



BASE DE DATOS EN PHP

DESARROLLO WEB EN ENTORNO  
SERVIDOR

Feedback unidad 4

David Herrero Estévez

## Contenido

Enunciado.....	3
Listado de archivos y su contenido .....	3
Descripción del fichero index.php.....	5
Alta.php .....	5
Descripción de la página de alta. ....	6
Listado.php .....	7
Página Modificar.php .....	9
Descripción modificar.php .....	11
Página de conexiones.php.....	12
Descripción conexiones.php .....	15
Ejemplo de funcionamiento .....	16
Si hacemos clic en Alta .....	17
Introducimos un valor duplicado. Nos devuelve un error .....	17
En caso de introducir valores únicos.....	18
En caso de introducir combinación de ciudad + país que ya existe. ....	18
Si hacemos clic en el enlace del listado.....	18
Si buscamos todas las ciudades de España (Spain) .....	19
Si hacemos clic en el icono de Modificar ciudad.....	19
Si cambiamos el país y damos a submit .....	20
Si hacemos clic en el botón de eliminar .....	20

## Enunciado

Realizar un programa PHP Crud Create, Read, Update, Select de la tabla city de Sakila.

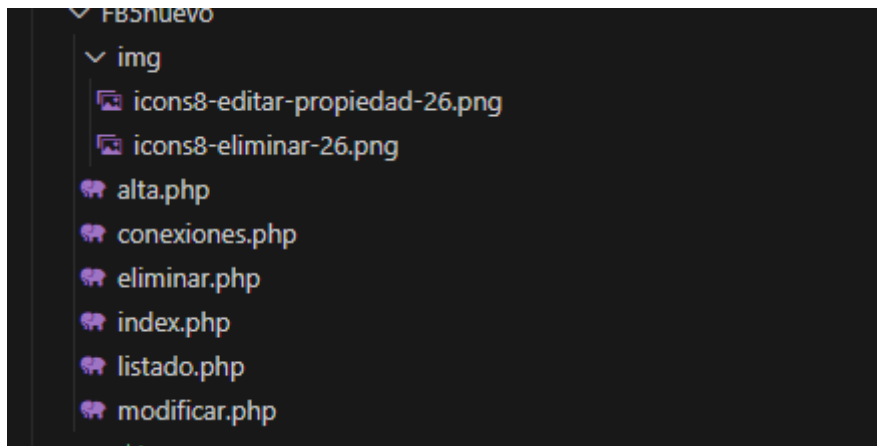
Obligatorio: usar mas de un formulario NOTA: Realizarlo en tecnología PDO.

Nota:

En la clase de hoy has comentado que puede ser de Country o de cualquier otra tabla.

Yo he elegido city (porque me he equivocado más que nada).

Estructura de archivos



Los archivos se encuentran situados en una carpeta dentro del directorio “DocumentRoot” de Xampp.

Estructura del programa

El programa se compone del fichero index.php que a su vez va llamando (con includes) a los diferentes módulos, como alta, eliminar, listado y modificar. También cuenta con una página php conexiones, donde se encuentran todas las funciones para la interacción con la base de datos.

## Listado de archivos y su contenido

Index.php

```
<!DOCTYPE html>
<html lang="en">
```

```

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Document</title>
  <style>
    .tabla {
      border: 1px solid black;
    }

    .tabla tr:nth-child(odd) {
      background-color: #f2f2f2;
    }

    .error {
      color: red;
    }

    .enunciado {
      margin: auto;
      font-family: Arial, sans-serif;
      font-size: medium;
      background-color: lightyellow;
    }

    ul {
      display: flex;
    }

    li {
      display: inline-block;
      margin-right: 50px;
    }
  </style>
</head>

<body>
  <div class="enunciado">
    Realizar un programa PHP
    Crud Create, Read, Update, Select de la tabla Country de
    Sakila.
    Obligatorio: usar mas de un formulario
    NOTA: Realizarlo en tecnologia PDO.
  </div>

  <div class="menu">
    <ul>
      <li> <a href="<?=$ _SERVER
['PHP_SELF']?>?opc=alta">Alta</a></li>
      <li> <a href="<?=$ _SERVER
['PHP_SELF']?>?opc=listado">Listado</a></li>
    </ul>
  </div>
  <div class="principal">
    <?php
    $opc=isset($_REQUEST['opc'])?$_REQUEST['opc']:"";

    switch($opc){
      case "alta":

```

```

        include_once("alta.php");
        break;
        case "listado":
            include_once("listado.php");
            break;
        case "modificar":
            include_once("modificar.php");
            break;
        case "eliminar":
            include_once("eliminar.php");
            break;
        default:
            include_once("listado.php");
    }
    ?>
</div>
</body>

</html>

```

## Descripción del fichero index.php

Este fichero cuenta primeramente con un listado de opciones para el usuario:

Listar y Alta.

Si pulsa en Listar hay un enlace para la propia página pasándole una opción de navegación :  
opc = listar

Si pulsar sobre alta, la opción de navegación es "Alta".

Tras el menú de la página, hay un switch que se encarga de mostrar el contenido elegido por el usuario.

Las diferentes opciones son :

Alta, listado, modificar y eliminar.

En caso de cualquier otra opción, se mostrará la página de listado.

## Alta.php

```

<?php
include_once("conexiones.php");
conecta();
$países=cargaPaíses();
?>
<form action="" method="post">
    <input type="hidden" value="opc" value="alta">
    <table>
        <tr>

```

```

        <td>Código ciudad</td><td><input type="number"
name="city_id" id="city_id" maxlength="3" required></td>
    </tr>
    <tr>
        <td>Ciudad</td><td><input type="text" name="city"
id="city" required></td>
    </tr>
    <tr>
        <td>País</td><td><select name="country_id"
id="country_id">
            <?php
                foreach ($países as $cod_pais=>$pais){
                    echo "<option value='$cod_pais'$>$pais</option>\r";
                }?>
            </select></td>
    </tr>
    <tr>
        <td><input type="submit" value="enviar"></td><td><input
type="reset" value="borrar formulario"></td>
    </tr>
</table>

</form>

<!--
    INSERT INTO `city` (`city_id`, `city`, `country_id`,
`last_update`) VALUES (NULL, '', '', current_timestamp())
-->
<?php
if (isset($_REQUEST['city_id']) && isset($_REQUEST['city'])) {
    $city_id=$_REQUEST['city_id'];
    $city=$_REQUEST['city'];
    $country_id=$_REQUEST['country_id'];
    try{
        $res=insertarCiudad($city_id,$city,$country_id);

        echo ($res==true?"Ciudad insertada correctamente":"Ha ocurrido un
error al insertar la ciudad");
        if (isset($errores) && !empty($errores)){
            foreach ($errores as $e){
                echo "<div class='error'$>$e</div>";
            }
        }
    }catch(PDOException $e){
        echo "Ha ocurrido un error al insertar la tabla en la base de
datos" . $e->getMessage() . " " . $e->getCode() . "<br>";
    }
}
?>

```

### Descripción de la página de alta.

La página alta cuenta con un formulario HTML y un script php embebido en la página.

El formulario tiene un *action* hacia la misma página.

El formulario tiene un campo oculto *opc* con el valor alta, para que la navegación se mantenga en esta página.

Para cargar los valores de país, se ha utilizado una función de php que devuelve todos los países disponibles en un array, con código de país y nombre de país.

Cuando el usuario pulsa en el botón *submit*, la acción es dirigida a la misma página, pero el código php mira si están definidos y con valor los campos del formulario *city\_id* , y *city*.

Una vez que estamos seguros de que se ha pulsado el botón enviar se llama a la función de php insertar ciudad, pasándole todos los parámetros del formulario.

La función *insertarCiudad* llama a su vez a la función *existeCiudad*, para no insertar valores duplicados.

Tras la inserción de la ciudad en la tabla se muestra un mensaje en caso de que la inserción haya sido exitosa, y un mensaje de error en caso contrario.

Las operaciones DML están dentro de un try catch para el control de errores.

## Listado.php

```
<div class="formulario_búsqueda">
  <form action="<?=$_SERVER['PHP_SELF']?>" method="post">
    <input type="hidden" name="opc" value="listar">
    <table>
      <tr>
        <td>Nombre</td>
        <td><input type="text" name="buscanombre"></td>
      </tr>
      <tr>
        <td>País</td>
        <td><input type="text" name="buscapais"></td>
      </tr>
      <tr>
        <td><input type="submit" value="buscar"></td>
        <td><input type="reset" value="borrar"
formulario"></td>
      </tr>
    </table>
  </div>
<?php
if (isset($_REQUEST['opc']) && $_REQUEST['opc']=="listar"){
  $city=$_REQUEST['buscanombre'];
  $pais=$_REQUEST['buscapais'];
  include_once("conexiones.php");
  $stm=buscaCity($city,$pais);
?>

<div class="listado">

  <table class="tabla">
    <tr>
      <th>código ciudad</th>
      <th> Ciudad</th>
```

```

        <th> País</th>

    </tr><?php
while($resultado=$stm->fetch(PDO::FETCH_ASSOC)) {
    ?>
    <tr>
        <?php
        foreach ($resultado as $k=>$v) {
            ?>

            <?php
            echo "<td>$v</td>\r";
            ?>

            <?php
            }
            ?>
            <td><a
href="<?=$_SERVER['PHP_SELF']?>?opc=modificar&id=<?=$resultado['city_id']?>"></a><a href="#"
onClick="if(confirm('¿Confirma que desea eliminar
la ciudad de
<?=$resultado['city']?>')) {window.location='eliminar.php?id=<?=$resultado['city_id']?>'}"
; &nbsp;</a></td>
        </tr>
        <?php
    }
}
    ?>
</table>
</div>

```

## Descripción de la página listado

La página listado.php cuenta con un formulario de búsqueda y una tabla con los resultados.

El formulario de búsqueda tiene 2 campos : nombre y país, así como un campo oculto “opc” para la navegación.

Una vez que el usuario ha introducido uno o dos valores en el formulario , se llama a la función buscaCity pasándole por parámetro los dos valores.

La función buscaCity utilizando PDO prepare y PDO bindParams compone una consulta SQL.

Admite valores aproximados pues utiliza LIKE en vez de el operador = , por lo que devuelve un listado con todos los registros que cumplen con los parámetros pasados.



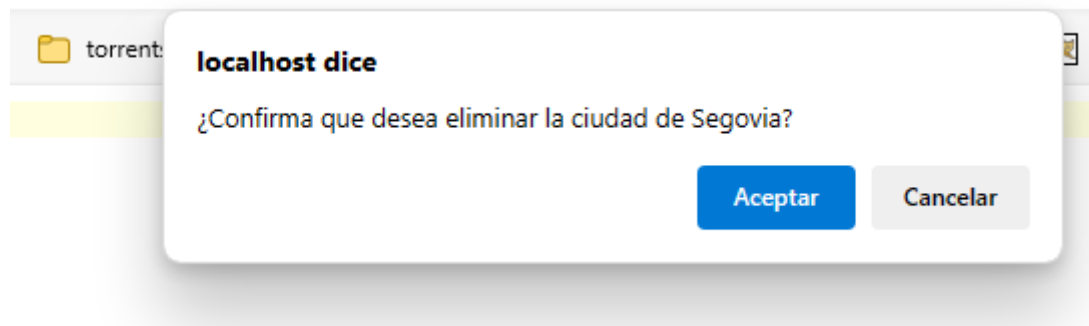
En caso de error será capturado mediante try / catch .

El listado de la página se compone de los campos de la tabla, y al final del listado muestra dos iconos con la función de modificar y eliminar registro.

código ciudad	Ciudad	País		
1	A Coruña (La Coruña)	Spain		
7	Aden	Spain		
146	Donostia-San Sebastián	Spain		
181	Gijón	Spain		
974	Madrid	Spain		
388	Ourense (Orense)	Spain		
459	Santiago de Compostela	Spain		
666	Segovia	Spain		

El icono de modificar redirige la acción a la página Modificar.php.

Si el usuario pulsa en el icono de eliminar, primero muestra un mensaje de advertencia



Este mensaje es mostrado mediante la función de javascript "confirm", y en caso de que el usuario elija "aceptar" será redirigido a la página de eliminar.

## Página Modificar.php

```
<?php
include_once ("conexiones.php");
conecta();
$países=cargaPaíses();
if (isset($_REQUEST) && isset($_REQUEST['id'])) {
    $id=$_REQUEST['id'];
    try{
        $registro=dameCiudad($id);
        $city_id=$registro['city_id'];
```

```

        $city=$registro['city'];
        $country_id=$registro['country_id'];
    }catch (PDOException $e){
        foreach ($errores as $e){
            echo "<div class='error'>$e</div>";
        }
        die ("Ha ocurrido un error al obtener la ciudad" . $e-
>getMessage() . " " . $e->getCode());
    }

}
?>
<form action="" method="post">
    <input type="hidden" value="opc" value="modificar">
    <table>
        <tr>
            <td>Código ciudad</td>
            <td><input type="number" name="city_id" id="city_id"
maxlength="3" readonly value='<?=$city_id?>'></td>
        </tr>
        <tr>
            <td>Ciudad</td>
            <td><input type="text" name="city" id="city" required
value='<?=$city?>'></td>
        </tr>
        <tr>
            <td>País</td>
            <td><select name="country_id" id="country_id">
                <?php
                    foreach ($paises as $cod_pais=>$pais){
                        echo "<option value='$cod_pais' " .
($cod_pais==$country_id?" selected ':'") . ">$pais</option>\r";
                    }?>
            </select></td>
        </tr>
        <tr>
            <td><input type="submit" value="enviar"></td>
            <td><input type="reset" value="borrar formulario"></td>
        </tr>
    </table>

</form>

<!--
    INSERT INTO `city` (`city_id`, `city`, `country_id`,
`last_update`) VALUES (NULL, '', '', current_timestamp())
-->
<?php
if (isset($_REQUEST['city_id']) && isset($_REQUEST['city'])) {
    $city_id=$_REQUEST['city_id'];
    $city=$_REQUEST['city'];
    $country_id=$_REQUEST['country_id'];
    try{
        $res=modificarCiudad($city_id,$city,$country_id);

        echo ($res==true?"Ciudad modificada correctamente":"Ha ocurrido un
error al modificar la ciudad");
        if (isset($errores) && !empty($errores)){
            foreach ($errores as $e){
                echo "<div class='error'>$e</div>";
            }
        }
    }
}

```

```

    }

} catch (PDOException $e) {
    echo "Ha ocurrido un error al modificar la tabla en la base de
datos" . $e->getMessage() . " " . $e->getCode() . "<br>";
}

}
?>

```

## Descripción modificar.php

La página modificar.php es muy similar a la de alta.

Utiliza el mismo formulario (copiado y pegado) y precarga los valores a modificar.

La página modificar recibe por parámetro el ID de la ciudad a modificar en el parámetro ID.

Al inicio de la página realiza una búsqueda de esa ID, y si devuelve resultados será lo que muestre en el formulario.

El campo PK de la tabla no se puede modificar (city\_id), por lo que el formulario tiene ese campo con el atributo readonly.

Una vez el usuario ha realizado la modificación (sólo se permite modificar el nombre de la ciudad y el país), el código PHP llama a la función **modificarciudad**.

Esta función está dentro de un try catch , y en caso de error lo mostrará al usuario.

## Página eliminar.php

```

<?php
include_once("conexiones.php");
if (isset($_REQUEST) && isset($_REQUEST['id'])) {
    $id=$_REQUEST['id'];
    try{
        if (eliminaCiudad($id)){
            echo "Ciudad eliminada correctamente";
        }else{
            echo "No se ha podido eliminar la ciudad";
        }
    } catch (PDOException $e) {
        echo "Ha ocurrido un error al eliminar la ciudad de la base de
datos " . $e->getMessage() . " " . $e->getCode();
        foreach($errores as $e){
            echo "<div class='error'>$e</div>\r\n";
        }
    }
}

?>
<br/>
<hr>
<a href="<?=$_SERVER['HTTP_REFERER']?>"> Volver </a>

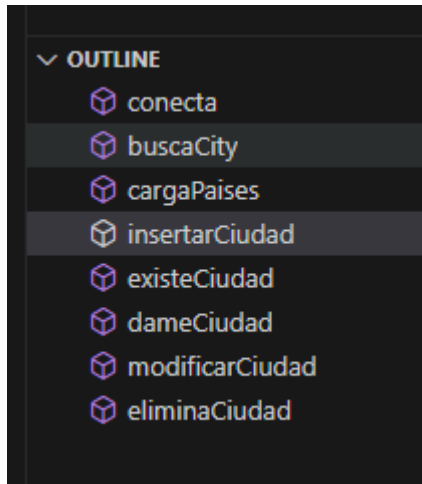
```

Esta página recibe un id de ciudad por parámetro y directamente trata de eliminar el registro llamando a la función **eliminaCiudad** (\$id).

Esta función elimina la ciudad y en caso de error lo controla mediante try catch.

La página muestra un enlace para volver a la página anterior.

## Página de conexiones.php



```
<?php
$host="192.168.1.134";
$username="david";
$password="MBcaYoZE1G[cRok6";
$dbname="sakila";

$conexion;
$errores=array();
function conecta() {
    global $host,$username,$password,$dbname;
    $dsn = "mysql:host=$host;dbname=$dbname;charset=utf8mb4";
    try {
        // Ejecutamos las variables y aplicamos UTF8
        $connect= new PDO ( $dsn, $username, $password );

        return $connect;
    } catch (PDOException $e) {
        exit ("Error: " . $e->getMessage ());
    }
}

function buscaCity($city,$pais)
{
    global $conexion;
    if ($conexion==null) {
        $conexion=conecta();
    }
    $sql ="select c.city_id , c.city , co.country from city c inner join
country co on c.country_id = co.country_id where 1";
    $a=false;
    if (!empty($city)) {
        $sql.=" AND c.city like CONCAT('%', :ciudad ,'%')";
    }
}
```

```

    }
    if (!empty($pais)) {
        $pais="%"$pais%";
        $sql.=" AND co.country like :pais";
    }
    $sql .= " order by c.city, c.last_update ";
    $stm = $conexion->prepare($sql);

    if (!empty($city)) {
        $stm->bindParam('ciudad', $city, PDO::PARAM_STR);
    }
    if (!empty($pais)) {
        $stm->bindParam('pais', $pais, PDO::PARAM_STR);
    }
    try {
        $stm->execute();
    }
    catch ( PDOException $Exception ) {
        // PHP Fatal Error. Second Argument Has To Be An Integer, But
        PDOException::getCode Returns A
        // String.
        throw new MyDatabaseException( $Exception->getMessage() ,
        $Exception->getCode() );
    }

    $rowCount = $stm->rowCount();
    return $stm;
}

function cargaPaises() {
    $devuelve=array();
    global $conexion;
    try{
        if ($conexion==null) {
            $conexion=conecta();
        }
        $stm= $conexion->prepare("SELECT country_id,country from
country");
        $stm->execute();
        while($resultado=$stm->fetch(PDO::FETCH_ASSOC)) {
            $devuelve[$resultado['country_id']]=$resultado['country'];
        }
    } catch(PDOException $e) {
        throw new MyDatabaseException($e->getMessage() , $e-
>getCode());
    }
    return $devuelve;
}

function insertarCiudad($city_id,$city,$country_id) {
global $conexion,$errores;
if ($conexion==null) {
    $conexion = conecta();
}

    try{
        if (existeCiudad($city,$country_id)) {
            $errores[]="La combinación de ciudad y país ya
existe";
            return false;
        }
    }

```

```

        $stm=$conexion->prepare ("INSERT INTO city
(city_id,city,country_id,last_update) values
(:city_id,:city,:country_id,current_timestamp())");
        $stm->bindParam('city',$city,PDO::PARAM_STR);
        $stm->bindParam('city_id',$city_id,PDO::PARAM_STR);
        $stm->bindParam('country_id',$country_id,PDO::PARAM_STR);
        $stm->execute();

```

```

        return $stm->rowCount()>0?true:false;

```

```

    } catch(PDOException $e) {
        throw $e;
    }
}
function existeCiudad($ciudad,$pais):bool{
    global $conexion;
    if ($conexion==null) {
        $conexion=conecta();
    }
    $stm=$conexion->prepare("SELECT * FROM city where city=:city
AND country_id=:pais");
    $stm->bindParam("city",$ciudad);
    $stm->bindParam("pais",$pais);
    try{
        $stm->execute();
    } catch(PDOException $e) {
        throw $e;
    }
    return ($stm->rowCount()>0?true:false);
}
function dameCiudad($id):array{
    global $conexion,$errores;
    if ($conexion==null) {
        $conexion=conecta();
    }
    $stm=$conexion->prepare("SELECT * FROM city where
city_id=:id");
    $stm->bindParam("id",$id);
    try{
        $stm->execute();
    } catch(PDOException $e) {
        throw $e;
    }
    if ($stm->rowCount()>0) {
        return $stm->fetch(PDO::FETCH_ASSOC);
    } else{
        $errores[]="No hay ninguna ciudad disponible";
    }
}

```

```

function modificarCiudad($city_id,$city,$country_id){
    global $conexion,$errores;
    if ($conexion==null) {
        $conexion = conecta();
    }
}

```

```

        try{
            $stm=$conexion->prepare ("Update city set
city=:city, country_id=:country_id, last_update=current_timestamp()
where city_id=:city_id");
            $stm->bindParam('city',$city,PDO::PARAM_STR);
            $stm->bindParam('city_id',$city_id,PDO::PARAM_STR);
            $stm->bindParam('country_id',$country_id,PDO::PARAM_STR);
            $stm->execute();

            return $stm->rowCount()>0?true:false;

        } catch(PDOException $e) {
            throw $e;
        }
    }
}

```

```

function eliminaCiudad($ciudad):bool{
    global $conexion;
    if ($conexion==null){
        $conexion=conecta();
    }
    $stm=$conexion->prepare("Delete from city where
city_id=:id");
    $stm->bindParam("id",$ciudad);
    try{
        $stm->execute();
    } catch(PDOException $e) {
        throw $e;
    }
    return ($stm->rowCount()>0?true:false);
}
?>

```

## Descripción conexiones.php

Esta página contiene las credenciales para la conexión a la base de datos así como todas las funciones usadas en el resto de páginas.

Las funciones que ejecuta operaciones contra la base de datos están dentro de try catch .

Las funciones contenidas en este fichero son :

```

Conecta()
buscaCity($city,$pais)
cargaPaíses()
insertarCiudad($city_id,$city,$country_id)
existeCiudad($ciudad,$pais)

```

```
dameCiudad($id)
modificarCiudad($city_id,$city,$country_id)
eliminaCiudad($ciudad)
```

### Descripción de conexiones.php

La función conecta se conecta a la base de datos y devuelve la variable \$conexion;

La función buscaCity recibe por parámetro \$city , \$pais y devuelve un array asociativo con el resultado

La función cargaPaises devuelve un array asociativo con código país país. Es usada para carga el contenido del select.

La función insertarCiudad recibe como parámetro todos los campos de la tabla e inserta en la base de datos tras comprobar que la combinación de ciudad / país no existe.

La función dameCiudad recibe un id y devuelve el registro correspondiente a esa PK.

La función modificarCiudad recibe como parámetro el id de la ciudad, el nombre y el código de país y realiza una modificación en la tabla.

Finalmente, la función eliminaCiudad recibe un \$id de la ciudad. Elimina la ciudad y devuelve el número de registros afectados.

## Ejemplo de funcionamiento

Realizar un programa PHP Crud Create, Read, Update, Select de la tabla Country de Sakila. Obligatorio: usar mas de un formulario NOTA: Realizarlo en tecnología PDO.


[Alta](#) [Listado](#)

Nombre

Pais



Si hacemos clic en Alta

Realizar un programa PHP Crud Create, Read, Update, Select de la tabla Country de Sakila. 

[Alta](#)   [Listado](#)

Código ciudad

Ciudad

País

Afghanistan

Algeria

American Samoa

Angola

Anguilla

Argentina

Armenia

Australia

Austria

Azerbaijan

Bahrain

Bangladesh

Belarus

Bolivia

Brazil

Brunei

Bulgaria

Cambodia

Cameroon

Canada

Mostramos el formulario de alta

Introducimos un valor duplicado. Nos devuelve un error

[Alta](#)   [Listado](#)

Código ciudad

Ciudad

País

Ha ocurrido un error al insertar la tabla en la base de datosSQLSTATE[23000]: Integrity constraint violation: 1062 Duplicate entry '1' for key 'PRIMARY' 23000

## En caso de introducir valores únicos

Realizar un programa PHP Crud Create, Read, Update

[Alta](#)   [Listado](#)

Código ciudad

Ciudad

País

Ciudad insertada correctamente

## En caso de introducir combinación de ciudad + país que ya existe.

[Alta](#)   [Listado](#)

Código ciudad

Ciudad

País

Ha ocurrido un error al insertar la ciudad  
La combinación de ciudad y país ya existe

Si hacemos clic en el enlace del listado

Si buscamos sin introducir ningún valor o buscamos por una cadena muy común, como "a"









































[Alta](#)[Listado](#)

Nombre

Pais

buscar

borrar formulario

código ciudad	Ciudad	Pais	
1	A Coruña (La Coruña)	Spain	 
2	Abha	Saudi Arabia	 
3	Abu Dhabi	United Arab Emirates	 
4	Acuña	Mexico	 
5	Adana	Turkey	 
6	Addis Abeba	Ethiopia	 
7	Aden	Spain	 
8	Adoni	India	 
190	Águas Lindas de Goiás	Brazil	 
9	Ahmadnagar	India	 
10	Akishima	Japan	 
11	Akron	United States	 
12	al-Ayn	United Arab Emirates	 
13	al-Hawiya	Saudi Arabia	 
14	al-Manama	Bahrain	 
15	al-Qadarif	Sudan	 
16	al-Qatif	Saudi Arabia	 
17	Alessandria	Italy	 
18	Allappuzha (Alleppey)	India	 
19	Allende	Mexico	 

Si buscamos todas las ciudades de España (Spain)



















[Alta](#)[Listado](#)

Nombre

Pais

buscar

borrar formulario

código ciudad	Ciudad	Pais	
1	A Coruña (La Coruña)	Spain	 
7	Aden	Spain	 
7654	Ciudad Rodrigo	Spain	 
146	Donostia-San Sebastián	Spain	 
181	Gijón	Spain	 
974	Madrid	Spain	 
388	Ourense (Orense)	Spain	 
459	Santiago de Compostela	Spain	 
666	Segovia	Spain	 

Si hacemos clic en el icono de Modificar ciudad

[Alta](#)[Listado](#)

Código ciudad

666

Ciudad

Segovia

Pais

Spain

enviar

borrar formulario

Se muestra el formulario de modificar con los valores precargados , siendo código ciudad de sólo lectura.

Si cambiamos el país y damos a submit

enviar

borrar formulario

Ciudad modificada correctamente

Consultamos de nuevo la ciudad modificada anteriormente

Nombre

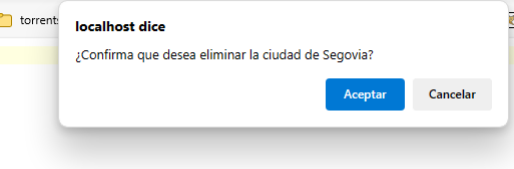
Pais

buscar

borrar formulario

código	ciudad	Ciudad	Pais
666	Segovia	Nepal	<div><div></div><div></div></div>

Si hacemos clic en el botón de eliminar



Se muestra un mensaje de advertencia

Ciudad eliminada correctamente

[Volver](#)

Y si pulsamos en aceptar, eliminamos la ciudad.