

# 06

# **Desenvolupament Web en entorn Client**

**Cicle: Desenvolupament d'aplicacions web**

**Curs 2024-2025**

©José Masip Alonso

## 1. Introducció

En aquest document aprendrem a utilitzar AJAX amb php i BD que generaran resultats amb json o XML.

## 2. Contingut

### 2.1. Interactuar amb un servidor de php.

En aquest apartat anem a veure com interactuem amb un servidor php. El php rebrà les dades necessàries, realitzarà els càlculs i ens retornarà la informació oportuna.

Vegem un exemple molt senzill. El nostre php no rep dades i sols ens retorna un text.

#### 015ajax.php

```
<?php  
echo("hola mon");  
?>
```

El nostre javascript per a la connexió ajax és:

```
<script type="text/javascript">  
  
var peticio_http;  
  
function mostraContingut() {  
  // 4 Completado.  
  if(peticio_http.readyState == 4) {  
    if(peticio_http.status == 200) { //si tot OK  
      document.getElementById("capa").innerHTML  
+=peticio_http.responseText;  
    }  
    else{ //mostra el estat en format de text  
      alert("error: "+peticio_http.statusText);  
    }  
  }  
}
```

```
function llegirdades() {  
    // Obtener instancia objeto XMLHttpRequest  
    if(window.XMLHttpRequest) {  
        peticio_http = new XMLHttpRequest();  
    }  
    else if(window.ActiveXObject) { //navegadors obsolets  
        peticio_http = new ActiveXObject("Microsoft.XMLHTTP");  
    }  
    // Prepara funcio rep dades del servidor  
    peticio_http.onreadystatechange = mostraContingut;  
    // Realizar peticio HTTP  
    peticio_http.open('GET', './ajax_php/015ajax.php', true);  
    //comence el proces. envies les dades  
    peticio_http.send(null);  
}  
</script>
```

El php ens retorna les dades com a text, per tant, simplement treballem amb elles a traves del responseText.

## 2.2. Enviar dades amb GET

Per a enviar les dades amb el mètode GET tenim que enviar-les a traves de la URL.

```
function llegirdades() {  
    // Obtener instancia objeto XMLHttpRequest  
    if(window.XMLHttpRequest) {  
        peticio_http = new XMLHttpRequest();  
    }  
    else if(window.ActiveXObject) { //navegadors obsolets  
        peticio_http = new ActiveXObject("Microsoft.XMLHTTP");  
    }  
    // Prepara funcio rep dades del servidor  
    peticio_http.onreadystatechange = mostraContingut;  
    // Realizar peticio HTTP  
    peticio_http.open('GET', './ajax_php/018ajax.php?variable=3', true);  
    //comence el proces. envies les dades  
    peticio_http.send(null);  
}
```

El php que rep les dades és:

## 018ajax.php

```
<?php  
if($_REQUEST["variable"]==1){  
    echo("11111111111111111111111111111111");  
}  
else{  
    echo("NO es 1");  
}  
?>
```

Si executem el codi amb

```
peticio_http.open('GET', './ajax_php/018ajax.php?variable=3', true);
```

Aquest trau la per pantalla el missatge: «NO es 1».

Si executem el codi amb

```
peticio_http.open('GET', './ajax_php/018ajax.php?variable=1', true);
```

Aquest trau la per pantalla el missatge: «111111111111111111111111111111».

Si volem enviar mes d'un valor cal separar-los amb un &

```
./ajax_php/a.php?variable=valor&variable2=valor
```

### 2.3. Enviar dades amb POST

Per a enviar les dades amb el mètode POST tenim que enviar-les a través del mètode send de l'objecte XMLHttpRequest.

```

peticio_http.open('POST', './ajax_php/019ajax.php', true);
//indiquem que les dades estan codificades com un formulari
peticio_http.setRequestHeader("Content-Type","application/x-www-form-
urlencoded");
//comence el proces. envies les dades
peticio_http.send("variable=3");

```



codi d'ajax:

```
<script type="text/javascript">
var peticio_http;

function mostraContingut() {
// 4 Completado.
    if(peticio_http.readyState == 4) {
        if(peticio_http.status == 200) { //si tot OK
            //la resposta amb etiquetes i tot.
            alert(peticio_http.responseText);
            //al passar la capçalera ho reconeix com a XML
            alert(peticio_http.responseXML);

document.getElementById("capa").innerHTML=peticio_http.responseXML.getE
lementsByTagName("hola")[0].innerHTML;

        }
        else{ //mostra el estat en format de text
            alert("error: "+peticio_http.statusText);
        }
    }
}

function llegirdades() {
    // Obtener instancia objete XMLHttpRequest
    if(window.XMLHttpRequest) {
        peticio_http = new XMLHttpRequest();
    }
    else if(window.ActiveXObject) { //navegadors obsolets
        peticio_http = new ActiveXObject("Microsoft.XMLHTTP");
    }
    // Prepara funcio rep dades del servidor
    peticio_http.onreadystatechange = mostraContingut;
    // Realizar peticio HTTP
    peticio_http.open('GET', './ajax_php/017ajax.php', true);
    //comence el proces. envies les dades
    peticio_http.send(null);
}

window.onload=llegirdades;
</script>
```

Si realitzem canvis al Contingut de la salutació ho vorem automàticament ja que no es manté en memòria cau com passava abans.



Com podem veure a la pestanya xarxa del navegador les dades que retorna el php són xml.

## 2.5. Ajax i php amb BD.

En aquest apartat vorem un exemple per a accedir a les dades del servidor de mariadb que tenim al docker. Cal importar o crear primer aquesta bd des de phpmyadmin.

L'arxiu php connecta amb la bd i retorna les dades com a text. A aquest exemple el format el fa el php, **com nosaltres estem a llenguatge client farem el format des de javascript.**

```
<?php
$servername= "localhost";
$username= "root";
$password="";
$dbname="bd_ajax";

//crear la connexio
$con= new mysqli($servername, $username, $password, $dbname);

//Comprovar que hem connectat
if ( $con->connect_error){
    die("ERROR DE CONNEXIO: ". $con->connect_error);
}

//executar consulta
$sql= "select * from taula";
$result=$con->query($sql);

if ($result->num_rows >0){//hi ha resultats
    //imprimir els resultats
    while ($fila=$result->fetch_assoc())
    {
```

```
        echo "camp a:".$fila['a']."<br>";
        echo "camp b:".$fila['b']."<br>";
        echo "camp c:".$fila['c']."<br>";
        echo "<br>";
    }
}
else{
    echo "no hi ha resultats.";
}

//tancar connexio
$con->close();
?>
```

### El codi javascript

```
var peticio_http;

function autocomp() {
// 0 No inicializado. 1 Cargando. 2 Cargado. 3 Interactivo. 4 Completado.
    if(peticio_http.readyState == 4) {
        if(peticio_http.status == 200) { //si tot OK
            //reiniciar capa resultats
            document.getElementById("resultats").innerHTML=peticio_http.responseText;
        }
        else{ //mostra el estat en format de text
            alert("error: "+peticio_http.statusText);
        }
    }
    // readyState=4 cargando mentres obtinc les dades
    else{
        document.getElementById("resultats").innerHTML="<img
src='./ima/cargando2.gif' />";
    }
}

function autocompletar() {
    // Obtener instancia objeto XMLHttpRequest
    if(window.XMLHttpRequest) {
        peticio_http = new XMLHttpRequest();
    }

    // Prepara funcio rep dades del servidor
    peticio_http.onreadystatechange = autocomp;

    // Realizar peticio HTTP
```



```
        peticio_http.open('POST', './ajax_php/029ajax.php', true);
        //indiquem que les dades estan codificades com un formulari
        peticio_http.setRequestHeader("Content-Type","application/x-www-form-
urlencoded");
        //comence el proces. envies les dades
        peticio_http.send(null);
    }

    function inici(){
        document.getElementById("boto").addEventListener('click',autocompletar,false);
    }

    window.onload=inici;
```

## 2.6. json.

<http://www.json.org/json-es.html>

«**JSON** (JavaScript Object Notation - Notación de Objetos de JavaScript) es un formato ligero de intercambio de datos.»

JSON utilitza dos estructures basiques:

- Una colecció de parelles nomvariable:valor.
- Una llista o vector de valors.

Un objecte es un conjunt desordenat de parelles nomvariable:valor (comença en una { i acaba en } i si especifiquem més d'una parella les separem per «,»).

Per exemple:

```
{ "a":1, "b": "ieee", "c":11 }
```

Una llista o vector es una col·lecció de valors (comença en una [ i acaba en ] i si especifiquem més d'un valor els separem per «,»). Per exemple:

```
[1, "ieee",11]
```

*Un exemple més elaborat seria aquest vector d'objectes:*

```
[{ "a":1, "b": "ieee", "c":11 },
 { "a":2, "b": "oeee", "c":34 },
 { "a":3, "b": "text", "c":534 }]
```

Altre exemple mes elaborat un objecte amb un llistat de nom:

```
{contactes:[
  {"nom":"juan", "cognoms":"garcia garcia" },
  {"nom":"luis", "cognoms":"de la fuente ruiz" },
  {"nom":"Pere", "cognoms":"perez garcia" },
  {"nom":"Pedro", "cognoms":"joan lopez" },
]}
```

### 2.7. Exemple D'AJAX amb json.

El nostre servidor ens retornarà una estructura json i tindrem que transformar aquesta a javascript per a poder utilitzar-la. Per a transformar el json a variables javascript tenim dos opcions:

- `eval()`. No es recomanable perquè ens podrien fer atacs si no validem be el codi. Si avaluem el valor d'un camp de text `eval(input.value)` si al camp ens escriuen `alert("ieee")` apareixeria la finestra emergent.
- `JSON.parse()`. [https://www.w3schools.com/js/js\\_json\\_parse.asp](https://www.w3schools.com/js/js_json_parse.asp) Aquesta funció valida que el json estiga format correctament. Cal utilitzar comilles simples fora i dins comilles dobles per als strings.

```
let vjson=JSON.parse('{ "a":1, "b": "ieee", "c":11 }');
```

crearia una estructura on utilitzant `vjson.a` accediríem al valor.

Arxiu php

```
<?php
$cadenajson='{ "a":["asdf", "qwer", "zxcv"],"b":5}';
// el ordre de les comiles es important per al json.parse.
//$cadenajson='{ 'a':['asdf', 'qwer', 'zxcv'], "b":5}';
//SyntaxError: JSON.parse: expected property name or '}'
?>
```

codi javascript (sols la funció que rep les dades)

```
function RepDadesAjax() {  
    if(peticio_http.readyState == 4) {  
        if(peticio_http.status == 200) { //si tot OK  
            var vjson = JSON.parse(peticio_http.responseText);  
            alert('a valor posicio 0:'+vjson.a[0]);  
            alert('a valor posicio 1:'+vjson.a[1]);  
            alert('b:'+vjson.b);  
        }  
        else{ //mostra el estat en format de text  
            alert("error: "+peticio_http.statusText);  
        }  
    }  
}
```

### 3. Activitats.

- E01. Tenim un formulari des d'on podem elegir de quin arxiu volem rebre les dades. Realitzar una connexió ajax que mostre el contingut a la capa.



Arxiu.php

```
<?php
//obrir arxiu
$f=fopen($_REQUEST["arxiu"],"r");
//llegir 1 linia arxiu
$l=fgets($f);

echo($l);
fclose($f);
?>
```

- E02. Puntuar una foto a través d'una petició ajax. Enviar les dades per GET.



## arxiu.php

```

<?php
//obrir arxiu
    $f=fopen("../ajax_txt/vots.txt","a");
//escriure al document
    fputs($f,$_REQUEST["punts"]."\n");
    fclose($f);

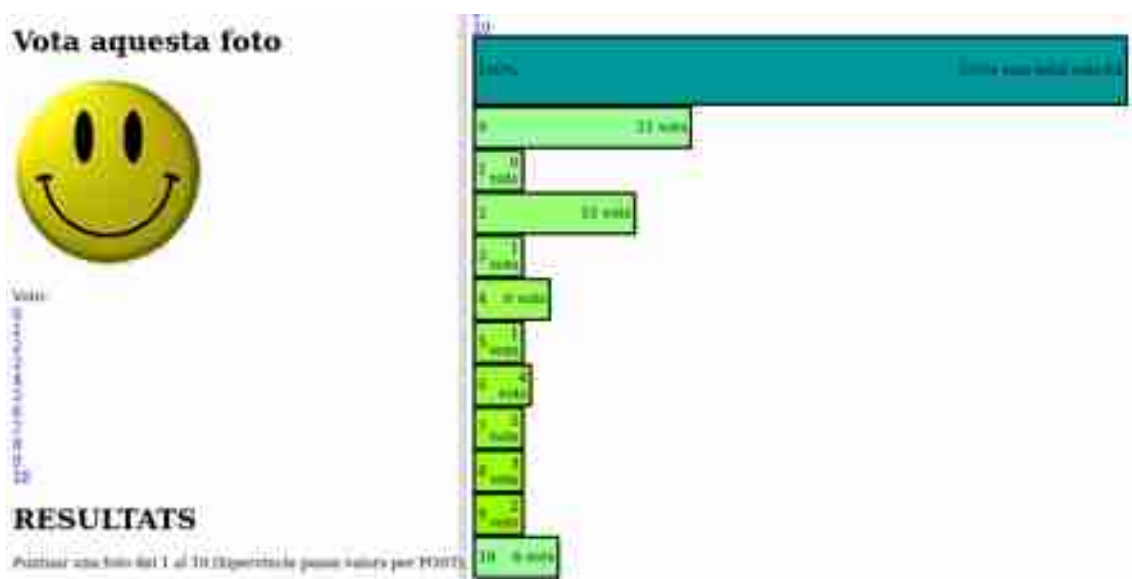
// echo($_REQUEST["punts"]);

//obrir arxiu
    $f=fopen("../ajax_txt/vots.txt","r");

$l="";
//mentres queden linies
    while (! feof($f))
    {
        $l=$l . fgets($f) . "<br/>";
    }
    echo($l);
//tancar arxiu
    fclose($f);
?>

```

- E03. Idem anterior però pulsem el vot amb un hipervincle i generem unes estadístiques processant les dades de l'arxiu en el javascript. Utilitzar el metode POST per a la petició ajax.



## Arxiu.php

```

<?php
//numero vots
$numvots = array(0,0,0,0,0,0,0,0,0,0);
//per a mostrar el vector
//var_dump($numvots);

//obrir arxiu
    $f=fopen("../ajax_txt/vots.txt","a");
//escriure al document
    fputs($f,$_REQUEST["punts"]."\n");
    fclose($f);

//    echo($_REQUEST["punts"]);

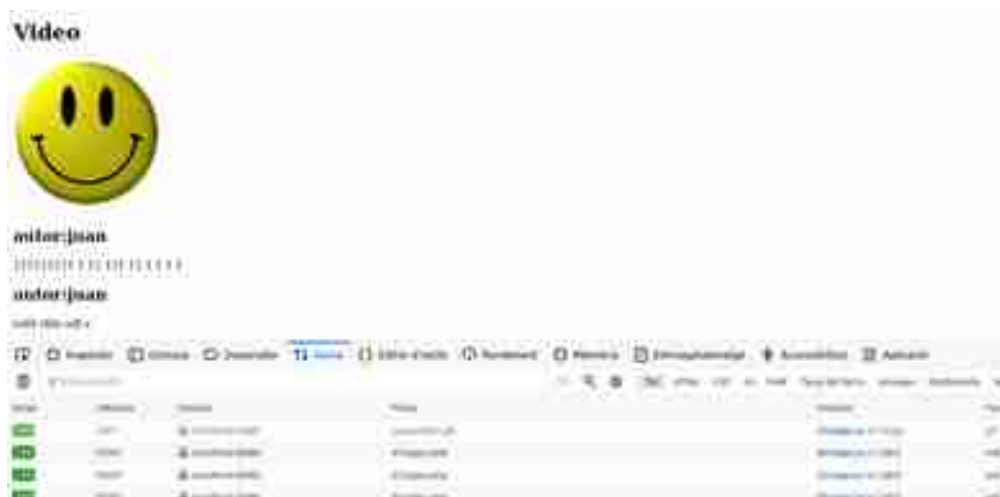
//obrir arxiu
    $f=fopen("../ajax_txt/vots.txt","r");

//total de vots
$cont=0;
$l="";
//mentres queden línies
    while (!feof($f))//contar un vot mes per al valor llegit
    {
        $l=$l . fgets($f);
    }
    echo($l);
//tancar arxiu
    fclose($f);

//generar estadístiques en javascript.
?>

```

- E04. Mostrar els comentaris d'un video. Aquest s'actualitzen cada 20 segons amb una petició ajax (importar BD comentaris\_ajax)



## Arxiu.php

```
<?php
$servername= "localhost";
$username= "root";
$password="";
$dbname="comentaris_ajax";

//video del que mostre els comentaris
$vid=$_REQUEST['video'];

//crear la connexio
$con= new mysqli($servername, $username, $password, $dbname);

//Comprovar que hem connectat
if ( $con->connect_error){
    die("ERROR DE CONNEXIO: ". $con->connect_error);
}

//executar consulta
$sql="SELECT * FROM `comentari` WHERE `id_video`=".$vid;
//echo $sql;
$result=$con->query($sql);

if ($result->num_rows >0){//hi ha resultats
    //crear xml
    $xml="<?xml version='1.0'?'>\n";
    $xml.="<comentaris>\n";

    //imprimir els resultats
    while ($fila=$result->fetch_assoc())
    {
        $xml.="<comentari>\n";
        $xml.="<autor>".$fila['autor']."</autor>\n";
        $xml.="<comenta>".$fila['comentari']."</comenta>\n";
        $xml.="</comentari>\n";
    }

    $xml.="</comentaris>\n";
    header('Content-Type: text/xml');
}
else{
    echo "no hi ha resultats.";
}

echo($xml);

//tancar connexio
$con->close();
?>
```

- E05. Idem anterior. Però paginem els resultats. Mostrem 4 comentaris cada volta.

PISTA: modificar sql per a que retorne sols 4 resultats a partir d'un indicat (variable ini).

```
$sql="SELECT * FROM `comentari` WHERE `id_video`=".$vid." limit ".$ini.",4";
```



- E06. Volem saber els horàries disponibles de tren des d'un origen a un destí. Per a això omplim el primer select des de php amb una consulta a la bd que ens trau tots els valors d'origen diferents. Al seleccionar un origen averiguarem tots els destins disponibles des d'aquest a través d'una petició ajax. Si després elegim un destí tindrem tots els horàries des d'aquest origen a aquest destí. Retornar les dades amb json (recomanació crear-lo manualment per a entendre millor com funciona)



Recomanació: Com hi ha dos peticions ajax crear dos php distints que generaran els jsons per a separar possibles errors.

Codi pagina d'inici

```
<!doctype html>
<html lang="es">
<head>
  <meta charset="utf-8">
  <title>AJAX</title>
  <meta name="description" content="descripcio">
  <meta name="author" content="Jose Masip">
  <script src="038ajax.js"></script>
</head>
<body>
  <h1>Origen - Desti estacio de tren</h1>
  origen
  <select id="origen">
    <option value="-1"> selecciona una opcio
    <?php
      $servername= "localhost";
      $username= "root";
      $password="";
      $dbname="tren_ajax";

      //crear la connexio
      $con= new mysqli($servername, $username, $password, $dbname);
      //Comprovar que hem connectat
      if ( $con->connect_error){
        die("ERROR DE CONNEXIO: ". $con->connect_error);
      }

      //executar consulta
      $sql= "SELECT DISTINCT origen FROM `rodalies` ";
      $result=$con->query($sql);

      if ($result->num_rows >0){//hi ha resultats
        //imprimir els resultats
        while ($fila=$result->fetch_assoc())
        {
          echo "<option>".$fila['origen'];
        }
      }
      else{
        echo "no hi ha resultats.";
      }
    }
  </select>

```

```
//tancar connexio
$con->close();

?>
</select>

desti
<select id="desti">
  <option value="-1"> selecciona una opcio
</select>
<div id="horaris"></div>
</body>
</html>
```

## Origen - Desti estacio de tren

origen  desti

## Origen - Desti estacio de tren

origen  desti

selecciona una opcio

valencia

xativa

## Origen - Desti estacio de tren

origen  → desti

selecciona una opció

valencia

sativa

gandia

| Estat | Mètode | Domini         | Fitxer      |
|-------|--------|----------------|-------------|
| 200   | GET    | localhost:8080 | 030ajax.php |
| 304   | GET    | localhost:8080 | 030ajax.js  |
| 404   | GET    | localhost:8080 | estilos.css |
| 200   | GET    | localhost:8080 | favicon.ico |
| 200   | POST   | localhost:8080 | 030ajax.php |

## Origen - Desti estacio de tren

origen  → desti

| Hora eixida | Hora arribada |
|-------------|---------------|
| 07:00:00    | 07:30:00      |
| 11:00:00    | 11:11:00      |
| 11:11:00    | 11:55:00      |

| Estat | Mètode | Domini         | Fitxer       |
|-------|--------|----------------|--------------|
| 200   | GET    | localhost:8080 | 030ajax.php  |
| 304   | GET    | localhost:8080 | 030ajax.js   |
| 404   | GET    | localhost:8080 | estilos.css  |
| 200   | GET    | localhost:8080 | favicon.ico  |
| 200   | POST   | localhost:8080 | 030ajax.php  |
| 200   | POST   | localhost:8080 | 030ajax2.php |

- E07. Exercici del parking realitzat a un document anterior. A aquesta versió les dades les obtenim des de la base de dades. Retornarem les dades en format XML (enviar sempre la mínima quantitat d'informació possible).
  - Al seleccionar un client farem una petició ajax per averiguar les matricules dels seus cotxes i omplirem el select corresponent.
  - Al seleccionar una matricula omplirem els detalls del cotxe
  - **AMPLIACIÓ:** implementar el camp per buscar client amb opció d'autocompletar (si escric 2 al camp mostrar tots els clients que comencen per 2, si escric 20 mostrar tots els clients que comencen per 20)
  - **AMPLIACIÓ:** implementar el camp per buscar per matricula.

**NOTA:** Per a aquesta funcionalitat utilitzar:

```
<input type="text" list="browsers">
<datalist id="browsers">
  < option value="Internet Explorer">
  < option value="Firefox">
  < option value="Chrome">
  < option value="Opera">
  < option value="Safari">
</datalist>
```

pagina d'inici de l'exercici

```
<!doctype html>
<html lang="es">
<head>
  <meta charset="utf-8">
  <title>AJAX</title>
  <meta name="description" content="descripcio">
  <meta name="author" content="Jose Masip">
  <script src="039ajax.js"></script>
</head>
<body>
<?php
  $servername= "localhost";
```

```

$username= "root";
$password="";
$dbname="parking_ajax";

//crear la connexio
$con= new mysqli($servername, $username, $password, $dbname);
//Comprovar que hem connectat
if ( $con->connect_error){
    die("ERROR DE CONNEXIO: ". $con->connect_error);
}
?>

<h1 align="center">Gesti&oacute; d'un parking</h1>

<form name="formulari">
<table align="center">
<tr>
<td width="60">client</td>
<td>
<select id="sclient">
<option value="-1"> selecciona una opcio
<?php
//executar consulta
$sql= "SELECT DISTINCT dni FROM `client` ";
$result=$con->query($sql);

if ($result->num_rows >0){//hi ha resultats
//imprimir els resultats
while ($fila=$result->fetch_assoc())
{
    echo "<option>".$fila['dni'];
}
}
else{
    echo "no hi ha resultats.";
}
?>
</select>
<input type="text" id="cclient" />

<div id="mclient"></div>
</td>
</tr>
<tr>
<td width="60">matricula</td>
<td>
<select id="smatricula">
<option value="-1"> selecciona una opcio

```

```

        <?php
            //executar consulta
            $sql= "SELECT DISTINCT matricula FROM `cotxe` ";
            $result=$con->query($sql);

            if ($result->num_rows >0){//hi ha resultats
                //imprimir els resultats
                while ($fila=$result->fetch_assoc())
                {
                    echo "<option>".$fila['matricula'];
                }
            }
            else{
                echo "no hi ha resultats.";
            }
            //tancar connexio
            $con->close();
        ?>
    </select>
    <input type="text" id="cmatricula" />
    
    <div id="mmatricula"></div>
</td>
</tr>
<tr>
    <td>data</td>
    <td><input type="text" id="data" size="40" disabled>
        
        
    </td>
</tr>
</table>
<div id="detall">
    <table align="center">
    <tr>
        <td>matricula</td>
        <td><input type="text" id="matricula" size="44" disabled></td>
    </tr>
    <tr>
        <td>marca</td>
        <td><input type="text" id="marca" size="44" disabled></td>

```

```
</tr>
<tr>
  <td>model</td>
  <td><input type="text" id="model" size="44" disabled></td>
</tr>
<tr>
  <td>color</td>
  <td><input type="text" id="color" size="44" disabled></td>
</tr>
<tr>
  <td width="60">dni</td>
  <td><input type="text" id="dni" size="44" disabled></td>
</tr>
<tr>
  <td width="60">nom</td>
  <td><input type="text" id="nom" size="44" disabled></td>
</tr>
<tr>
  <td width="60">direccio</td>
  <td><input type="text" id="direccio" size="44" disabled></td>
</tr>
<tr>
  <td width="60">poblacio</td>
  <td><input type="text" id="poblacio" size="44" disabled></td>
</tr>
<tr>
  <td width="60">sexe</td>
  <td><input type="text" id="sexe" size="44" disabled></td>
</tr>
</table>
</div>

  <center><input type="button" name="boto" value="entrada parking"></center>
</form>
</body>
</html>
```

- E08. Idem a l'anterior retornant les dades en json.

#### 4. Activitats entregables.

Entrega les següents activitats:

- E03
- E04
- E06
- E07 (Sols averiguar els cotxes d'un client)

#### 5. Bibliografia.

- [https://www.w3schools.com/xml/ajax\\_intro.asp](https://www.w3schools.com/xml/ajax_intro.asp)
- <https://www.w3.org/TR/XMLHttpRequest1/>

#### 6. Json

La meua recomanació és crear el JSON manualment per a així comprendre millor el seu funcionament i després poder adaptar-se a una estructura generada per altres persones o funcionalitats. Si saps crear el json sap com es traduirà a variables i podràs accedir als valors sense problemes.

L'únic problema al crear el json es recordar llevar l'última «,» que ens quedarà al crear aquest amb un bucle.

[1, 2, 3,]

Aquest json està mal creat i ens retornara errors al fer el JSON.parse().

Per a eliminar la «,» tenim diverses opcions. Algunes opcions serien:



## Opció 1

```
$cont=0;
$json="[";
while ($fila=$result->fetch_assoc())
{
    $json.=$cont.' ';
    $cont=$cont+1;

}
//cal eliminar l'ultim caràcter. Tenim algo aixi: [1, 2, 3,
$json= substr($json, 0 , -1)
//[1, 2, 3

$json.="]";
```

## Opció 2:

```
$cont=0;
$json.="[";
while ($fila=$result->fetch_assoc())
{
    $json.='{"x":"y","x2":"y2"}';

    if ($result->num_rows -1 > $cont ){// no es ultim , reste un a numero de files
per a que comence des de 0
        $json.=",";
    }
    $cont=$cont+1;

}
$json.="]";
```

Per altra banda tenim opcions que ens generen els json automàticament (bases de dades no relacionals, etc). Des de php podem crear aquesta estructura json automaticament amb `json_encode`.

[https://www.w3schools.com/php/func\\_json\\_encode.asp](https://www.w3schools.com/php/func_json_encode.asp)

```
json_encode($v)

string json_encode ( mixed $value [, int $options = 0 ] )

    $value, valor volem convertir a JSON.
    $options, mascara de bits amb diferents opcions de codificació
```

Per a relacionar aquest amb les nostres consultes a les BD podem utilitzar alguna de les següents opcions:

Opcio 1:

```
if ($result->num_rows >0){//hi ha resultats
    //imprimir els resultats
    while ($fila=$result->fetch_assoc())
    {
        $v[]=$fila;
    }
}
else{
    echo "no hi ha resultats.";
}

$myJSON = json_encode($v);
echo $myJSON;
```

Opció 2:

```
$v = array();
$v = $result->fetch_all(MYSQLI_ASSOC);

//tancar connexio
$con->close();

$myJSON = json_encode($v);
echo $myJSON;
```

Crearà un JSON amb l'estructura més idonea que considere la funció. Cal imprimir aquest i veure el format per a verificar com accedir als valors.

## 7. Aclariments

No intentes copiar exercicis d'altres companys o internet. ni tans sols pegar una miradeta. Si que pots consultar la sintaxis d'una funció, etc. Però no et serveix per a res intentar buscar la solució.

És el major error que pot realitzar qui comença a programar. Sols et deixarà una falsa sensació d'aprenentatge i després no sabràs solucionar els problemes per tu mateix.

Cal intentar que el resultat que genere el vostre codi siga el mes paregut possible a l'exercici (si hi ha imatge d'exemple).

Com els Navegadors no son compatibles entre sí al 100% jo corregire els examens amb FIREFOX. Per tant, es recomanable utilitzar aquest per als nostres exercicis.

Crea uns arxius per a cada exercici o apartat. Així un error a un no afectarà a la resta.

Per a entregar els exercicis crear una carpeta amb el vostre nom i comprimiu aquesta (el nom de l'arxiu comprimit serà elteunom.zip o elteunom.tar.gz). Si entregueu arxius solts o arxius genèrics sense identificar el vostre nom puc confondre'm i no puntuar-vos algun exercici al no saber de qui és.

Cal entregar tots els arxius necessaris per a l'exercici (fotos, llibreries, etc).