si" ) ? "checked": "";

        $chkcss=( isset($\_POST["chkcss"]) == "si" ) ? "checked": "";

    }

?>

<body>

<form action="ejercicio31.php" method="post">

        <h2>Estas aprendiendo...</h2>

        <label for="">PHP:</label><input type="checkbox" <?php echo $chkphp; ?> value="si" name="chkphp" id="">

        <br>

        <label for="">HTML:</label><input type="checkbox" <?php echo $chkhtml; ?> value="si" name="chkhtml" id="">

        <br>

        <label for="">CSS:</label><input type="checkbox" <?php echo $chkcss; ?> value="si" name="chkcss" id="">

        <br>

        <button type="submit">Enviar</button>

    </form>

</body>

**VALORES SELECT OPTION**

<?php

    $lsAnime="";

    if($\_POST){

        $lsAnime=( isset($\_POST["lsAnime"]) ) ?$\_POST["lsAnime"]:"";

        // print\_r($lsAnime);

    }

?>

<body>

    <?php if($\_POST) { ?>

        <strong>Tu anime favorito es:</strong>

        <?php echo $lsAnime; ?>

    <?php } ?>

    <form action="ejercicio31.php" method="post">

        <h2>¿Que Anime te gusta?</h2>

        <select name="lsAnime" id="">

            <option value="">Ninguna serie</option>

            <option <?php echo ($lsAnime == "naruto" ? "selected" : ""); ?> value="naruto">Naruto</option>

            <option <?php echo ($lsAnime == "bleach" ? "selected" : ""); ?> value="bleach">Bleach</option>

            <option <?php echo ($lsAnime == "dragonBall" ? "selected" : ""); ?> value="dragonBall">Dragon Ball</option>

        </select>

        <br>

        <button type="submit">Enviar</button>

    </form>

</body>

**VALORES TEXTAREA**

<?php

    $txtComentario="";

    if($\_POST){

        $txtComentario= (isset($\_POST["txtComentario"])) ? $\_POST["txtComentario"] : "";

    }

?>

<body>

    <?php if($\_POST) { ?>

        <strong>Tu mensaje es: </strong><?php echo $txtComentario; ?>

    <?php } ?>

    <form action="ejercicio31.php" method="post">

        <h2> ¿Tienes alguna duda? </h2>

        <textarea name="txtComentario" id="" cols="30" rows="10">

            <?php echo $txtComentario; ?>

        </textarea>

        <br>

        <button type="submit">Enviar</button>

    </form>

</body>

**VALORES INPUT FILE**

Necesitamos en el form el **enctype=”multipart/form-data”**.

<?php

    if($\_POST){

        print\_r($\_POST);

        print\_r($\_FILES["archivo"]);

        // print\_r($\_FILES["archivo"]["name"]);

    }

?>

<body>

    <form action="ejercicio32.php" enctype="multipart/form-data" method="post">

        <label for="">Imagen:</label>

        <input type="file" name="archivo" id="">

        <br>

        <input name="enviar" type="submit" value="Enviar">

    </form>

</body>

**GUARDAR ARCHIVO CON PHP**

Utilizamos la function **move\_uploaded\_file();** para obtener el archivo original. Como primer parámetro pasamos el nombre temporal del archivo, y como segundo parámetro pasamos la ruta donde queremos guardar el archivo.

<?php

    if($\_POST){

        print\_r($\_POST);

        // print\_r($\_FILES["archivo"]);

        print\_r($\_FILES["archivo"]["name"]);

        move\_uploaded\_file($\_FILES["archivo"]["tmp\_name"], $\_FILES["archivo"]["name"]);

    }

?>

<body>

    <form action="ejercicio32.php" enctype="multipart/form-data" method="post">

        <label for="">Imagen:</label>

        <input type="file" name="archivo" id="">

        <br>

        <input name="enviar" type="submit" value="Enviar">

    </form>

</body>

**FUNCIÓN PARA JSON DECODE**

Utilizamos **json\_decode** para decodificar un JSON.

<?php

    $jsonContenido='[

        {"nombre" : "Joaquín", "apellido" : "Serra"},

        {"nombre" : "Franco", "apellido" : "Gomez"}

    ]';

    $resultado=json\_decode($jsonContenido);

    // print\_r($resultado);

    foreach($resultado as $persona){

        echo $persona->nombre." ".$persona->apellido."<br/>";

    }

?>

Para volver a transformar en un JSON utilizamos **json\_encode();**

<?php

    $personas= array("Joaquin"=>24, "Alumine"=>21, "Marisa"=>48);

    echo json\_encode($personas);

?>

**CONSUMIR UNA API**

<?php

    $url="https://api.dailymotion.com/videos?channel=sport&limit=10";

    // para leer la lectura de sitios con https necesitamos crear un array con las opciones para consumir esos datos

    $opciones= array("ssl"=>array("verify\_peer"=>false, "verify\_peer\_name"=>false));

    $respuesta=file\_get\_contents($url, false, stream\_context\_create($opciones));

    $objRespuesta= json\_decode($respuesta);

    // print\_r($objRespuesta);

    // foreach($objRespuesta->list as $video){

    //     print\_r($video->title);

    //     print\_r($video->channel);

    // }

?>

<ul>

    <?php foreach($objRespuesta->list as $video) { ?>

       <li><?php echo $video->title ?> | <?php echo $video->channel ?></li>

    <?php } ?>

</ul>

**FUNCTION INCLUDE**

Funciona para pegar el contenido de un archivo en otro. Lo podemos utilizar para el header y footer del sitio web.

Documento 36\_1.php:

<?php

    echo "Hola develoteca";

?>

Documento 36.php:

<body>

    <?php include "ejercicio36\_1.php"; ?>

</body>

Así va a aparecer en el documento 36.php el contenido del documento 36\_1.php.

**FUNCTION REQUIRE**

Ejercicio 37\_1.php:

<?php

    echo "Hola mundo";

?>

Ejercicio 37.php:

<body>

    <?php require("ejercicio37\_1.php"); ?>

</body>

**DIFERENCIA ENTRE EL INCLUDE Y EL REQUIRE**

La diferencia entre el **include** y el **require** es que el include si no tiene errores dentro del interior del include muestra un warning, pero va a seguir mostrando las siguientes instrucciones con <?php?>.

**Include:**

<body>

    <?php include "ejercicio36\_.php"; ?> <!-- me equivoco por ejemplo en el nombre del archivo -->

    <?php echo "Hola estoy en la página principal"; ?>

</body>

Me va a mostrar el warning, pero el siguiente contenido me lo muestra igual:

**Warning**: include(ejercicio36\_.php): Failed to open stream: No such file or directory in **D:\xampp\htdocs\Ejercicios\ejercicio36.php** on line **10**  
  
**Warning**: include(): Failed opening 'ejercicio36\_.php' for inclusion (include\_path='D:\xampp\php\PEAR') in **D:\xampp\htdocs\Ejercicios\ejercicio36.php** on line **10**  
Hola estoy en la página principal

**Require:**

El require no muestra el siguiente código si hay un error:

<body>

    <?php require("ejercicio37\_.php"); ?><!-- Me equivoco en el nombre del archivo -->

    <?php echo "Hola estoy en la página principal"; ?><!-- No me muestra el contenido -->

</body>

**INCLUDE\_ONCE Y REQUIRE\_ONCE**

Se utiliza para que solo lea el include o require una sola vez en caso de haberlo puesto dos veces.

<?php require\_once("ejercicio37\_1.php"); ?><!-- Va a leer solo este -->

<?php require("ejercicio37\_1.php"); ?>

**ABRIR UN ARCHIVO EN PHP**

<?php

    $archivo="contenido.txt";//si esta en otra carpeta el archivo hay que armar la ruta

    $archivoAbierto= fopen($archivo, "r");//abrimos el archivo en modo de lectura con "r"

    $contenido= fread($archivoAbierto, filesize($archivo));//leemos todo el contenido de ese archivo con su peso

    echo $contenido; //imprimimos el archivo

?>

**ESCRITURA DE ARCHIVOS**

<?php

    $nombreArchivo="archivo.txt";

    $contenidoArchivo= "Hola como estas?";

    $archivoACrear= fopen($nombreArchivo, "w");//abrimos el archivo en modo escritura "w"

    fwrite($archivoACrear, $contenidoArchivo); //llenamos el contenido del archivo

    fclose($archivoACrear);//cerramos el archivo para que no quede abierto por la aplicación

?>