

05

Desenvolupament Web en entorn Client

Cicle: Desenvolupament d'aplicacions web

Curs 2025-2026

©José Masip Alonso

1. Introducció

En aquest document aprendrem a utilitzar AJAX amb fetch i un servidor Couchdb.

2. Contingut

2.1. fetch.

La funció fetch en JavaScript és una API moderna per fer peticions HTTP asíncrones des del navegador. És una alternativa més neta i senzilla a XMLHttpRequest.

Exemple hola mon amb un arxiu txt amb fetch

```
<script type="text/javascript">
  function descarregaArxiu() {
    fetch('http://localhost/ruta/holamon.txt')
      .then(resposta => {
        if (!resposta.ok) {
          throw new Error("Error: " + resposta.statusText); // Si hi
ha error HTTP
        }
        return resposta.text(); // Convertim la resposta a text
      })
      .then(contingut => {
        alert(contingut); // Mostrem el contingut rebut
      })
      .catch(error => {
        alert("S'ha produït un error: " + error.message); //
Gestionem errors de xarxa o resposta
      });
  }

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

Explicació de les funcions

<code>fetch(url)</code>	Fa una petició HTTP a la URL especificada. Per defecte és de tipus GET.
<code>.then(response => ...)</code>	Quan la petició es completa, s'executa aquest bloc amb la resposta.
<code>response.ok</code>	Retorna <code>true</code> si l'estat HTTP és OK (200)
<code>response.text()</code>	Converteix la resposta a text. També pot ser <code>json()</code> , <code>blob()</code> , etc.
<code>.catch(error => ...)</code>	Captura errors de xarxa o excepcions llançades en els <code>.then()</code> .

Per a fer la petició asincrònica, és a dir, el codi pot continuar executant-se mentre espera la resposta del servidor, cal afegir `async`.

```
async function obtenirDades() { //Declara funció asíncrona. Això permet fer servir await dins seu.
  try {
    const resposta = await fetch('https://onsiga'); //Fa la petició HTTP i espera la resposta sense bloquejar el fil principal.

    if (!resposta.ok) {
      throw new Error('Error HTTP: ' + resposta.status);
    }

    const dades = await resposta.text ();
    alert('Dades rebudes:', dades);
  } catch (error) {
    console.error('Error en la petició:', error.message);
  }
}

obtenirDades();
```

2.2. CouchDB.

CouchDB és una base de dades NoSQL documental dissenyada per ser simple, robusta i accessible via HTTP. Guarda les dades en format JSON, permet consultes amb JavaScript, i és ideal per a aplicacions web i mòbils que necessiten sincronització i replicació.

Característiques principals:

- Documents JSON: Cada registre és un document amb estructura flexible.
- Accés via HTTP/REST: Pots fer CRUD amb GET, POST, PUT, DELETE.
- Replicació automàtica: Ideal per sincronitzar dades entre servidors o dispositius.
- Vistes amb JavaScript: Pots crear consultes personalitzades amb map/reduce.
- Gestió d'usuaris i rols: Per controlar l'accés a les dades.
- Conflictes gestionats: CouchDB permet múltiples versions d'un document i resol conflictes.

Exemple de document:

```
{
  "_id": "usuari",
  "nom": "Jose Luis Maria de la Fuente",
  "edat": 35,
  "professió": "Desenvolupador",
  "habilitats": ["CouchDB", "Python", "JavaScript"]
}
```

Abans de continuar amb CouchDB expliquem la base de json.

2.3. 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 «,»).

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

Un exemple més elaborat seria aquest vector d'objectes:

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

Altre exemple més 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" },  
]}
```

2.4. Docker CouchDB

Per a crear el docker de CouchDB executarem:

```
docker run -d --name couchdb -p 5984:5984 -e COUCHDB_USER=admin -e  
COUCHDB_PASSWORD=admin couchdb:latest
```

Amb aquesta instrucció descarreguem i creem l'última versió del docker oficial de couchdb creant un usuari admin(contrasenya «admin») per al seu ús.

Una volta descarregat i creat simplement arrancarem el docker amb:

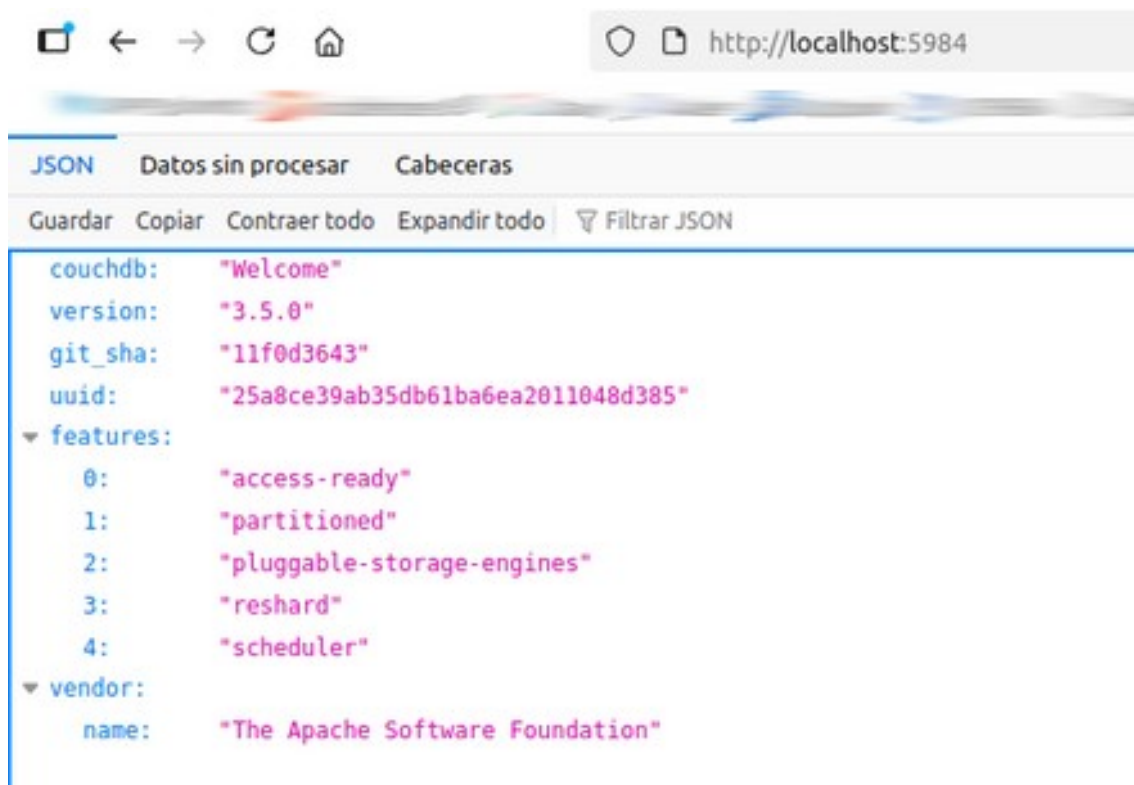
```
docker start couchdb
```

Per a parar el contenidor docker cal fer:

```
docker stop couchdb
```

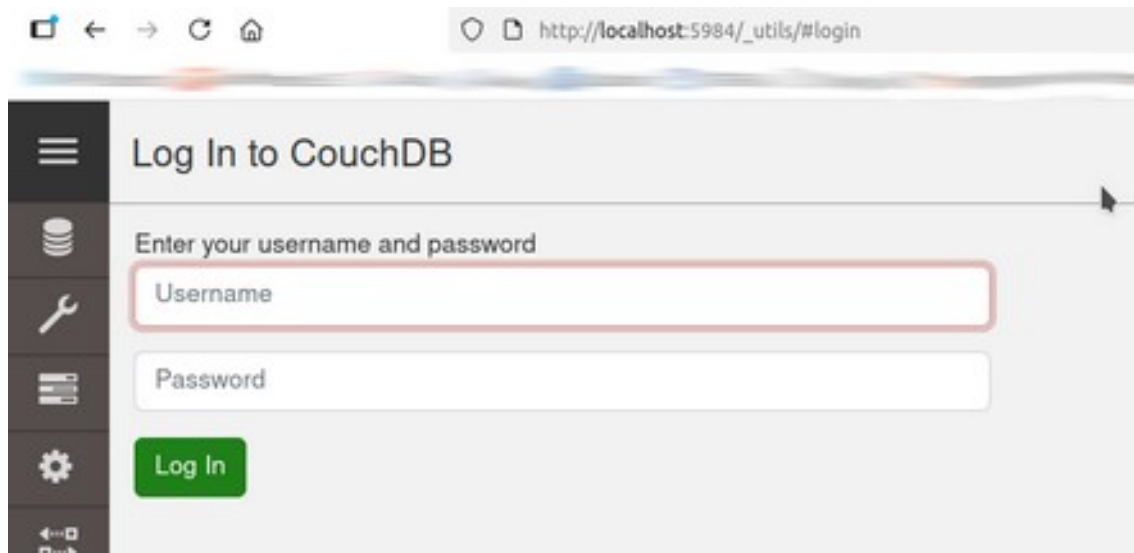
Cal esperar un poc per a que arranque completament. Per a verificar funciona escriure al navegador

```
http://localhost:5984/
```



2.5. Preparar dades exemple

Quan escrivim al navegador `http://localhost:5984/_utils` ens demanarà un usuari i una contrasenya (hem indicat aquestes crear el docker)

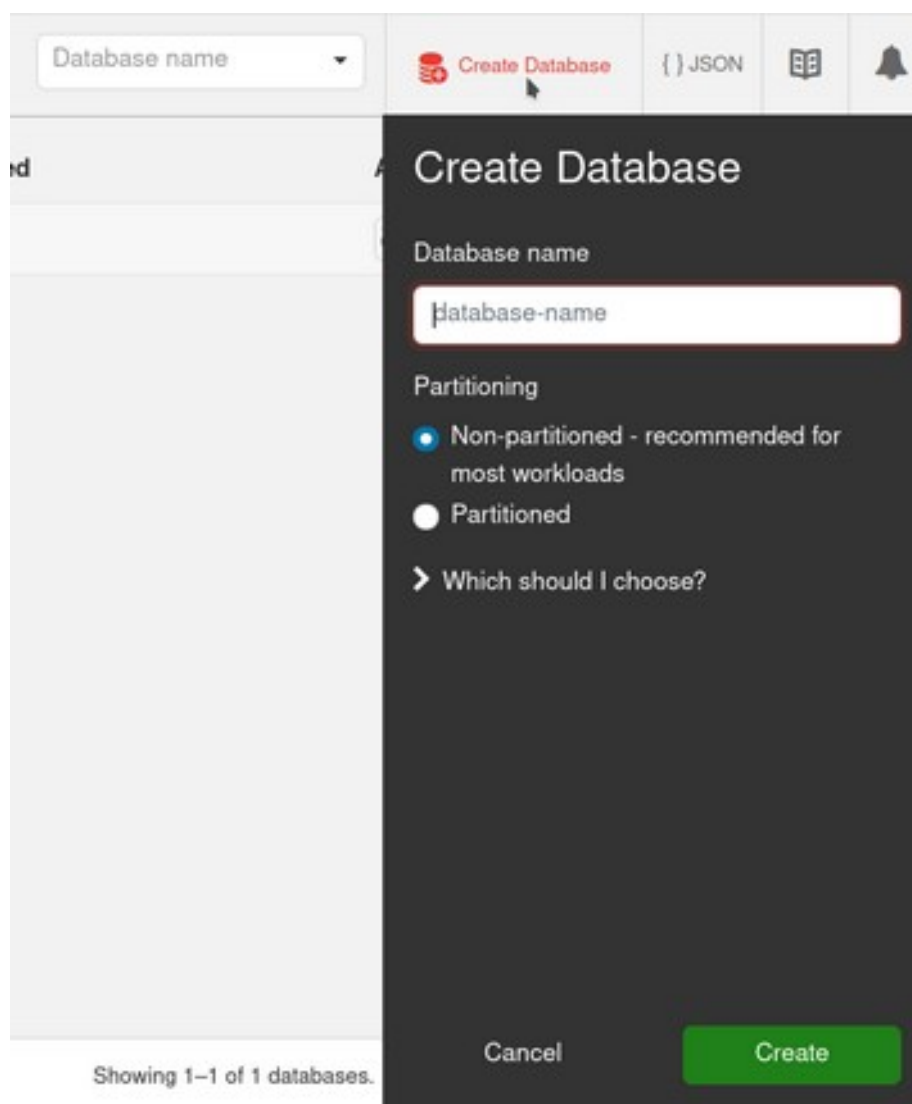


Escriurem «admin» com a nom d'usuari i «admin» com a contrasenya.

Al iniciar sessió podem observar les bases de dades existents i crear de noves.



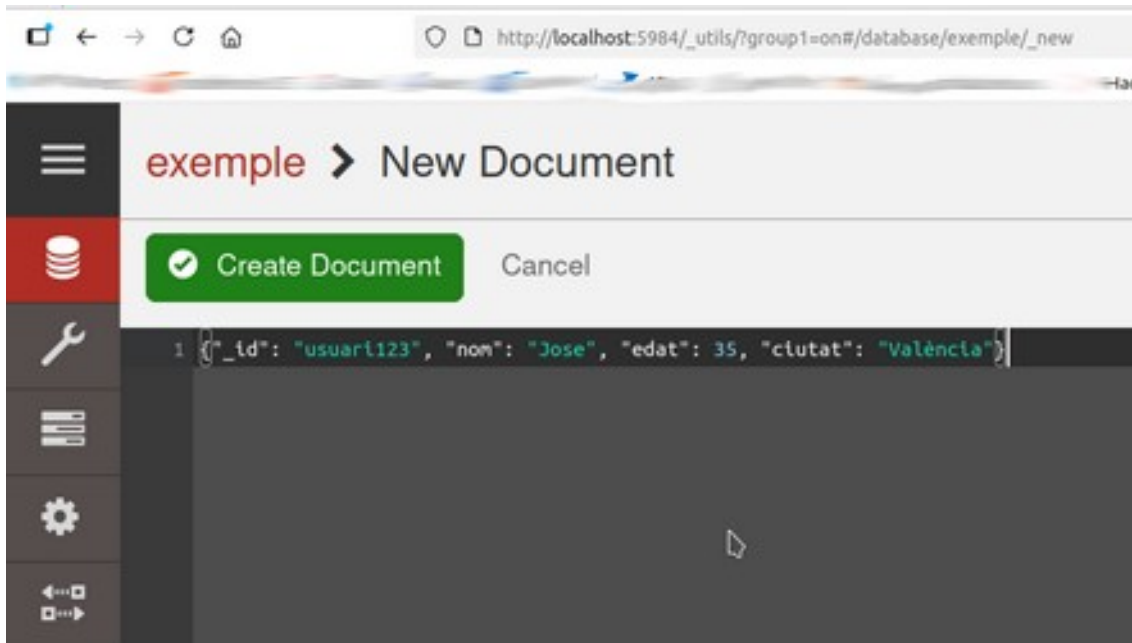
Per a crear una Base de dades nova cal polsar «Create database».



Escrivim «exemple» i polsem «Create» i ens mostrarà.



Si polsem ara damunt de la base de dades «exemple» podrem veure els documents (objectes json amb un id) existents o afegir de nous. Polsem en «New document».



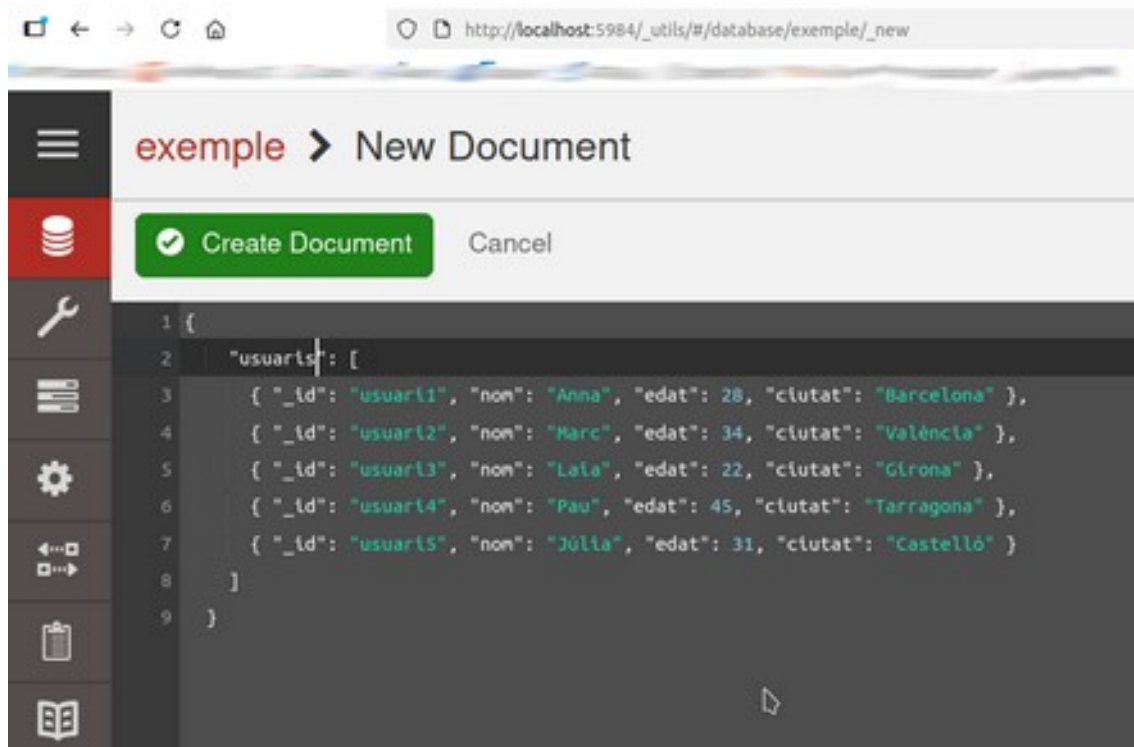
Posem un objecte json i polsem «Create Document». Per exemple:

```
{
  "_id": "usuari123",
  "nom": "jose",
  "edat": 35,
  "ciutat": "Valencia"
}
```

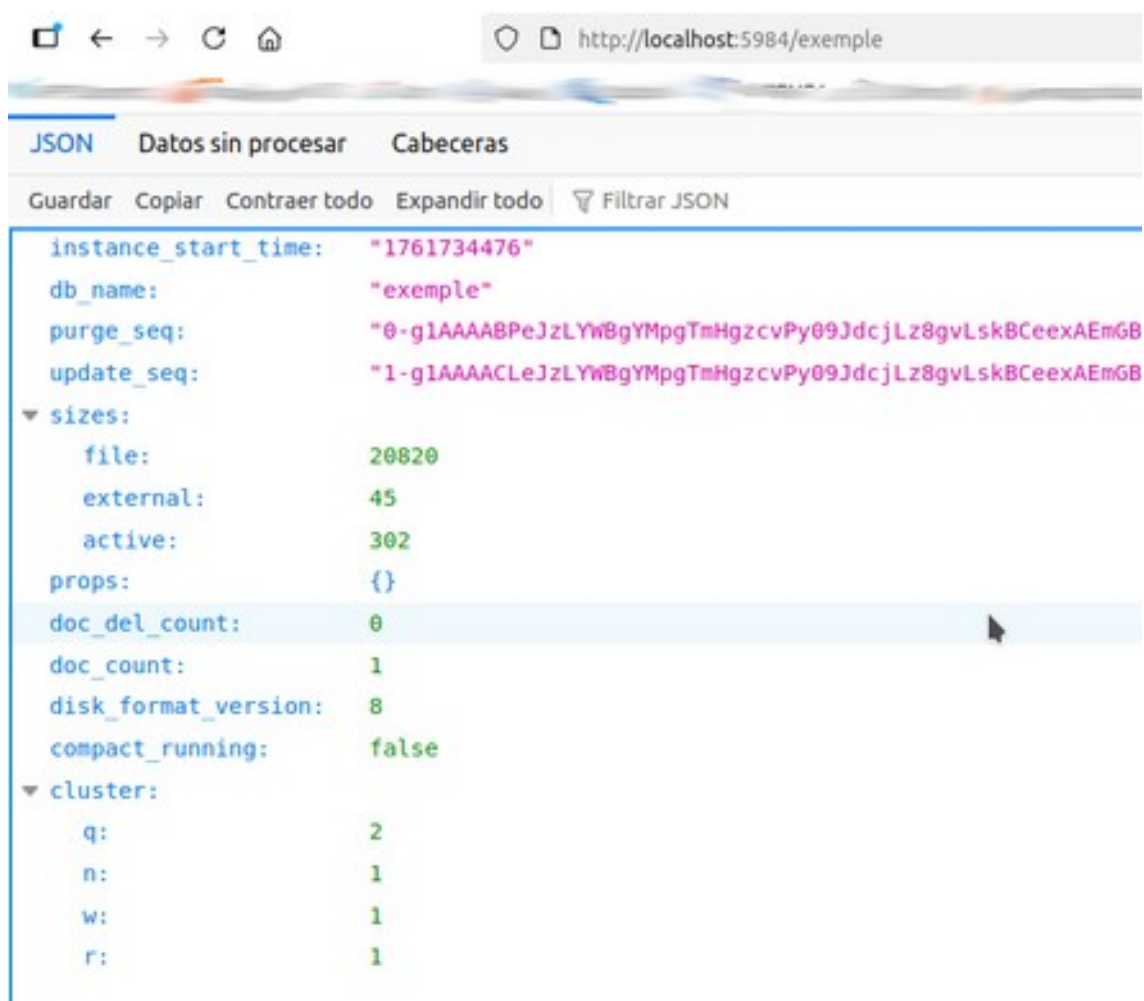


Dades exemple on creem un vector d'usuaris:

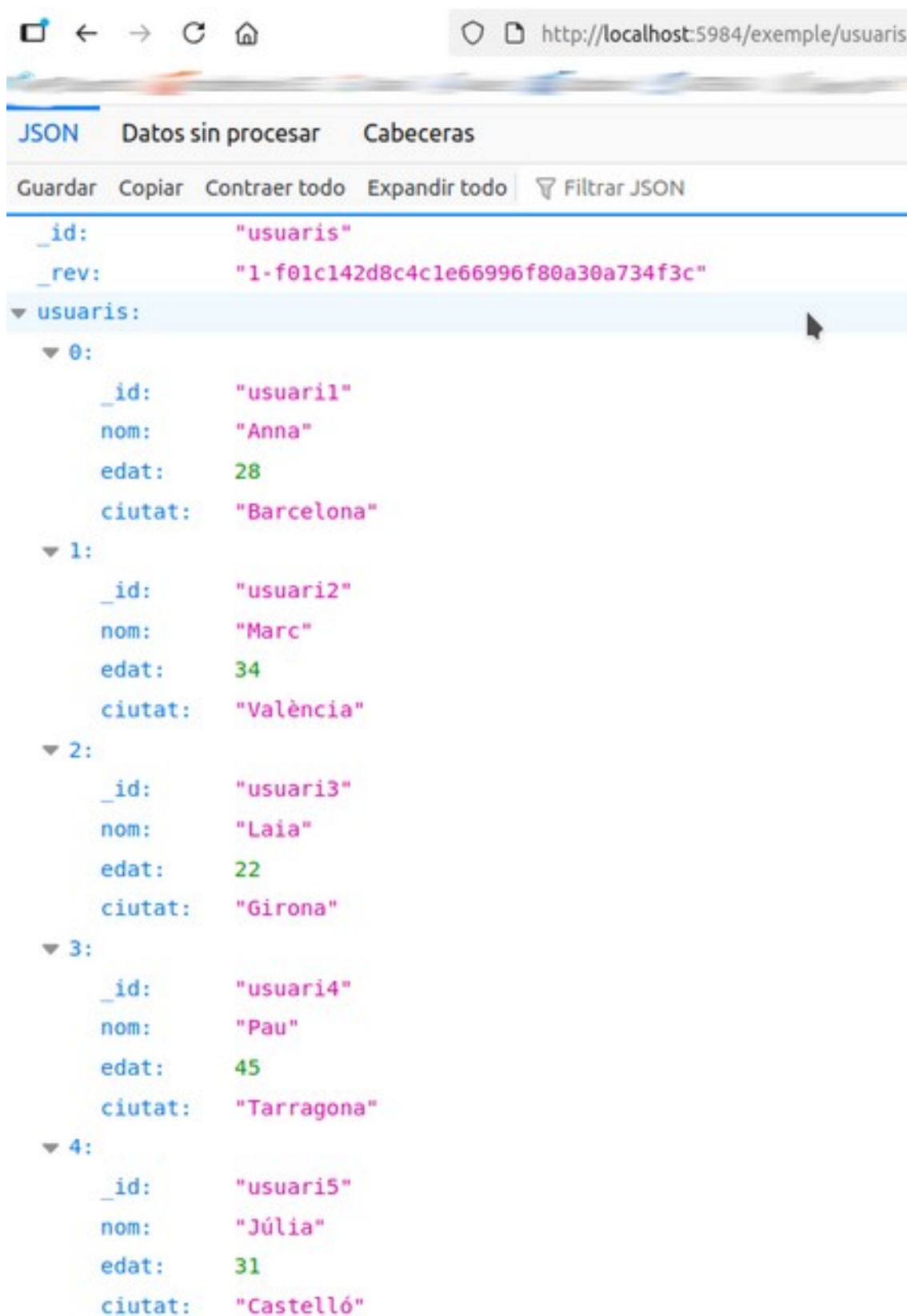
```
{
  "_id": "usuaris",
  "_rev": "1-f01c142d8c4c1e66996f80a30a734f3c",
  "usuaris": [
    {
      "_id": "usuari1",
      "nom": "Anna",
      "edat": 28,
      "ciutat": "Algemesi"
    },
    {
      "_id": "usuari2",
      "nom": "Marc",
      "edat": 34,
      "ciutat": "Alzira"
    },
    {
      "_id": "usuari3",
      "nom": "Laia",
      "edat": 22,
      "ciutat": "Alcudia"
    },
    {
      "_id": "usuari4",
      "nom": "Pau",
      "edat": 45,
      "ciutat": "Alzira"
    },
    {
      "_id": "usuari5",
      "nom": "Júlia",
      "edat": 31,
      "ciutat": "Algemesi"
    }
  ]
}
```



Per a accedir a les dades ho podem fer a través de la URL del navegador. Per a accedir a la base de dades simplement escrivim darrere del servidor i el port el nom de la base de dades.



Per a accedir a un document possarem després del nom de la base de dades el nom del document.



JSON Datos sin procesar Cabeceras

Guardar Copiar Contraer todo Expandir todo 🔍 Filtrar JSON

```
{
  "_id": "usuaris",
  "_rev": "1-f01c142d8c4cle66996f80a30a734f3c",
  "usuaris": [
    {
      "_id": "usuari1",
      "nom": "Anna",
      "edat": 28,
      "ciutat": "Barcelona"
    },
    {
      "_id": "usuari2",
      "nom": "Marc",
      "edat": 34,
      "ciutat": "València"
    },
    {
      "_id": "usuari3",
      "nom": "Laia",
      "edat": 22,
      "ciutat": "Girona"
    },
    {
      "_id": "usuari4",
      "nom": "Pau",
      "edat": 45,
      "ciutat": "Tarragona"
    },
    {
      "_id": "usuari5",
      "nom": "Júlia",
      "edat": 31,
      "ciutat": "Castelló"
    }
  ]
}
```

2.6. Primera consulta a couchdb

Anem a fer una petició al servidor per a obtenir les dades del document usuari123.

Crea un document couchdb01.html amb el següent contingut:

```
<!DOCTYPE html>
<html lang="ca">
<head>
  <meta charset="UTF-8">
  <title>Consulta CouchDB</title>
</head>
<body>
  <h2>Consulta couchdb</h2>
  <p>

    <iframe width="560" height="315"
src="https://www.youtube.com/embed/xru5FreD97w?si=iExrnm6bffflraJEi"
title="YouTube video player" frameborder="0" allow="accelerometer; autoplay;
clipboard-write; encrypted-media; gyroscope; picture-in-picture; web-share"
referrerpolicy="strict-origin-when-cross-origin" allowfullscreen></iframe>
  </p>
  <button onclick="obtenirUsuari()">Consultar</button>
  <pre id="resultat"></pre>

  <script>
    async function obtenirUsuari() {

      const usuari = 'admin';
      const contrasenya = 'admin';
      const credencials = btoa(`${usuari}:${contrasenya}`); //base64

      try {
        const resposta = await fetch('http://localhost:5984/exemple/usuari123', {
          headers: {
            'Authorization': `Basic ${credencials}`
          }
        });

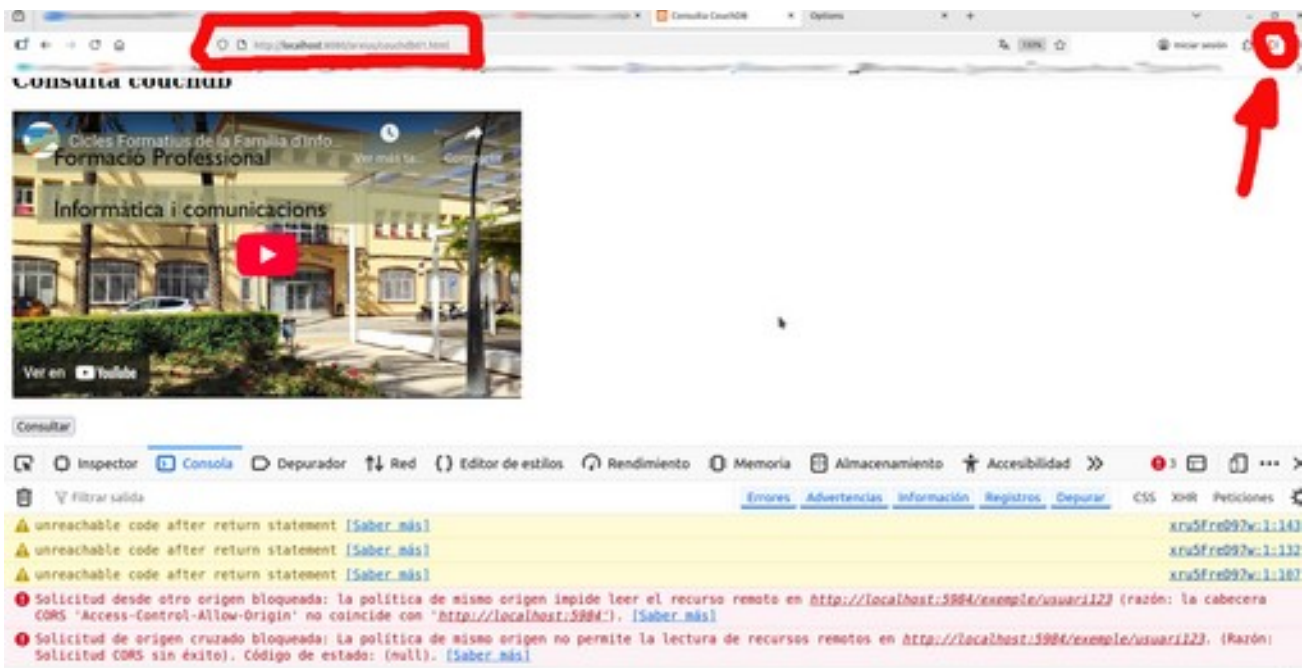
        if (!resposta.ok) {
          throw new Error(`Error al obtenir el document: ${resposta.status}`);
        }

        const dades = await resposta.json();
        document.getElementById('resultat').textContent = JSON.stringify(dades,
null, 2);
      } catch (error) {
        document.getElementById('resultat').textContent = 'Error: ' +
error.message;
      }
    }
  </script>
```

```
</body>  
</html>
```

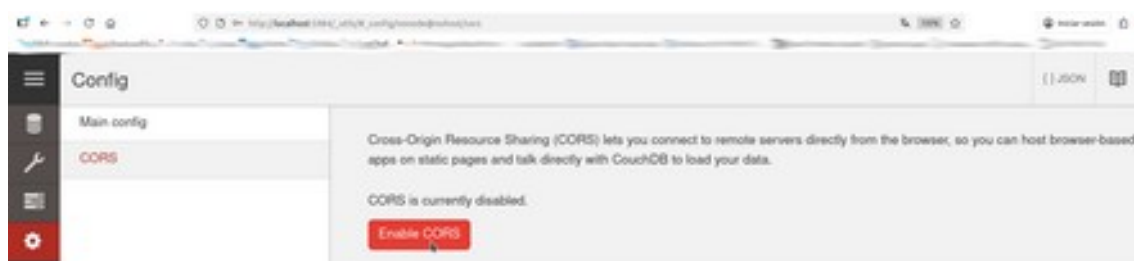
A aquest document accedim al couchdb per a sol·licitar les dades de usuari123 amb la consulta ajax a **http://localhost:5984/exemple/usuari123**.

Recorda passar el document dins del htdocs del xampp (docker o instal·lat al teu equip).



Com podem observar a la captura, cal activar l'extensió del firefox per a permetre CORS.

També cal configurar couchdb per a permetre les peticions CORS. Per a tal fi, polsa en «Configuració», després en CORS i polsa en «enable CORS».

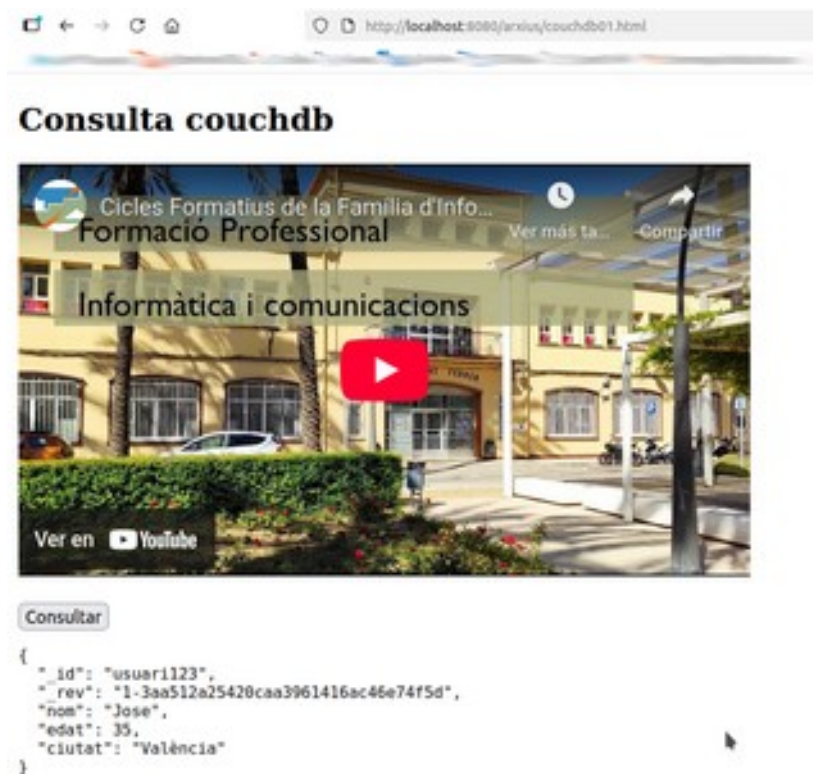


Una volta habilitat cal marcar que ens accepti tots els dominis (Recordar que no són les millors opcions de configuració però són les que vos generaran menys problemes a vosaltres).

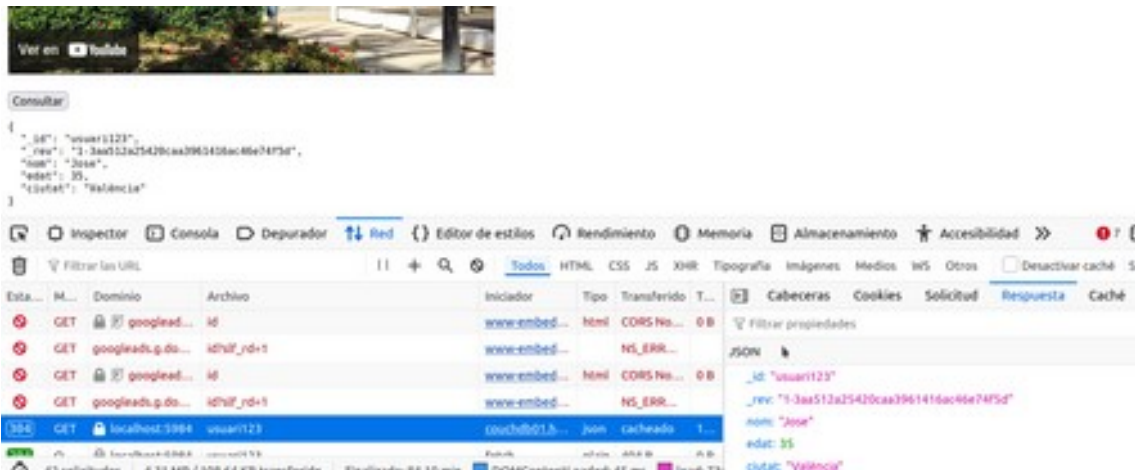


Para i arranca el docker de couchdb per a assegurar-te de que s'aplica correctament la configuració.

Al refrescar la pàgina i polsar sobre el boto obtenim el json com a text.



Podem observar com hem rebut el json a la pestanya «Red» del navegador Firefox



2.7. Formatar o treballar el json.

Una volta rebem el json

```
const dades = await resposta.json();
```

Tenim una constant `dades` on tenim els valors del json. Per a accedir a les distintes parts simplement cal fer:

```
dades.nom
```

Si substituïm al codi anterior per a formatar les dades amb titol 1 i paragrafs obtindrem:

```
const dades = await resposta.json();
document.getElementById('resultat').innerHTML =
"<h1>"+dades.nom+"</h1><p>"+dades.edat+"</p><p>"+dades.ciutat+"</p>";
```

Com a resultat genera:



Consultar

Jose

35

València

3. Activitats.

- E01. Tenim una estructura json usuaris que representa un vector d'usuaris. Realitzar una connexió ajax que mostre el contingut a la capa. Mostra el nom en un titol 2 i la edat i la ciutat en un paragraf separats per «-+-+».
- E02. Donat el json que es mostra a continuació de la cdteca. Formata aquest en una taula per a cadascun dels cd's. Obtin les dades a traves d'una consulta ajax.

```
{
  "_id": "cdteca",
  "CATALEG": {
    "CD": [
      {
        "TITOL": "cantar",
        "ARTISTA": "tamara",
        "PAIS": "espanya",
        "PREU": 2,
        "ANY": 2010
      },
      {
        "TITOL": "manolo",
        "ARTISTA": "manolo
escobar",
        "PAIS": "espanya",
        "PREU": 5,
        "ANY": 1985
      },
      {
        "TITOL": "cd",
        "ARTISTA": "juanito",
        "PAIS": "francia",
        "PREU": 23,
        "ANY": 2012
      },
      {
        "TITOL": "cd2",
        "ARTISTA": "juanito",
        "PAIS": "francia",
        "PREU": 32,
        "ANY": 2013
      },
      {
        "TITOL": "recopilatori",
        "ARTISTA": "dj",
        "PAIS": "espanya",
        "PREU": 4,
        "ANY": 2008
      }
    ]
  }
}
```

capa

cdclub

caratula cd	cantar
	tamara
	espanya
	2 euros
	2010

caratula cd	manolo
	manolo escobar
	espanya
	5 euros
	1985

caratula cd	cd
	juanito
	francia
	23 euros
	2012

caratula cd	cd2
	juanito
	francia
	32 euros
	2013

caratula cd	recopilatori
	dj
	espanya
	4 euros
	2008

- E03. Tenim un json on guardem les notícies. De les notícies guardem les dades com es mostra a la taula següent. Cal formatar mostrant el nom de l'usuari en un h1, a continuació la data en un h3 i el comentari en un paràgraf. Obtindre les dades en una consulta AJAX.

```
{
  "_id": "noticies",
  "_rev": "1-486a3ddc8d4bd06e5e368affb39b275c",
  "noticies": [
    {
      "data": "27-12-2012",
      "usuari": "juanito",
      "comentari": "comentari de la noticia"
    },
    {
      "data": "1-12-2012",
      "usuari": "manuel",
      "comentari": "asdf asdf asdf asdf asdf asdf asdf asdf asdf"
    },
    {
      "data": "11-11-2010",
      "usuari": "juanito",
      "comentari": "asdf jklñ jklñ jklñ jklñ jklñ jklñ jklñ"
    }
  ]
}
```

- E04. Volem crear una pàgina que simule la gestió d'un parking. Per a tal fi, crear una base de dades parking i dins dos documents:
 - Clients: Usuaris registrats al parking.
 - Cotxes: Cotxes dels clients identificats i assignats a un usuari .

```
{
  "_id": "parking_clients",
  "clients": [
    {
      "dni": "20202020a",
      "nom": "jose",
      "via": "carrer",
      "ciutat": "madrid",
      "sexe": "home"
    },
    {
```

```
{
  "_id": "parking_cotxes",
  "cotxes": [
    {
      "matricula": "1111asd",
      "idclient": "21212121a",
      "marca": "ford",
      "model": "ka",
      "color": "roig"
    },
    {
```

<pre> "dni": "21212121a", "nom": "luis", "via": "avenida", "ciutat": "valencia", "sexe": "home" }, { "dni": "22222222a", "nom": "juan", "via": "plaça", "ciutat": "castello", "sexe": "home" }, { "dni": "23232323a", "nom": "luis", "via": "avenida", "ciutat": "barcelona", "sexe": "home" }, { "dni": "24242424a", "nom": "juana", "via": "plaça", "ciutat": "castello", "sexe": "dona" }, { "dni": "25252525a", "nom": "luisa", "via": "avenida", "ciutat": "alicant", "sexe": "dona" }] } </pre>	<pre> "matricula": "2345tre", "idclient": "21212121a", "marca": "ford", "model": "focus", "color": "negre" }, { "matricula": "6545try", "idclient": "22222222a", "marca": "ford", "model": "kuga", "color": "negre" }, { "matricula": "6754gfg", "idclient": "25252525a", "marca": "ford", "model": "fiesta", "color": "roig" }, { "matricula": "7867hgy", "idclient": "23232323a", "marca": "ford", "model": "fiesta", "color": "blau" }] } </pre>
--	--

- Quan carreguem la pàgina omplirem el select de clients amb el dni del client a traves d'una consulta ajax.
- Quan carreguem la pàgina omplirem el select de cotxes amb tots els cotxes a traves d'una consulta ajax.

- Al seleccionar un client actualitzarem les matricules del select per a sols mostrar les d'aquest usuari.
- Al seleccionar una matricula omplirem els detalls del cotxe

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="a.js"></script>
</head>
<body>
  <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
          </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
          </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>
```

4. **Activitats entregables.**

Entrega les següents activitats:

- E03
- E04

5. **Bibliografia.**

- 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

```
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 comprimi 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).