

Exercici 1 – Implementació AJAX i propietats de l'objecte XMLHttpRequest

Seguint els passos que tens a continuació, crea una pàgina com aquesta a on:

- En carregar la pàgina, el quadre de text ha de mostrar per defecte la URL de la pròpia pàgina.
- En prémer el botó "Mostrar Continguts", s'ha de descarregar mitjançant peticions AJAX el contingut corresponent a la URL introduïda per l'usuari. El contingut de la resposta rebuda del servidor s'ha de mostrar a la zona de "Continguts del fitxer". (Pas 1: desencadena la petició)
- A la zona "Estats de la petició" s'ha de mostrar en tot moment l'estat en què es troba la petició (No inicialitzat anteriorment, carregant, completada, etc.) (Pas 4 i 5)

Opcional:

- Mostra el contingut de totes les capçaleres de la resposta del servidor a la zona "Capçaleres HTTP de la resposta del servidor".
- Mostra el codi i text d'estat de la resposta del servidor a la zona "Codi d'estat".

URL:

Mostrar contenidos

Contenidos del archivo:

Estados de la petición:

Código de estado:

Cabeceras HTTP de la respuesta del servidor:

Pas 1: "Event" desencadena la petició

Crea un document HTML que contingui un botó, el qual serà l'encarregat d'iniciar la petició i un àrea a on es mostraran les dades rebudes del servidor de forma asíncrona.

Pas 2: Creació de l'objecte XMLHttpRequest

Dins de l'etiqueta <head>, defineix un script amb una variable global que contingui la referència a l'objecte XMLHttpRequest.

```
var peticio_http;
```

Tingues en compte que ha d'estar preparat per funcionar en navegadors obsolets (.ActiveXObject). Pots fer servir el exemples explicats a la teoria.

Pas 3: Creació de la petició al servidor

Crea la petició d'informació al servidor, fent servir els mètodes open i send de l'objecte XMLHttpRequest.

```
open(method, url, [async],[user],[password]);
```

- method: Mètode d'enviament. Valors possible: POST, GET, PUT, DELETE.
- url: Adreça url a on es realitza la petició.
- async: Determina si la petició es realitzarà de forma síncrona o asíncrona. Valors possibles: true o false. True predeterminat.
- user: identificador d'usuari. Aquesta i la següent només quan es necessaria l'acreditació de credencials.
- password

```
send(dades);
```

- dades: No sempre s'envien paràmetres, en aquest cas es farà servir la paraula reservada null.

Pas 4: Resposta del servidor

Defineix el mecanisme que et permetrà detectar la recepció de la resposta del servidor. Amb aquest mecanisme hauràs de manegar la resposta, conèixer l'estat de la petició mitjançant l'event onreadystatechange associat a l'objecte XMLHttpRequest, que es dispara cada cop que s'actualitza el valor de la propietat readyState del mateix objecte.

```
peticio_http.addEventListener('readystatechange', gestionarResposta);
```

o

```
peticio.http.onreadystatechange = function(){};
```

NOTA: **gestionarResposta** és la funció encarregada de manegar l'event.

Pas 5: Tractament de la informació rebuda

El fet de rebre una resposta per part del servidor no vol dir que la petició s'hagi portat a terme de forma correcta.

Fes servir les propietats readyState i status de l'objecte XMLHttpRequest per comprovar la validesa de la resposta del servidor, tenint en compte que el valor 4 de la propietat readyState indica que l'operació s'ha portat a terme completament i que el valor 200 de la propietat status indica que la petició s'ha portat a terme de forma satisfactòria.

Fes servir la documentació adequada per saber més d'aquests mètodes:

- XMLHttpRequest.[readyState](#)
- XMLHttpRequest.[status](#). [Valors](#) de la propietat status.