HTML5.

Envia formulario <form id=”formulario” a php

Html

<form id =”formulario” action=”recibir.php” method=”post”>

<input type="submit" id="btnEnviar" name="btnEnviar" value="Enviar formulario">

js file

$(document).ready(function () {

$("#formulario").bind("submit",function(){

// Capturamnos el boton de envío

var btnEnviar = $("#btnEnviar");

$.ajax({

type: $(this).attr("method"),

url: $(this).attr("action"),

data:$(this).serialize(),

data:$(this).serialize(),

data:$(#formulario).serialize(),

php

echo "Hola ".$\_REQUEST["nombre"].", Tienes ".$\_REQUEST["edad"]." Años.";

<html lang="es">

<head>

<meta charset="UTF-8">

<title>Enviar formulario con ajaxtitle>

<head>

<body>

<form id="formulario" action="recibir.php" method="post">

<input type="text" placeholder="Escribe tu nombre"

name="nombre" required autofocus title="Ingresa tu nombre">

<input type="number" placeholder="Ingresa tu edad"

name="edad" required title="Ingresa tu edad">

<br><br>

<input type="submit" id="btnEnviar" name="btnEnviar" value="Enviar formulario">

form>

<hr>

<p class="respuesta">

<p>

<script src="https://code.jquery.com/jquery-2.2.2.min.js">script>

<script src="js/enviar.js">script>

<body>

En el archivo html vemos que tenemos un formulario con dos campos (Nombre, Edad) Los cuales serán enviados por método POST.

2. Archivo \*.js

Para realizar el envío por medio de AJAX en este caso vamos a hacer uso de la librería Jquery, para lo cual insertamos el script haciendo referencia al jquery por encima de nuestro código o archivo en donde hacemos el ajax; Esto es importante de lo contrario vamos a tener problemas y al final no va a funcionar nuestro código.

<script src="https://code.jquery.com/jquery-2.2.2.min.js">script>

<script src="js/enviar.js">script>

Ahora vamos a ver nuestro código para realizar el envío:

$(document).ready(function () {

$("#formulario").bind("submit",function(){

// Capturamnos el boton de envío

var btnEnviar = $("#btnEnviar");

$.ajax({

type: $(this).attr("method"),

url: $(this).attr("action"),

data:$(this).serialize(),

beforeSend: function(){

/\*

\* Esta función se ejecuta durante el envió de la petición al

\* servidor.

\* \*/

// btnEnviar.text("Enviando"); Para button

btnEnviar.val("Enviando"); // Para input de tipo button

btnEnviar.attr("disabled","disabled");

},

complete:function(data){

/\*

\* Se ejecuta al termino de la petición

\* \*/

btnEnviar.val("Enviar formulario");

btnEnviar.removeAttr("disabled");

},

success: function(data){

/\*

\* Se ejecuta cuando termina la petición y esta ha sido

\* correcta

\* \*/

$(".respuesta").html(data);

},

error: function(data){

/\*

\* Se ejecuta si la peticón ha sido erronea

\* \*/

alert("Problemas al tratar de enviar el formulario");

}

});

// Nos permite cancelar el envio del formulario

return false;

});

});

El método bind() se utiliza para conectar un controlador de eventos directamente a los elementos, es decir nosotros podemos invocar a un elemento $(".boton") y por medio del método bind() ejecutar uno o mas eventos sobre sí.

También dentro de este método bind() usamos un return false , esto sirve para cancelar el envió directo del formulario hacia el archivo php, dado que el ajax se ejecuta sobre la misma pantalla sin refrescarla.

3. El archivo \*.php

echo "Hola ".$\_REQUEST["nombre"].", Tienes ".$\_REQUEST["edad"]." Años.";

Al final debemos encontrar un resultado como este cuando hagamos el envió del formulario.