

# Gestionando las respuestas del servidor

Hemos realizado la solicitud, nuestro cliente está trabajando en el formulario (mientras el servidor gestiona la respuesta), y ahora el servidor termina la ejecución.

El servidor entonces comprueba la propiedad **onreadystatechange** y ve cual es el **método que tiene que invocar**.

Una vez que ésto ocurre, podemos pensar en nuestra aplicación como otra aplicación asíncrona o no.

En otras palabras, no tenemos que realizar ninguna acción especial para escribir métodos que respondan a esa respuesta proveniente del servidor; simplemente modificaremos el formulario, o llevaremos al cliente a otra dirección URL o haremos lo que nos haga falta según la respuesta del servidor.

En este caso programaremos que se cambia una parte del formulario.

## Respuestas y Ajax

Hemos visto cómo indicarle al servidor qué hacer cuando termina la solicitud: **hemos puesto en la propiedad onreadystatechange del objeto XMLHttpRequest la función que debe ejecutar cuando termine de buscar la respuesta**.

Entonces cuando el servidor ha procesado la respuesta, automáticamente llamará a esa función.

Véase el listado 11.

### Listado 11. Código para gestionar la respuesta

```
<script language="javascript" type="text/javascript">
    var request = false;
    try {
        request = new XMLHttpRequest();
    } catch (trymicrosoft) {
        try {
            request = new ActiveXObject("Msxml2.XMLHTTP");
        } catch (othermicrosoft) {
            try {
                request = new ActiveXObject("Microsoft.XMLHTTP");
            } catch (failed) {
                request = false;
            }
        }
    }

    if (!request)
        alert("Error inicializando XMLHttpRequest!");

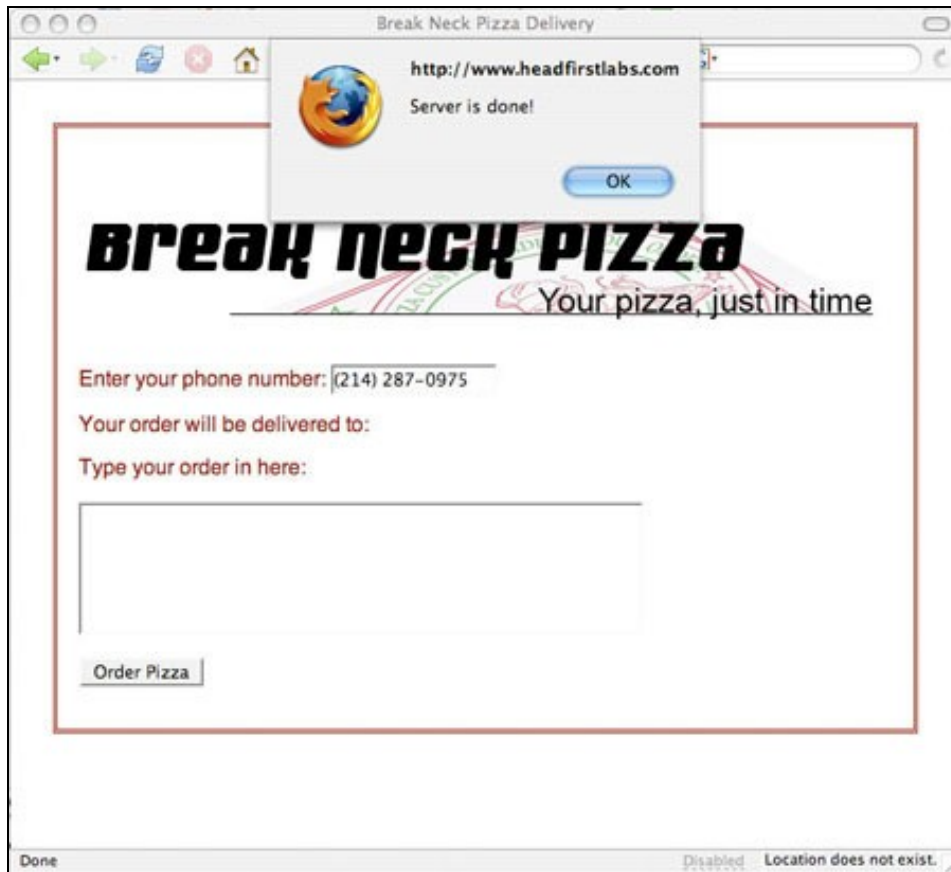
    function obtenerInfoCliente() {
        var phone = document.getElementById("phone").value;
        var url = "buscarcliente.php?phone=" + escape(phone);
        request.open("GET", url, true);
        request.onreadystatechange = actualizarPagina;
        request.send(null);
    }

    function actualizarPagina() {
        alert("El servidor ha terminado la consulta en la pagina php!");
    }
</script>
```

En este caso simplemente cuando termina la ejecución de dicha página php nos informará mediante una alerta.

Si ejecutamos este código veremos que cuando introducimos un teléfono y nos vamos a otro campo nos aparecerá ese mensaje de alerta una y otra vez.

Figura 3. Código mostrando la alerta.



Dependiendo del navegador, obtendremos 2, 3 o incluso 4 alertas.

¿Entonces, qué está ocurriendo? Pues tiene que ver con el ciclo de solicitud/respuesta de HTTP.