

## Table of contents

Table of contents .....	1
About Pure PHP Upload 3 .....	2
Features in Detail.....	3
The Basics: Uploading Files with Pure PHP Upload 3 .....	12
Advanced: Using Pure PHP Upload 3 with Insert Record .....	23
Advanced: Using Pure PHP Upload 3 with Update Record .....	39
Advanced: Creating an Ajax Upload Page .....	48
Advanced: Uploading Files to a Dynamic Folder.....	54
Advanced: Uploading Large Files.....	58
Reference: Customizable Progress Bar .....	60
Video: Uploading Files with Pure PHP Upload 3.....	63
Video: Using Pure PHP Upload 3 with Insert Record .....	63
Video: Using Pure PHP Upload 3 with Update Record .....	63
Video: Uploading Files to a Dynamic Folder .....	63
Video: Creating an Ajax Upload Page.....	63

## About Pure PHP Upload 3



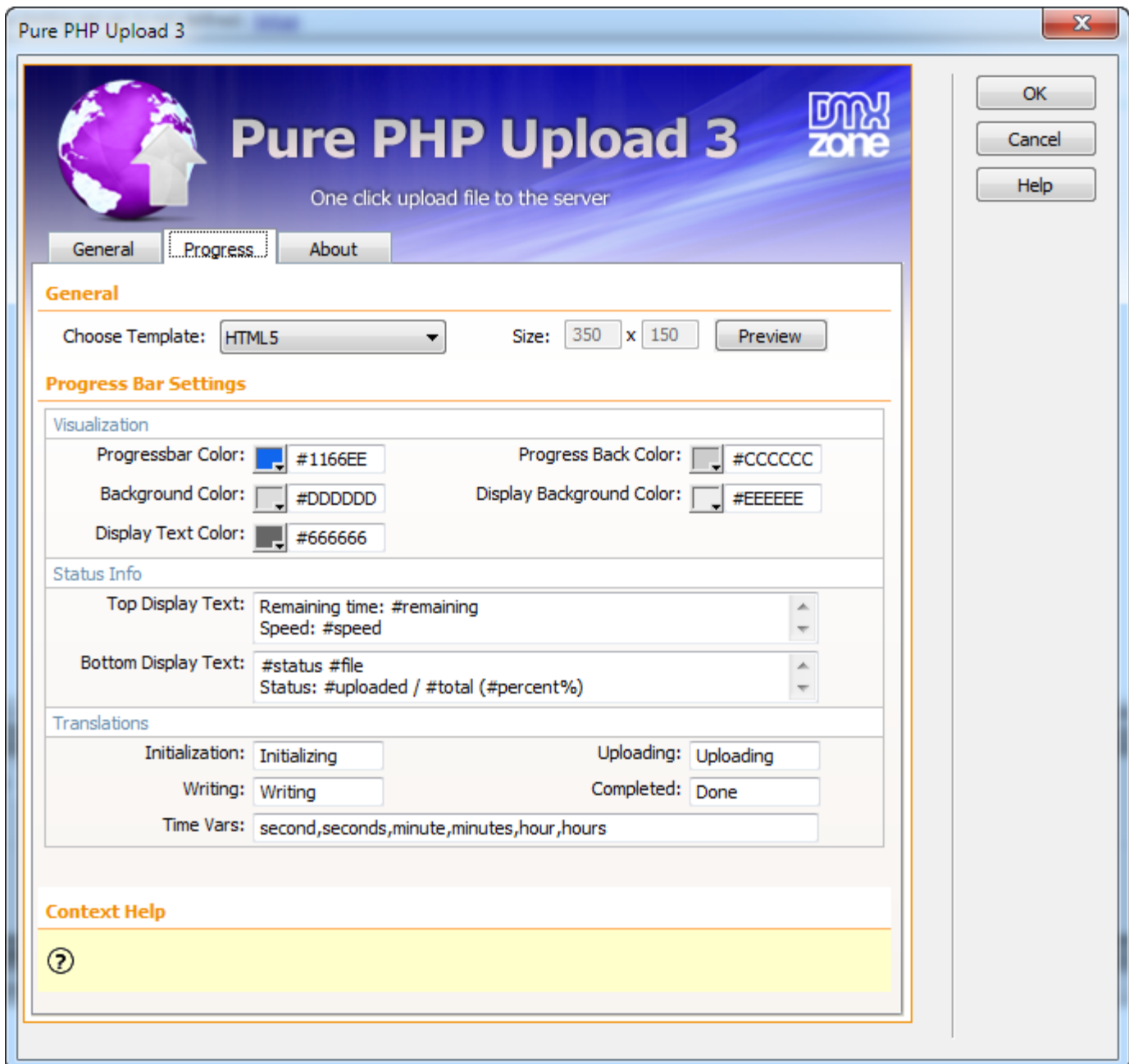
Pure PHP Upload 3 is the most **advanced upload solution** ever! It combines all the latest technologies in the web development world like **AJAX**, **HTML5** and **CSS3**. All available at your fingertips and in your trusted Dreamweaver environment. With just a few clicks, you become an expert.

Pure PHP Upload 3 empowers strong validated client and server side restricted uploads, making them more secure than ever. Full support for mobile file upload - now with IOS 6 and Android 4 you can upload any image from your mobile phone directly with a great looking HTML5 progress bar.

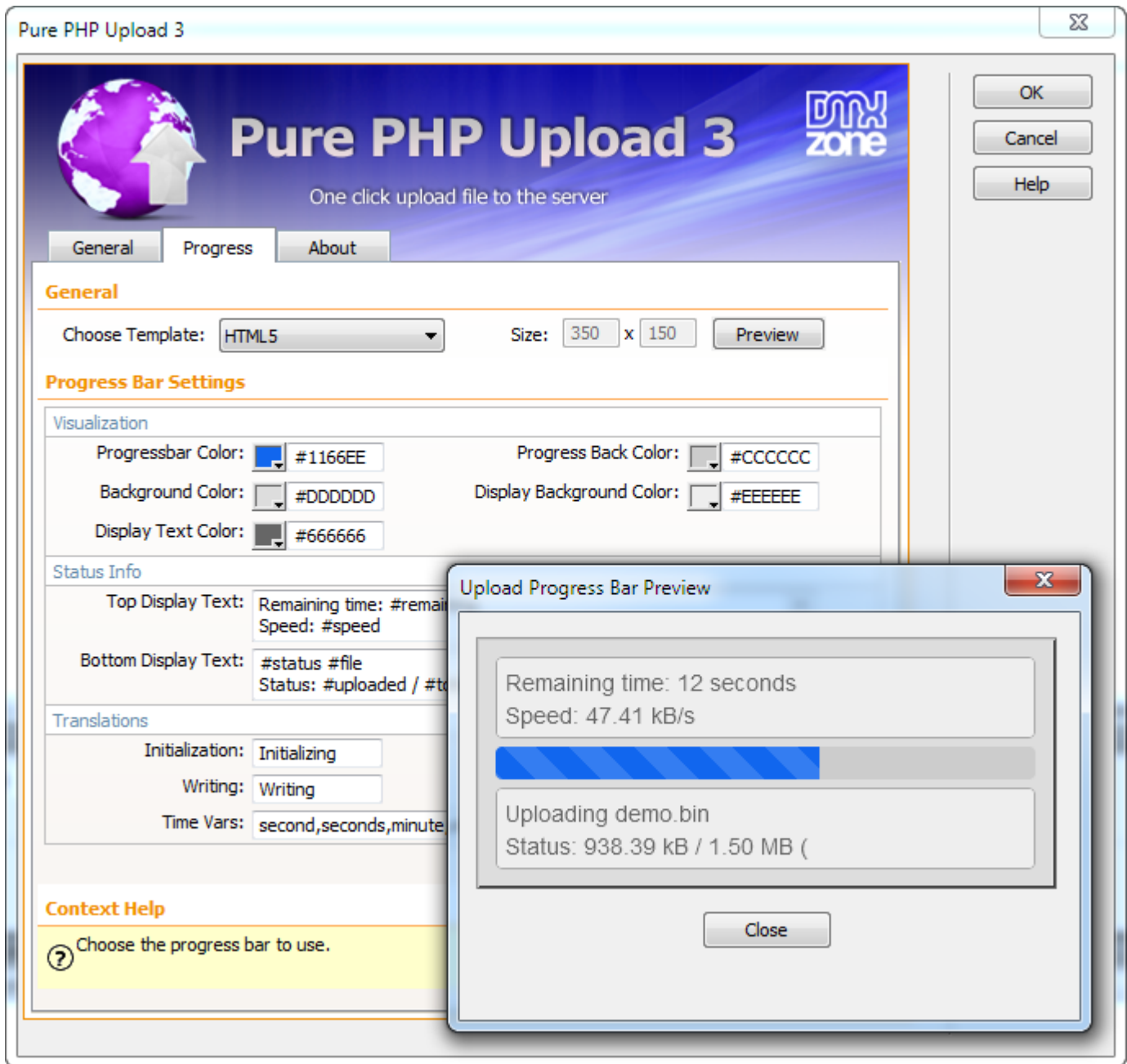
It also supports the [DMXzone Ajax Form](#) - turn your form to a dynamic Ajax form so that it is submitted and processed directly without any page refreshes and show the result directly inline! Combine [Pure PHP Upload 3](#) with [DMXzone Ajax Form](#) and you will get full Ajax File Upload! With real time accurate progress tracking in all the latest browsers!

## Features in Detail

- **Pure Ajax File Upload** - With [DMXzone Ajax Form](#) and **Pure PHP Upload 3** you can easily upload a file without the need to refresh the page in order to preview the result. Also with Ajax Upload the progress bar will run much smoother on the latest browsers as all progress is tracked by the client side browser and there is no need to trouble your server for it. You will get progress information even on PHP servers that do not support progress information or don't have the needed components installed.
- **Customizable progress bars** - The build-in Dreamweaver menu enables you to select multiple predefined Progress bars and customize them.



- **Progress bar preview** - You can preview your progress bar design directly in Dreamweaver by clicking the "preview" button.



- **Different options for each file upload field** – You can specify different options for each file upload field. This way you can have each upload field stored in a different location and specify which file types are allowed per field.

The screenshot shows the 'Usage' configuration section. At the top, there is a dropdown menu for 'Options for:' set to '\*\*\* All Files'. Below this is a section titled 'Upload location and type' containing: 'Upload Folder:' with a text input and a 'Browse...' button; 'Max File Size:' with a text input followed by 'KB' and a 'Require Upload' checkbox; 'Allowed Files:' with a dropdown menu set to 'All' and 'Extensions:' with a text input. The next section is 'Uploaded Images' with 'Min Width:', 'Max Width:', 'Min Height:', and 'Max Height:' each followed by a text input and 'px'. Below that is the 'After Upload' section with 'Go To:' and a 'Browse...' button. At the bottom is a 'Context Help' section with a question mark icon.

- **File conflict handling** – Determine how a file should be handled if its name conflicts with the name of an existing file in the directory.

The screenshot shows the 'General' configuration section of the Pure PHP Upload 3 interface. It features a header with a globe icon, the title 'Pure PHP Upload 3', and the DMXzone logo. Below the header are tabs for 'General', 'Progress', and 'About'. The 'General' tab is active and contains: 'Use Form:' with a dropdown menu set to 'dmxUniform1'; 'Max Form Size:' with a text input followed by 'KB'; 'Conflict Handling:' with a dropdown menu showing 'Overwrite', 'Skip', 'Error', and 'Make Unique'; 'Timeout:' with a text input followed by 'Sec'; and 'Store:' with radio buttons for 'Filename' (selected) and 'Full Path'. A 'Usage' section is partially visible at the bottom with an 'Options for:' dropdown menu.

- **Specify the uploaded file types** – From the dropdown menu you can select the file types that can be uploaded or enter their extensions in a comma separated list. Leave empty for no file restrictions.

Upload location and type

Upload Folder:  Browse...

Max File Size:  KB  Require Upload

Allowed Files: **All** Extensions:

Uploaded Images

Min Width:  Max Width: **Custom...**

Min Height:  px Max Height:  px

- **Upload engine for Advanced HTML Editor 3** - You can use Pure PHP Upload 3 extension as an uploading solution in the **Advanced HTML Editor 3**.

**Image Upload & Resize**

Allow Upload:  Upload Folder:  New Folder Browse...

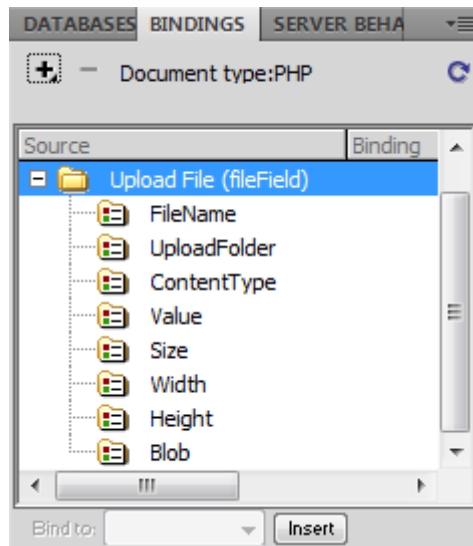
Allow Resize:  Max Width:  333 Max Height:  333

? Enable Image Resize after it is uploaded.

[? Online manual](#) OK Cancel

- **Advanced client and server side upload restrictions** - File size, image dimensions and type restrictions are checked client-side as well as server-side! This is the most secure way to allow uploads. Note that client side file size and image dimensions might not be allowed to be checked by all browsers due to security settings, but they will be checked server side anyway.
- **Multiple language support** - Pure PHP Upload 3 contains advanced browser detection that determines the language of your visitor and translates the messages of the extension into their language.

- **Extended Data bindings** - extended data bindings are available so that you can easily access the properties of the uploaded files and have total freedom to use them any way you intent to. Pure PHP Upload 3 allows you to set the properties of your form fields in the bindings menu.

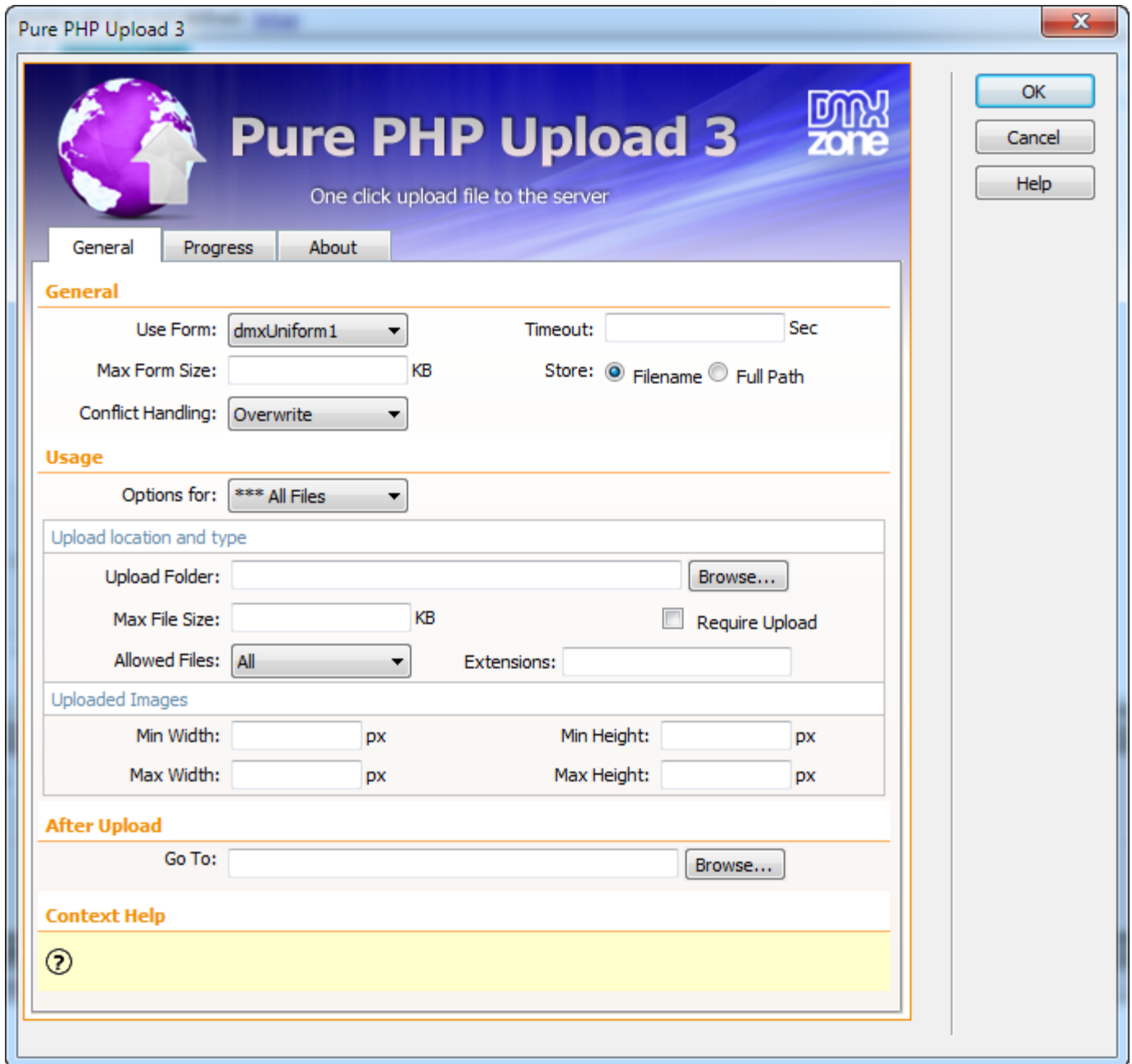


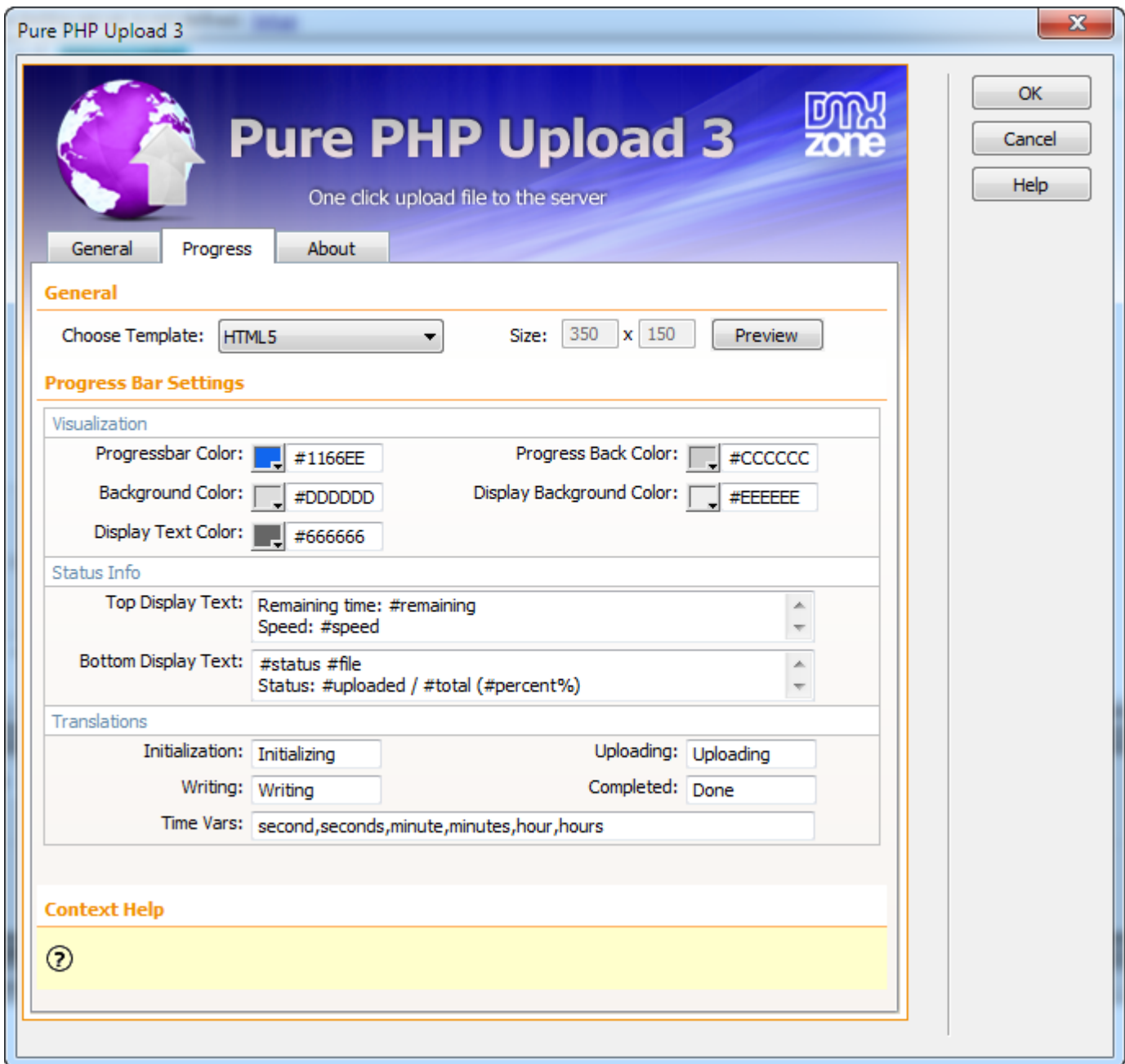
- **Fully object oriented** - Pure PHP Upload 3 is fully object oriented, making the extension more advanced than ever. It collects all the information of the files that you upload server-side. The extension collects data such as filenames, file sizes and the size of images from the upload class so you can store these values in your database.

```
1 <?php require_once('ScriptLibrary/incPureUpload.php'); ?>
2 <?php
3 /*** Pure PHP File Upload 3.0.0
4 // Process form form1
5 $ppu = new pureFileUpload();
6 $ppu->nameConflict = "over";
7 $ppu->storeType = "file";
8 $ppu->progressBar = "html5.htm";
9 $ppu->progressWidth = 350;
10 $ppu->progressHeight = 150;
11 $ppu->path = "New folder";
12 $ppu->redirectUrl = "";
13 $ppu->checkVersion("3.0.0");
14 $ppu->doUpload();
15 ?>
16 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
17 Transitional//EN"
18 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"
19 >
20 <html xmlns="http://www.w3.org/1999/xhtml">
21 <head>
22 <meta http-equiv="Content-Type" content="text/html;
23 charset=utf-8" />
24 <title>Untitled Document</title>
25 <script type="text/javascript"><?php echo $ppu->
26 generateScriptCode() ?></script>
27 <script src="ScriptLibrary/incPU3.js" type=
28 "text/javascript"></script>
29 </head>
30 <body>
31 <form action="<?php echo $DMX_uploadAction ?>" method=
32 "post" enctype="multipart/form-data" name="form1" id=
33 "form1" onsubmit="<?php echo $ppu->getSubmitCode() ?>
34 :return document.MM.returnValue">
```



- **Great Dreamweaver Integration** - Interactive dialog in Dreamweaver with all the options you'll need.





- **Fully cross-browser compatible** - Runs perfectly on any up-to-date desktop browser as well as mobile devices.



## **Before you begin**

Make sure to setup your site in Dreamweaver first before creating a page. If you don't know how to do that please [read this](#).

## **Installing the extension**

Read the tutorial at <http://www.dmxzone.com/go/4671/how-to-install-extensions-from-dmxzone/>

Login, download the product, and look up your serial at <http://www.dmxzone.com/user/products>

Get the latest extension manager from Adobe at [http://www.adobe.com/exchange/em\\_download/](http://www.adobe.com/exchange/em_download/)

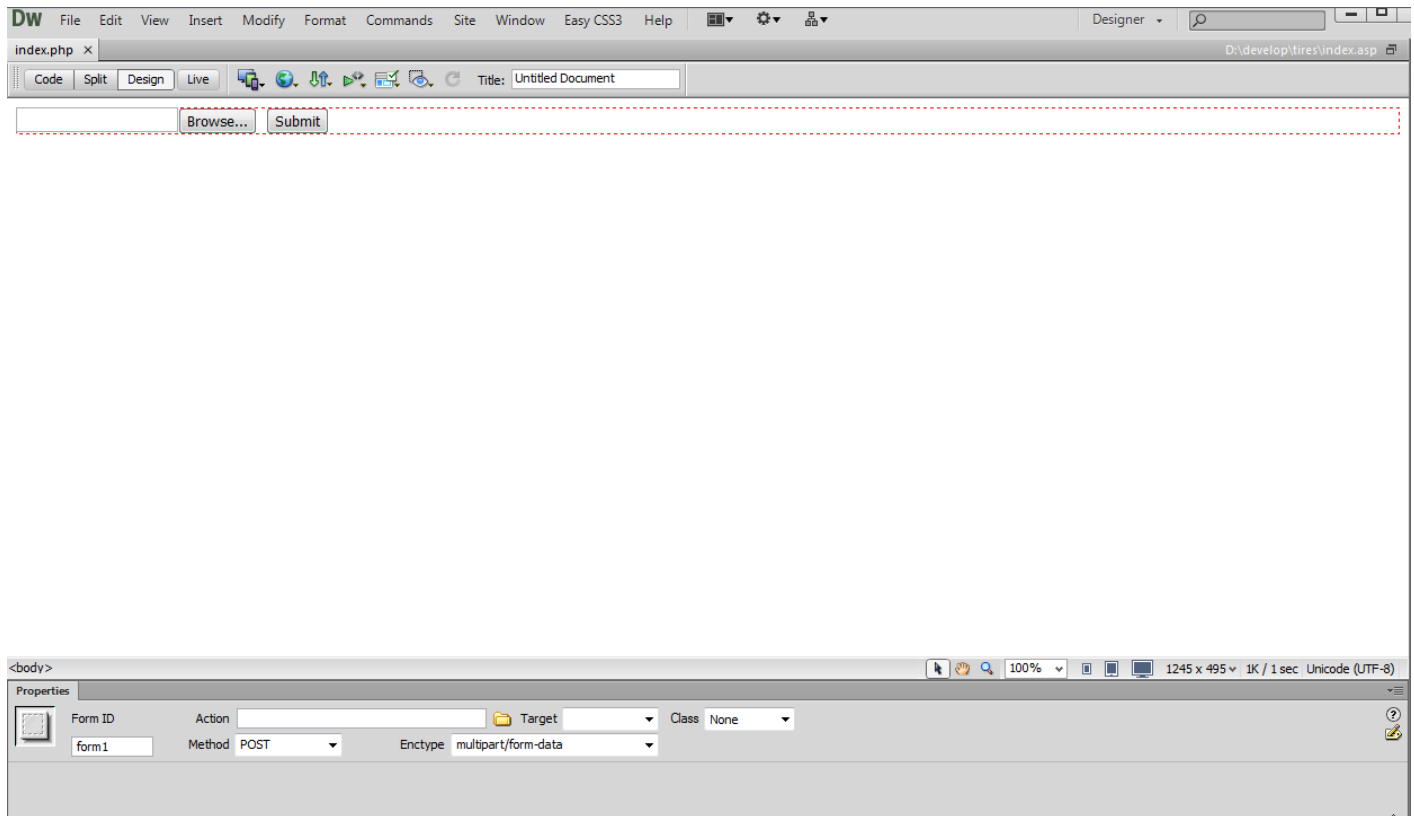
Open the .mxp file (the extension manager should start-up) and install the extension.

**If you experience any problems contact our Live Support!**

## The Basics: Uploading Files with Pure PHP Upload 3

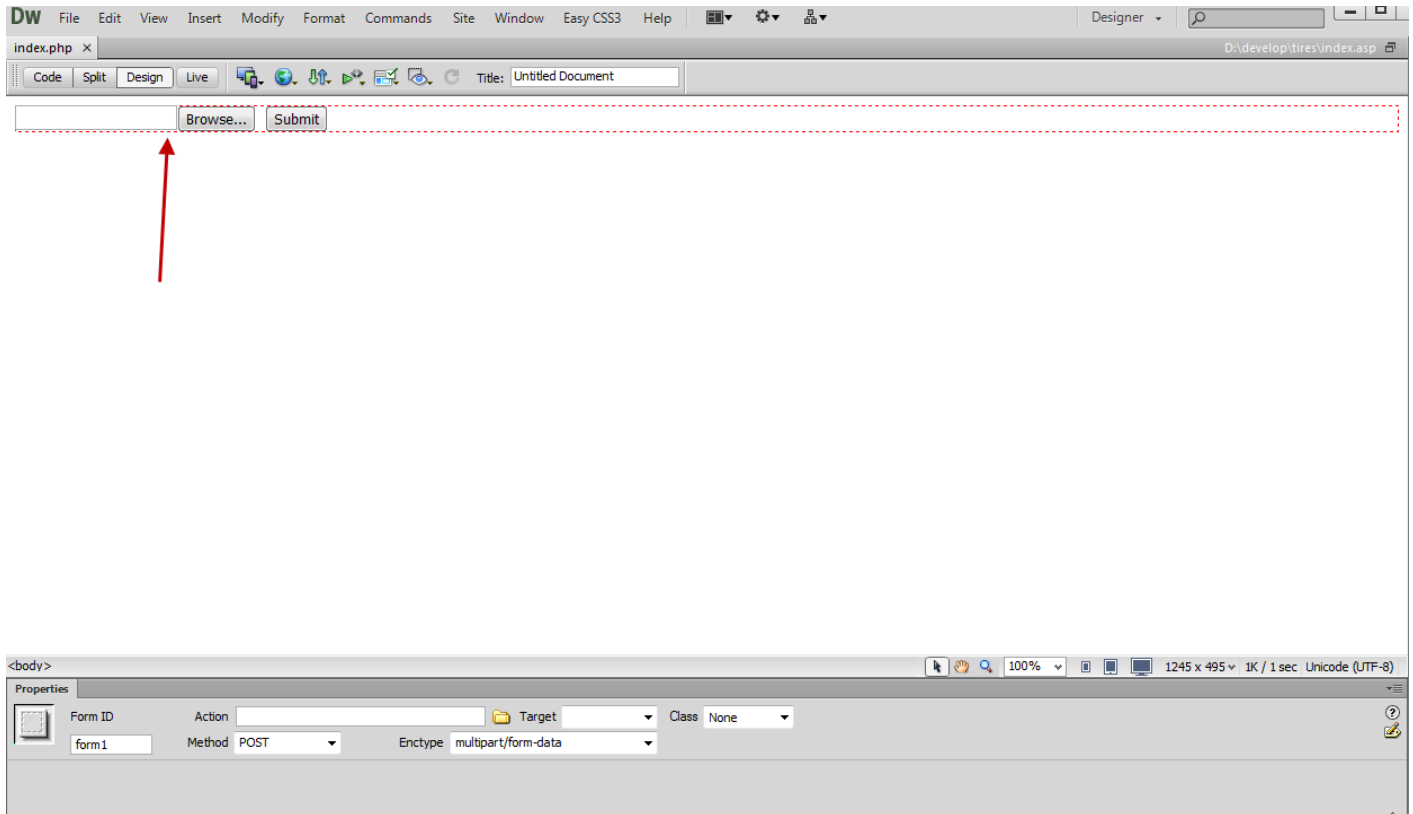
In this tutorial we will show you how to upload files using [Pure PHP Upload 3](#).

\*You can also check out the [video](#).

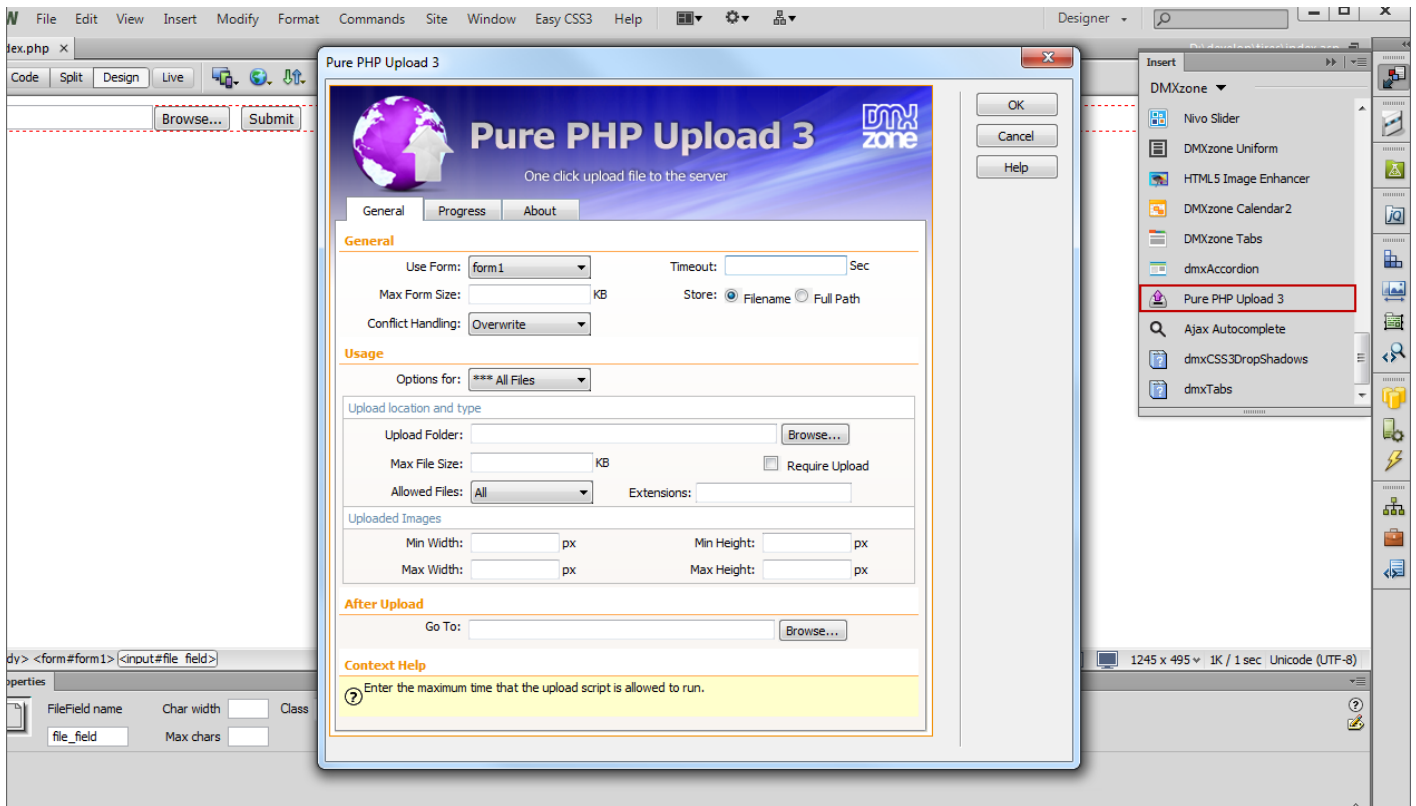


## How to do it

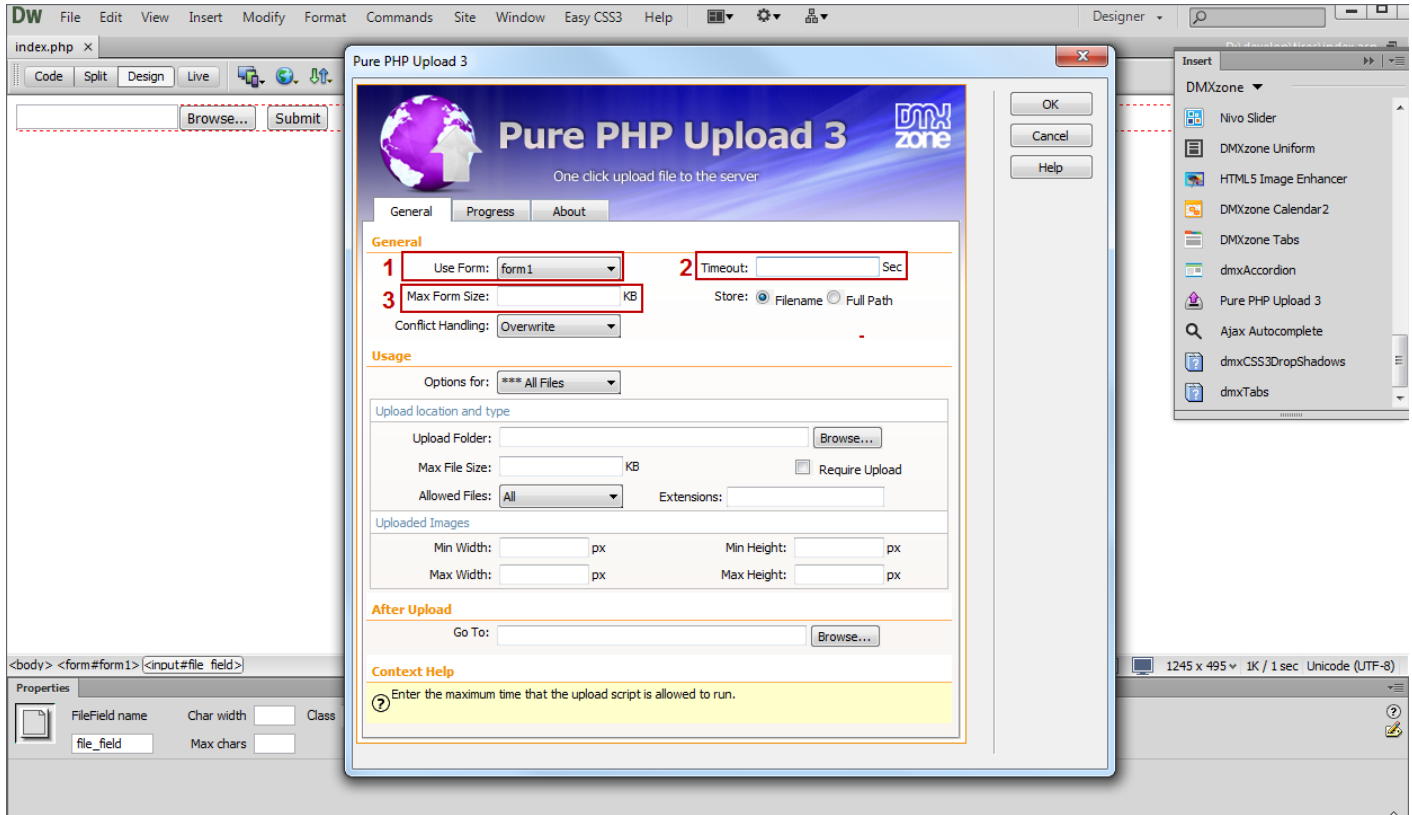
1. We've already created a simple form that contains a file upload field.



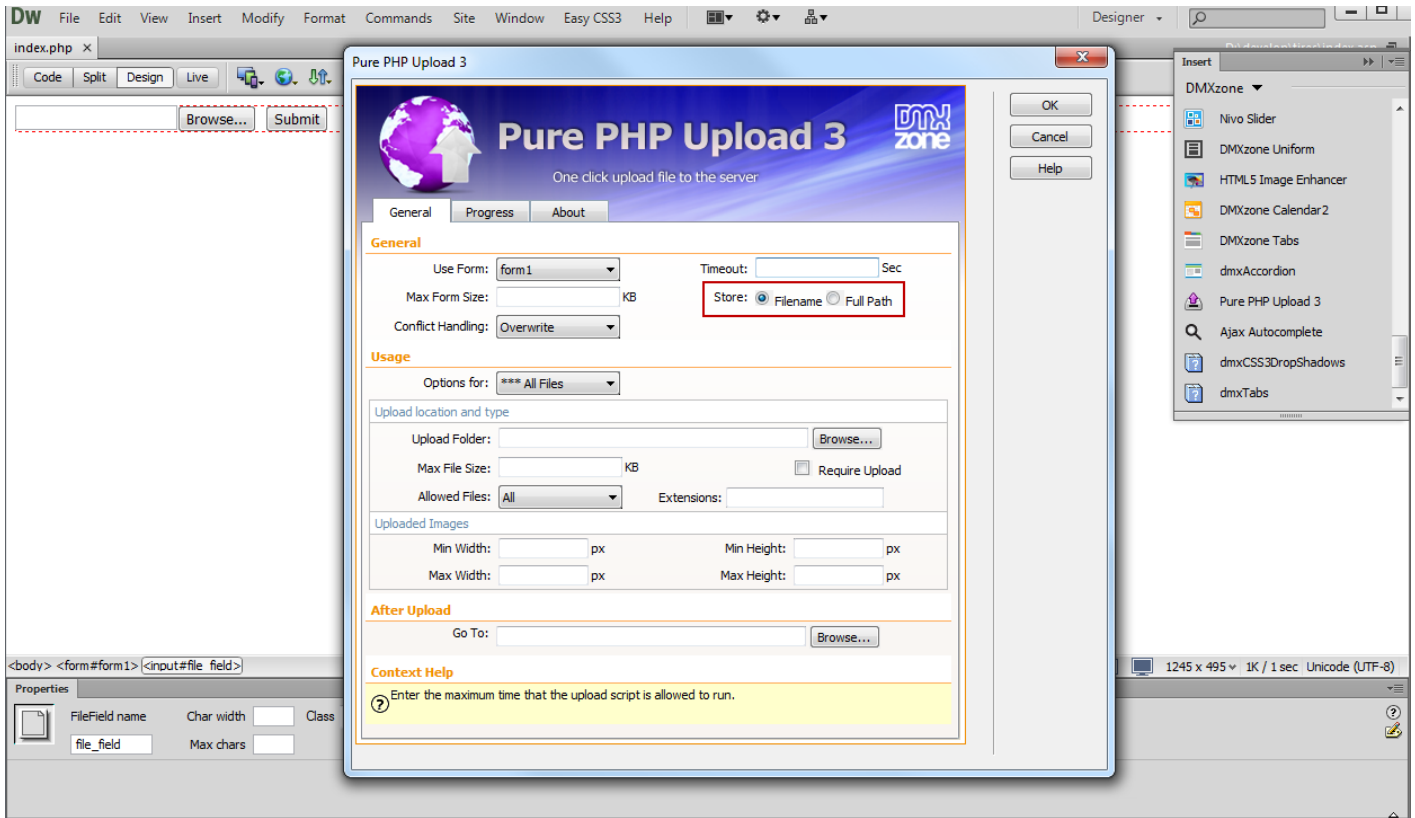
2. Click on the **Pure PHP Upload 3** icon and the window will appear.



- In the general options **select your form (1)**. We will leave the **timeout (2)** (maximum time that the upload script is allowed to run) empty for no time limitation and the **max form size (3)** (the max form size for all the form fields allowed to be uploaded) blank for the same reason.

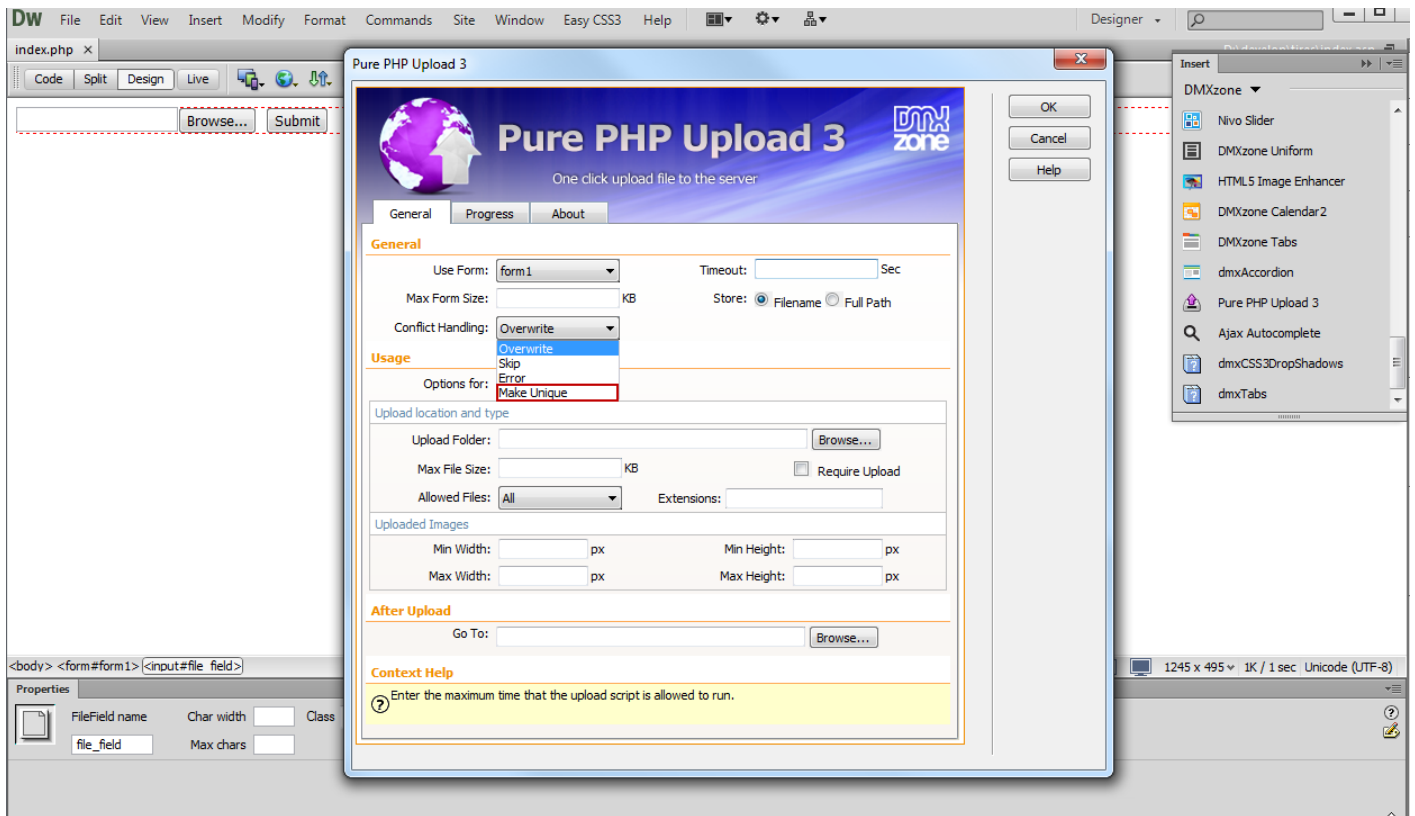


- Choose whether to store the **Filename** or the **Full path** if you are using a database to store the uploaded file names. In our case we choose to store the **Filenames**.

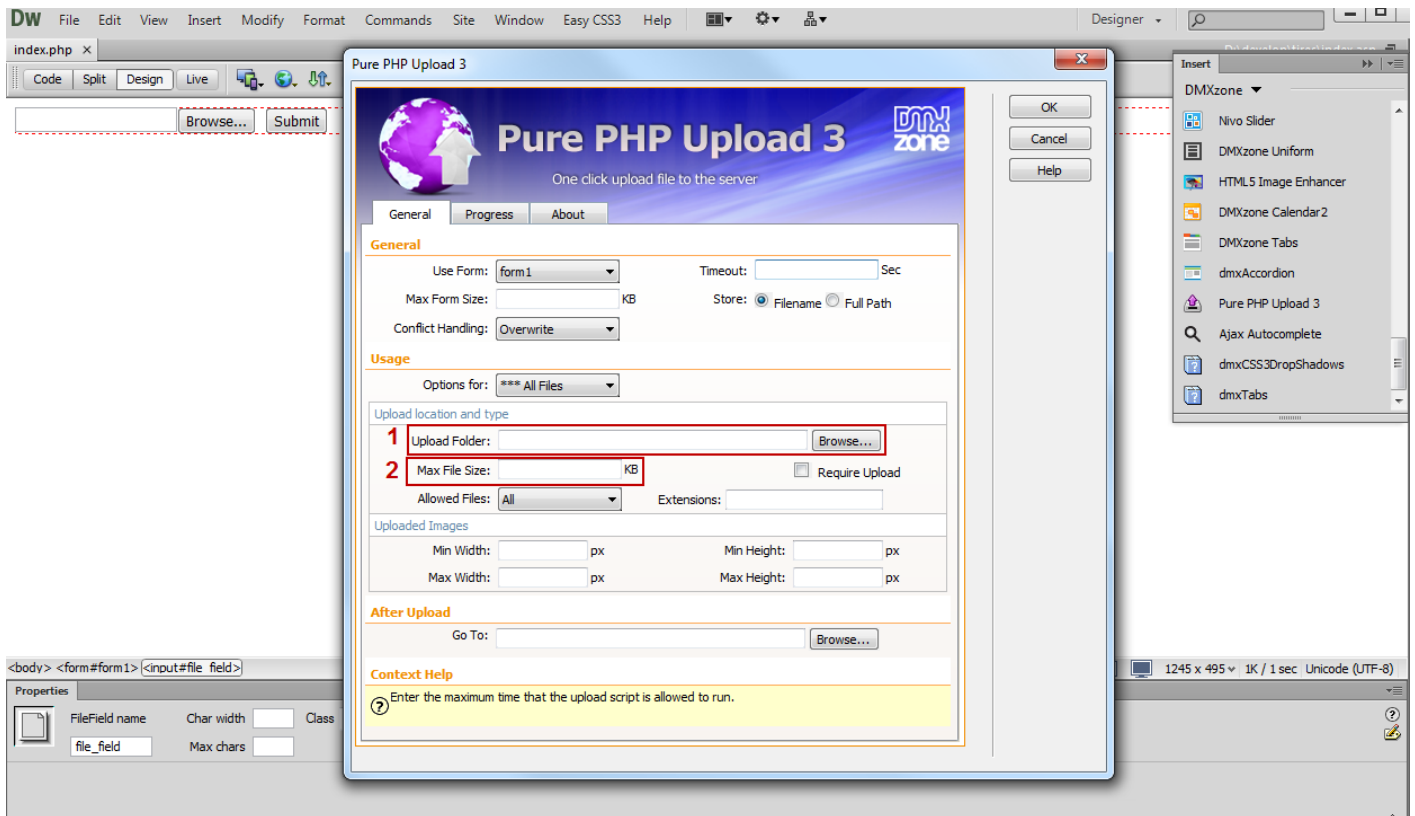




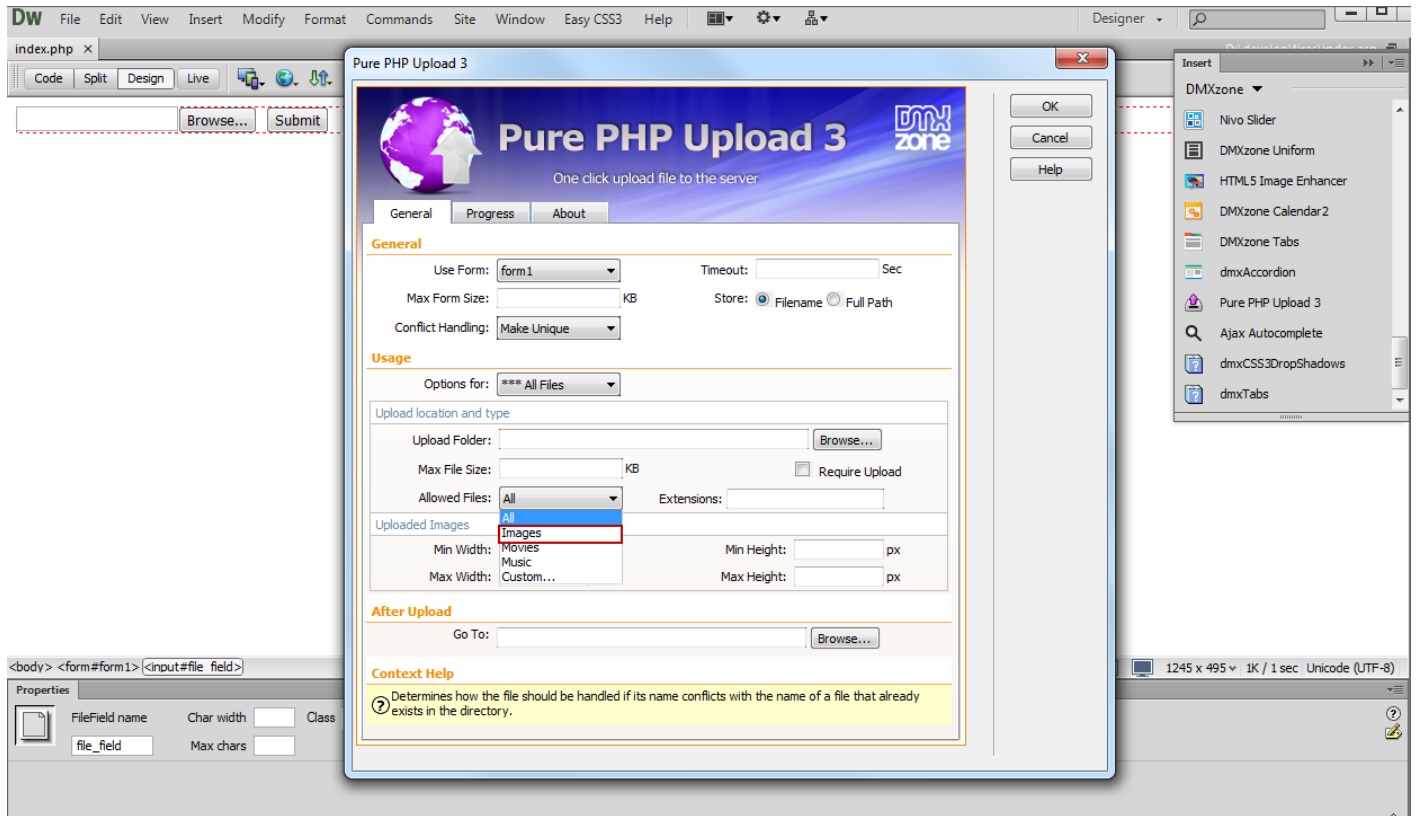
- Select how the file should be handled if its name conflicts with a file that already exists in the upload folder. We choose the **Make unique option**.



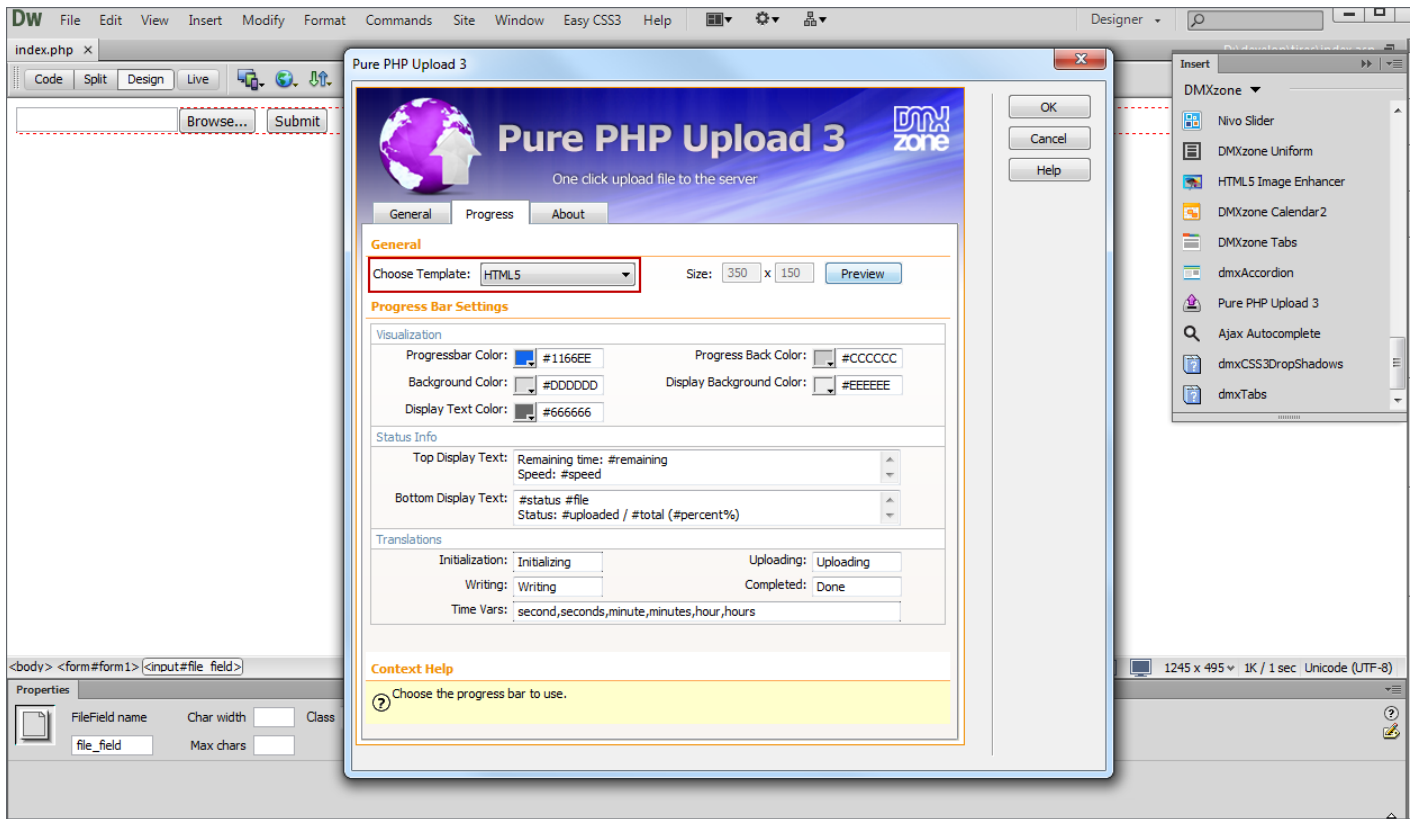
- We choose a different folder from the **Upload folder field (1)** as we want to store the files in a different one. Also, we leave empty the **Max file size (2)** as we don't want any size limitation.



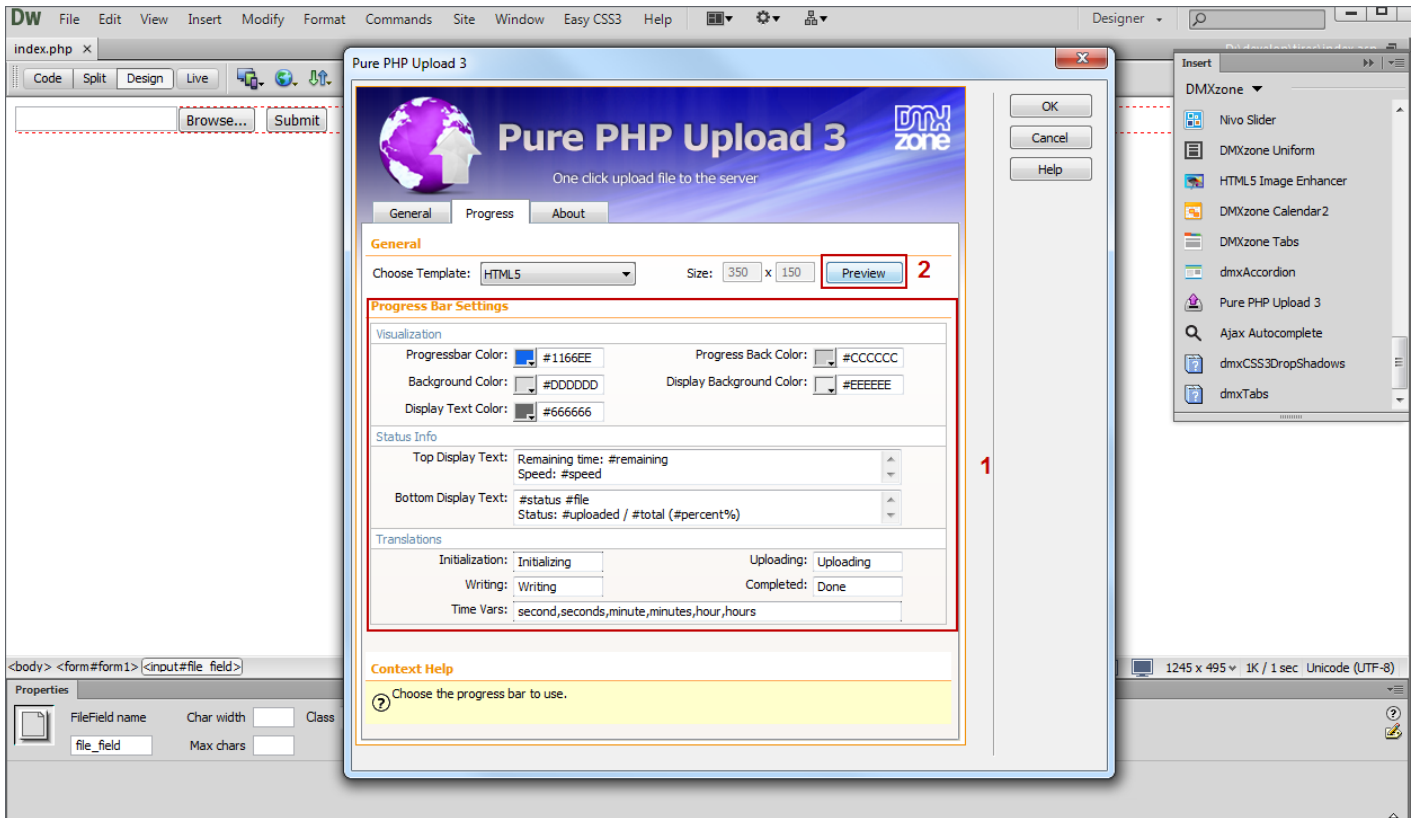
- For our tutorial we will limit the **uploaded file type** only to **images**, without any restrictions for the images size. The redirect URL is only needed if you are NOT using Insert/Update Record behavior.



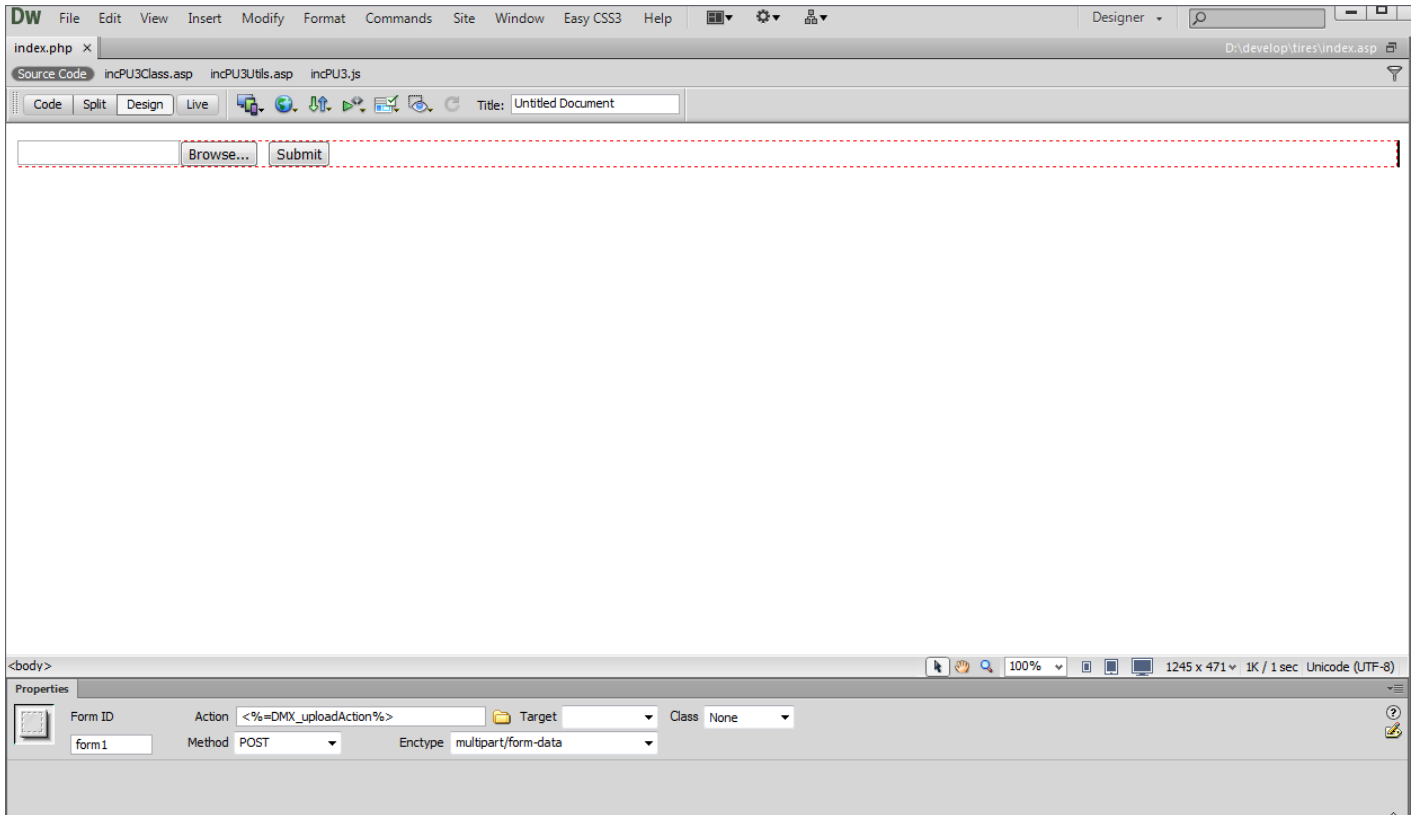
8. In the **Progress tab (1)** we will select one of the predefined progress bar designs, the **HTML5 one (2)**.



9. You can **customize the progress bar (1)** the way you want and **preview it directly in Dreamweaver (2)**.



10. Now you can save your work and preview the result in the browser.



## Advanced: Using Pure PHP Upload 3 with Insert Record

In this tutorial we will show you how to upload images and save their names, width and height into a database.

\*You can also check out the [video](#).

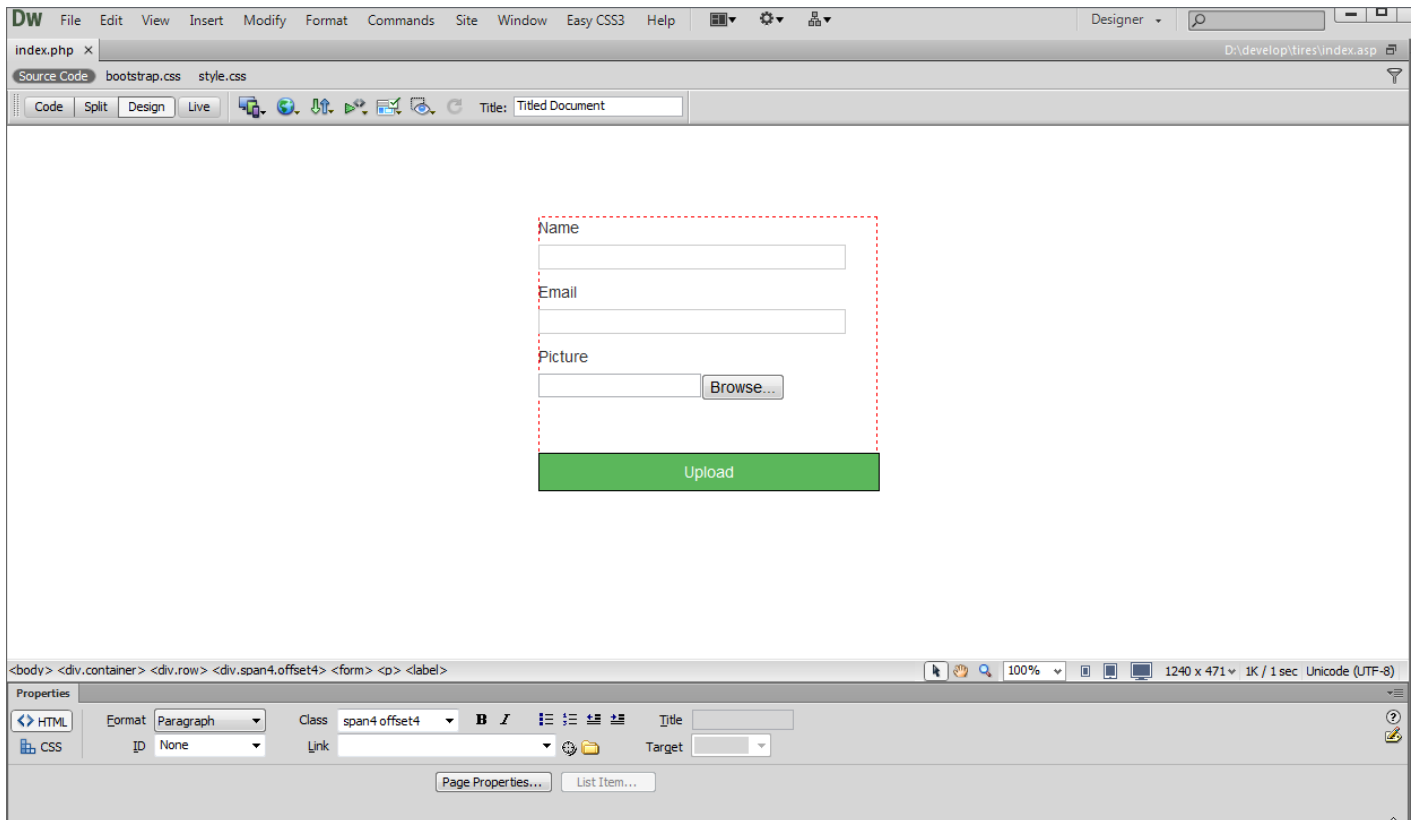
Name

Email

Picture

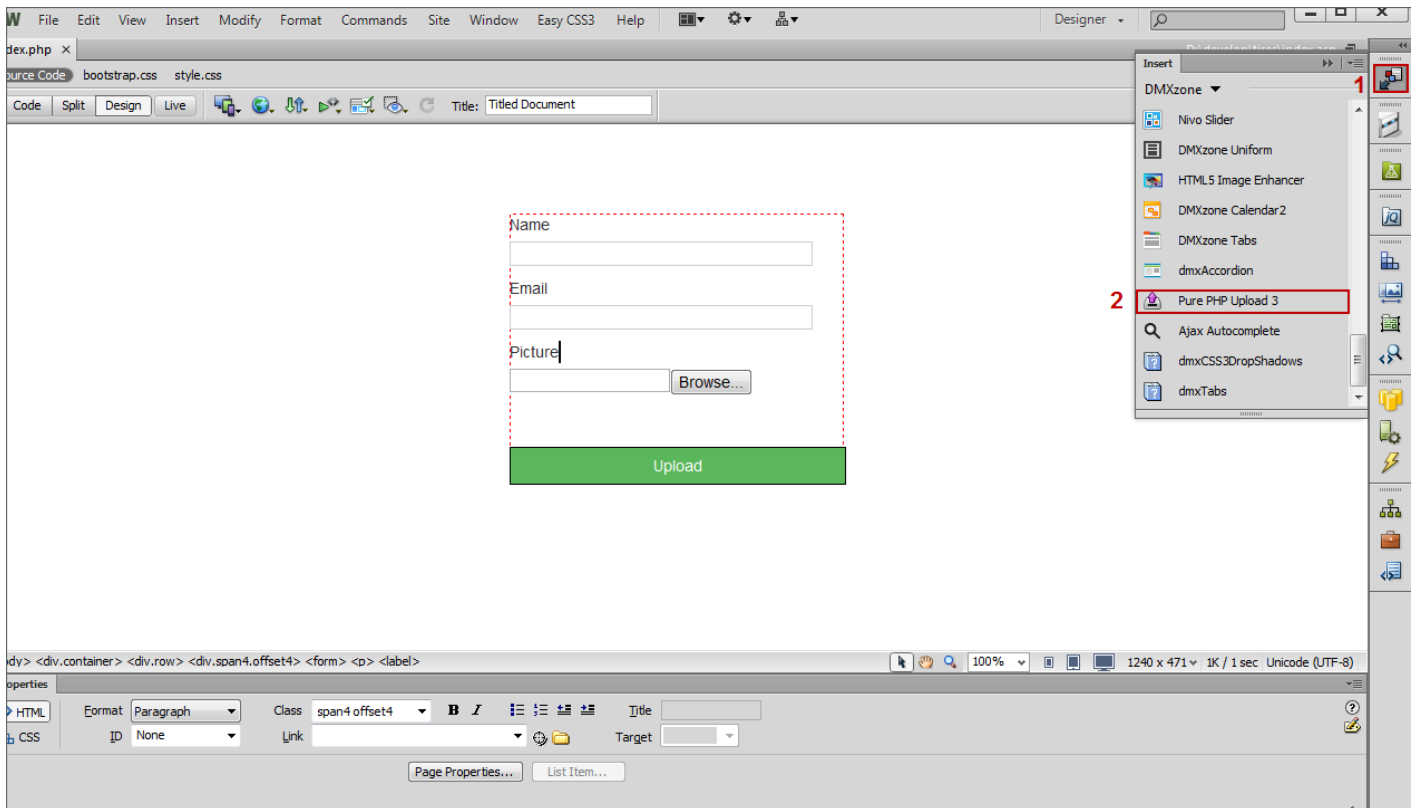
## How to do it

1. We've already created a simple form with three fields in it.

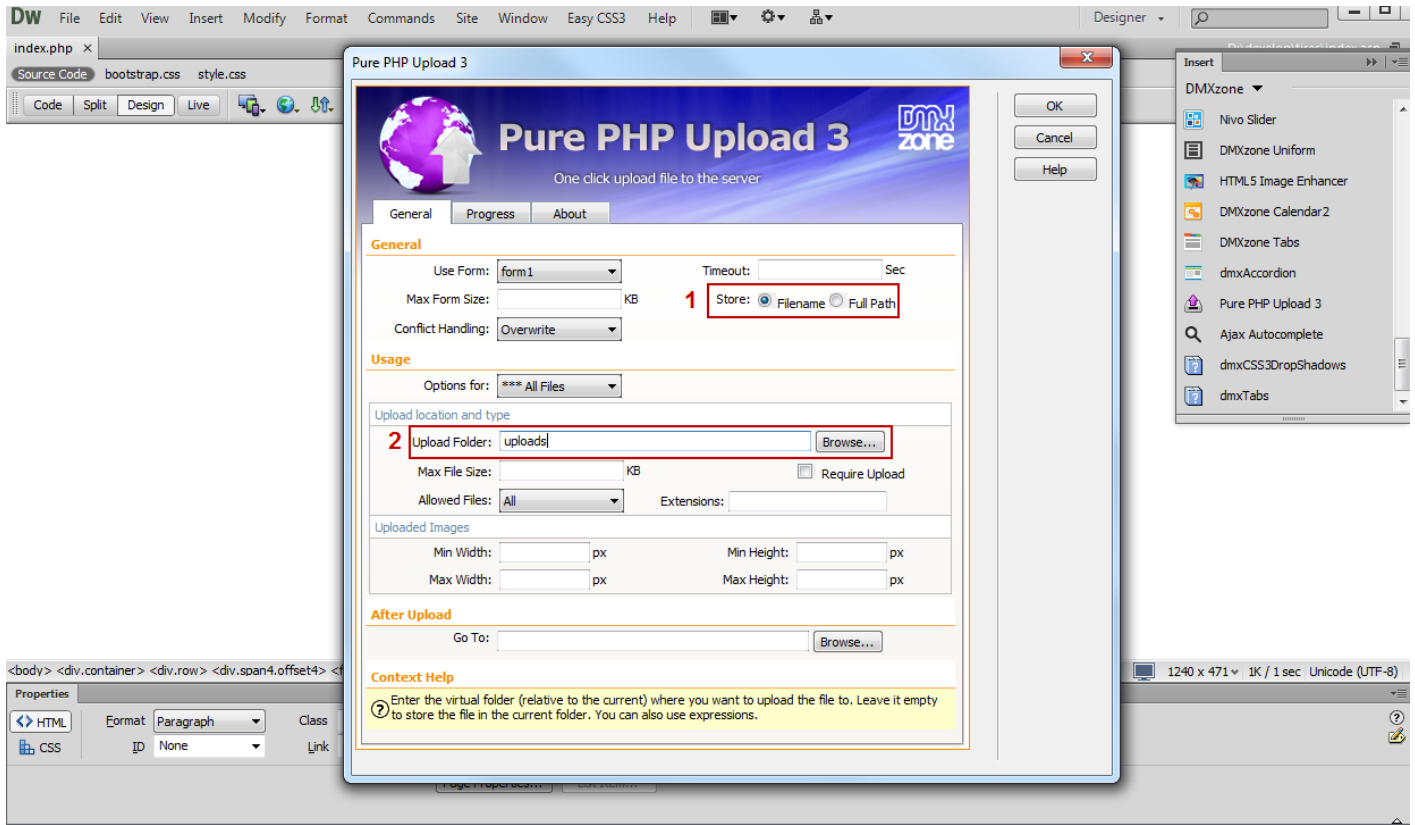




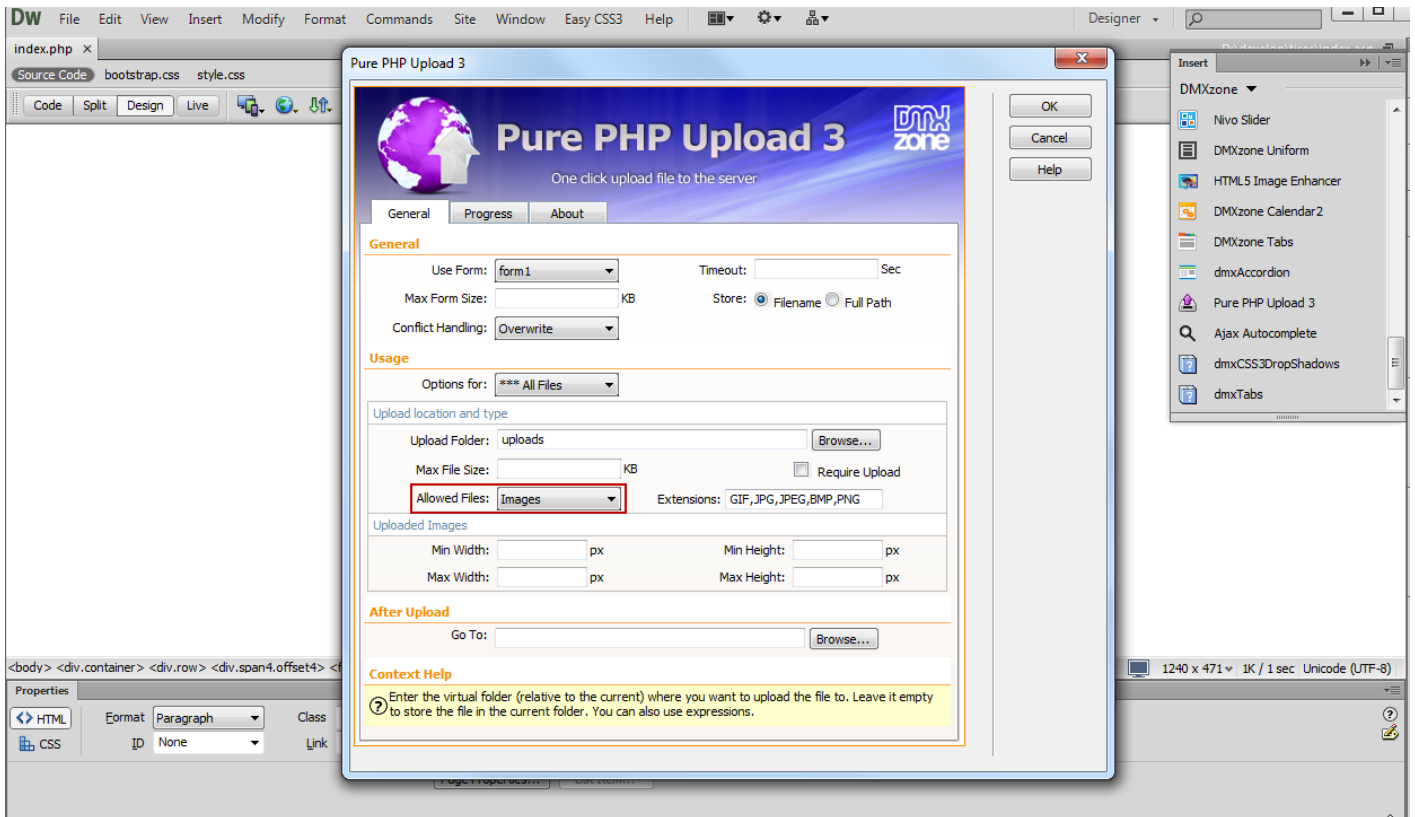
2. Click the **Insert button (1)** and select the **Pure PHP Upload 3 icon (2)** in order the window to appear.



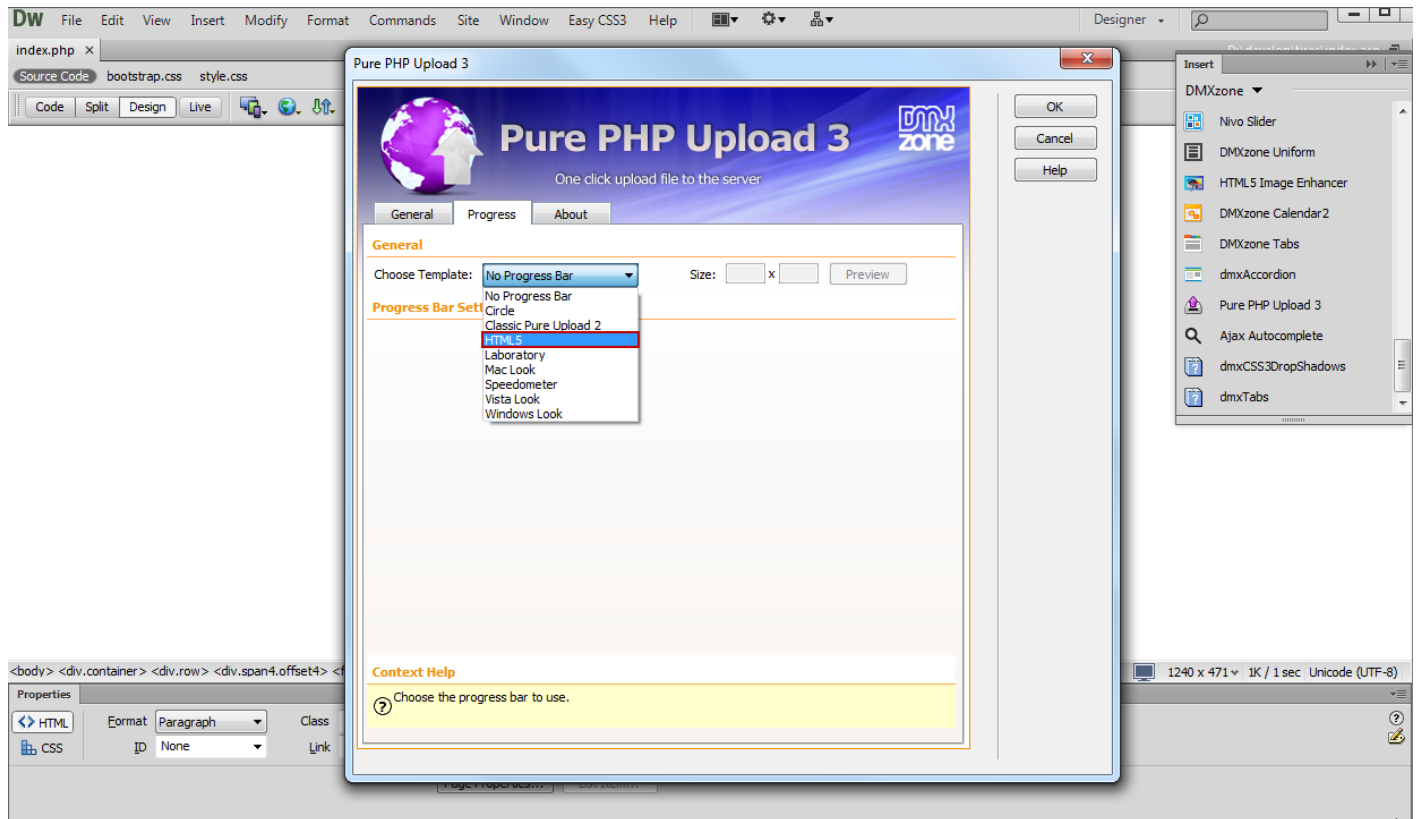
3. We want to store just the **filename (1)** in our database, in a **different folder** than the current one **(2)**.



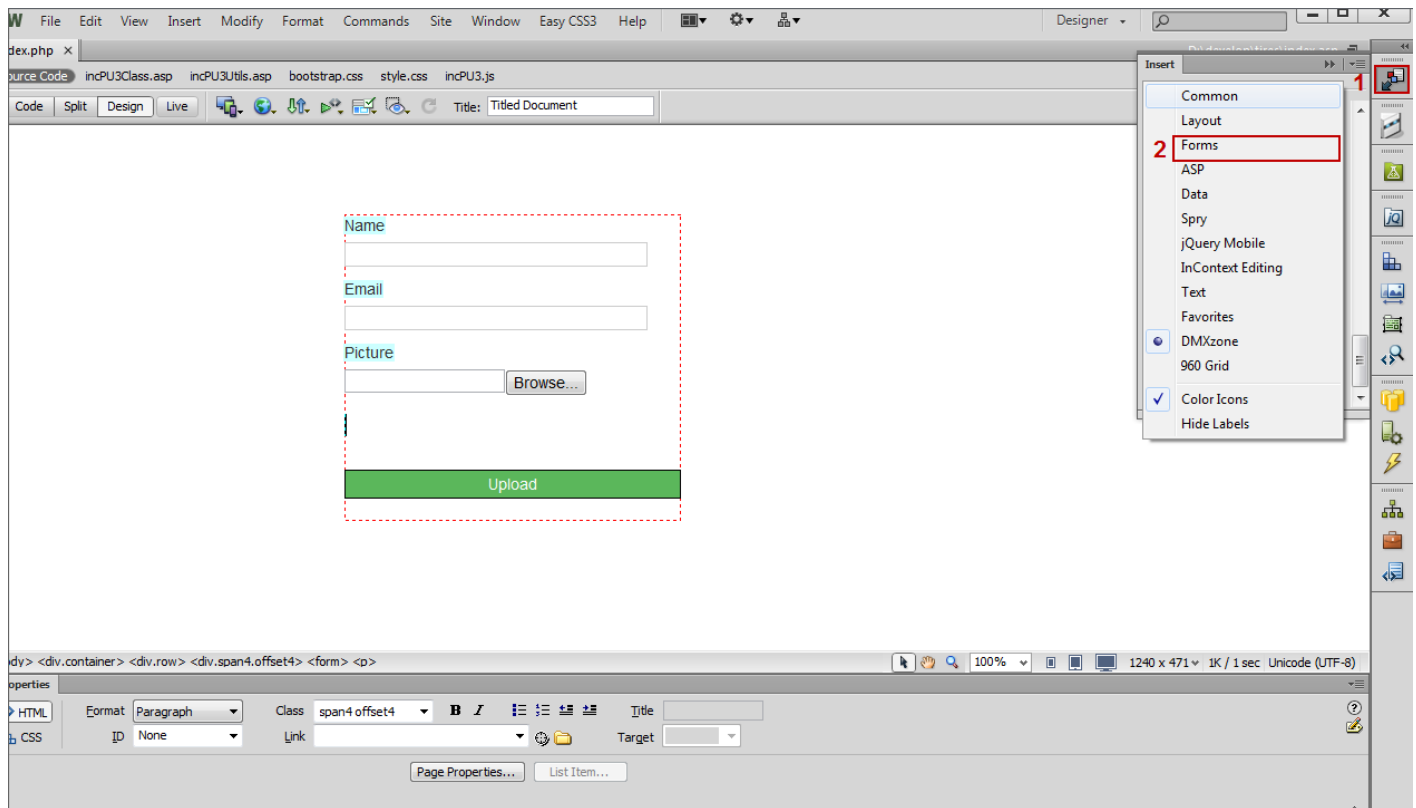
4. For our tutorial we limit the **uploaded file type** only to **images**.



5. We choose the **HTML5 progress bar** from the **Progress tab**.



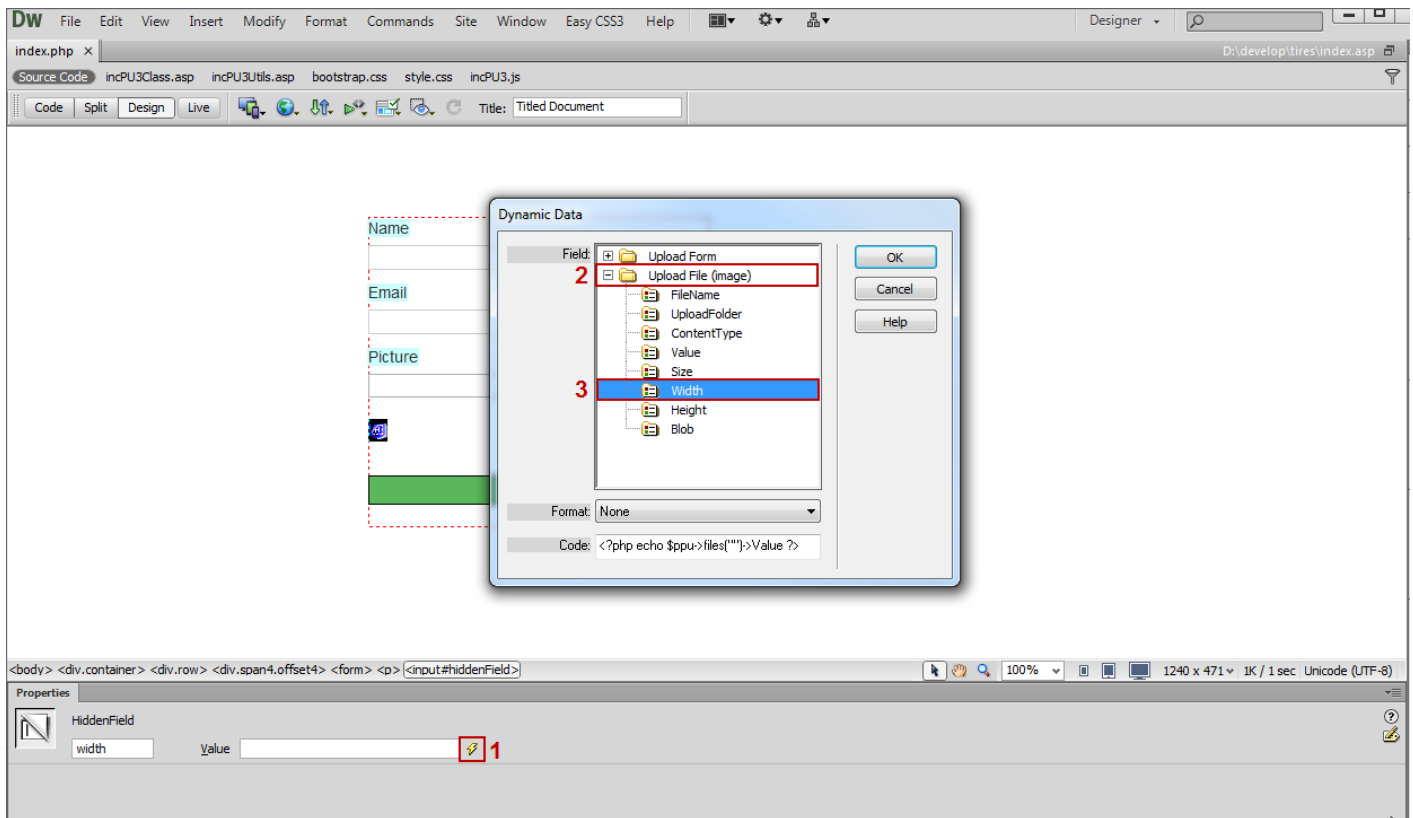
6. Now we are going to **add two hidden fields** that will be used to store the images width and height. From the **Insert menu (1)**, select the **Forms menu item (2)**.



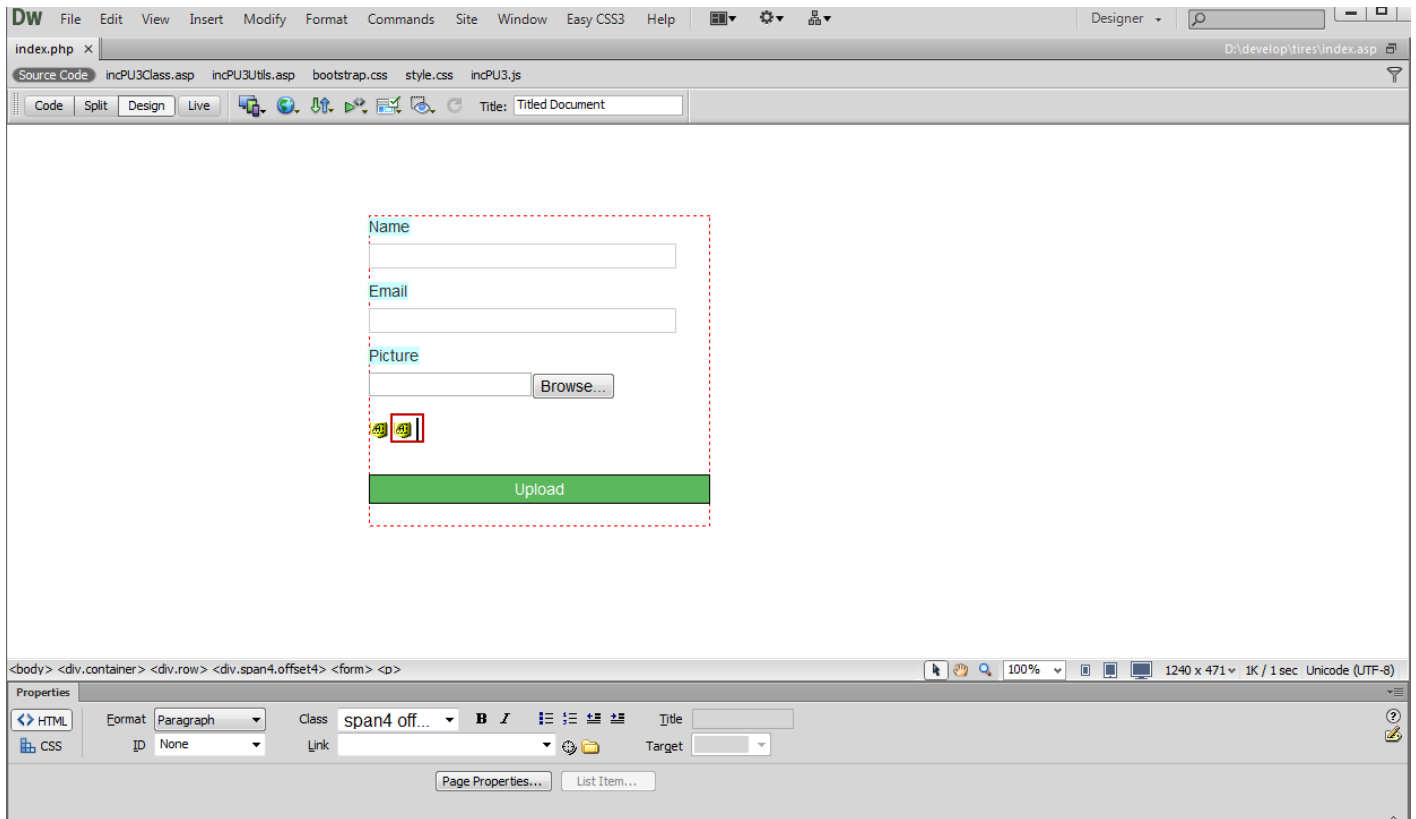
7. Select the **hidden field (1)** and **click on it (2)** once it's added. We give this hidden field a proper name: **width (3)**.

The screenshot shows a web design tool interface. The main workspace displays a form with three input fields labeled "Name", "Email", and "Picture", followed by a "Browse..." button and a green "Upload" button. A red dashed box highlights the "Picture" field and the "Browse..." button. A red box labeled "2" is positioned over a small icon in the workspace. On the right side, the "Insert" panel is open, showing a list of form elements. The "Hidden Field" element is highlighted with a red box and labeled "1". At the bottom, the "Properties" panel shows the "HiddenField" element with the "width" property highlighted in a red box and labeled "3".

- Next, click the **Add dynamic item button (1)**, select **Upload file (2)** and choose the **Width binding (3)**. When you're done, click OK.

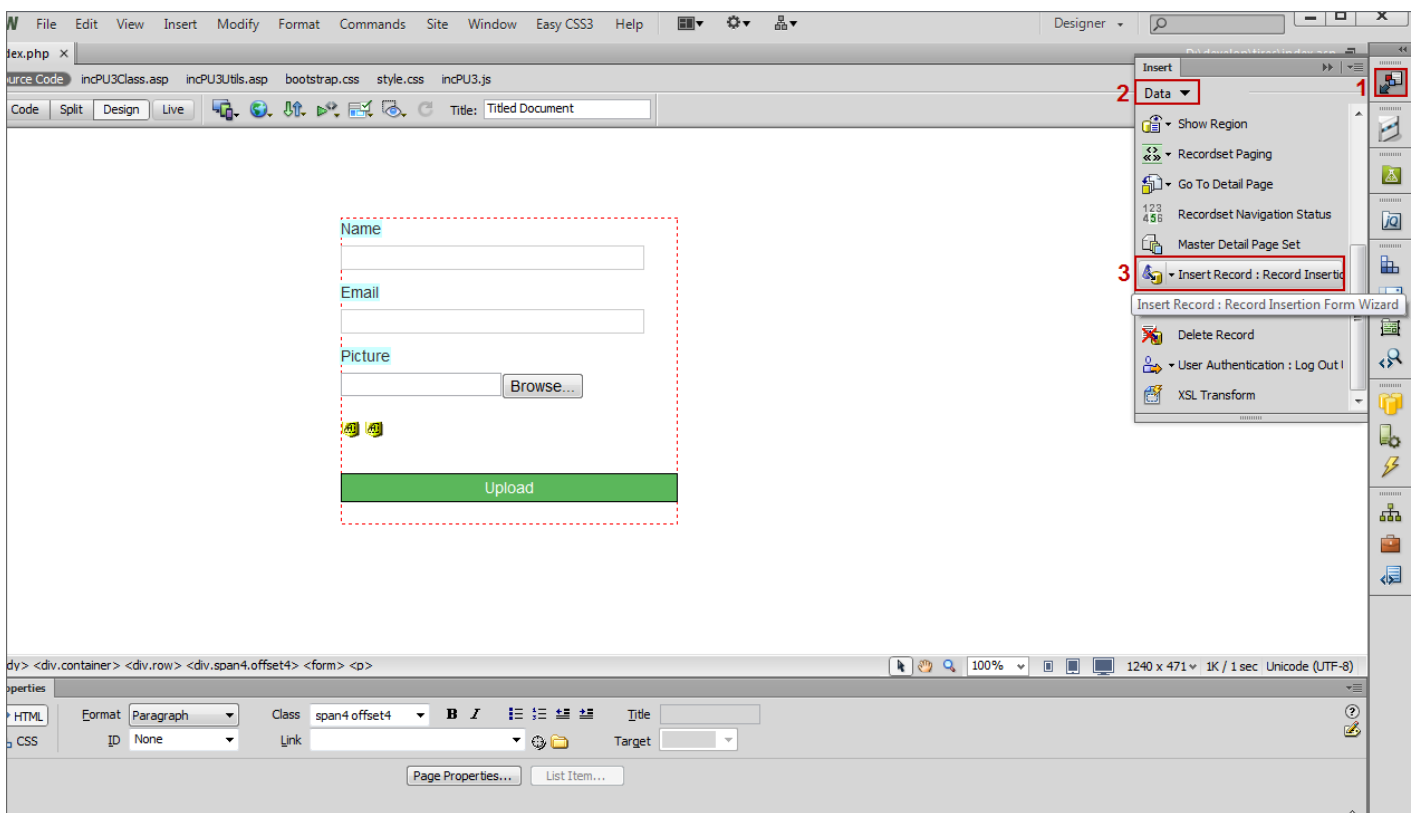


9. We do the same with the **height hidden field**.

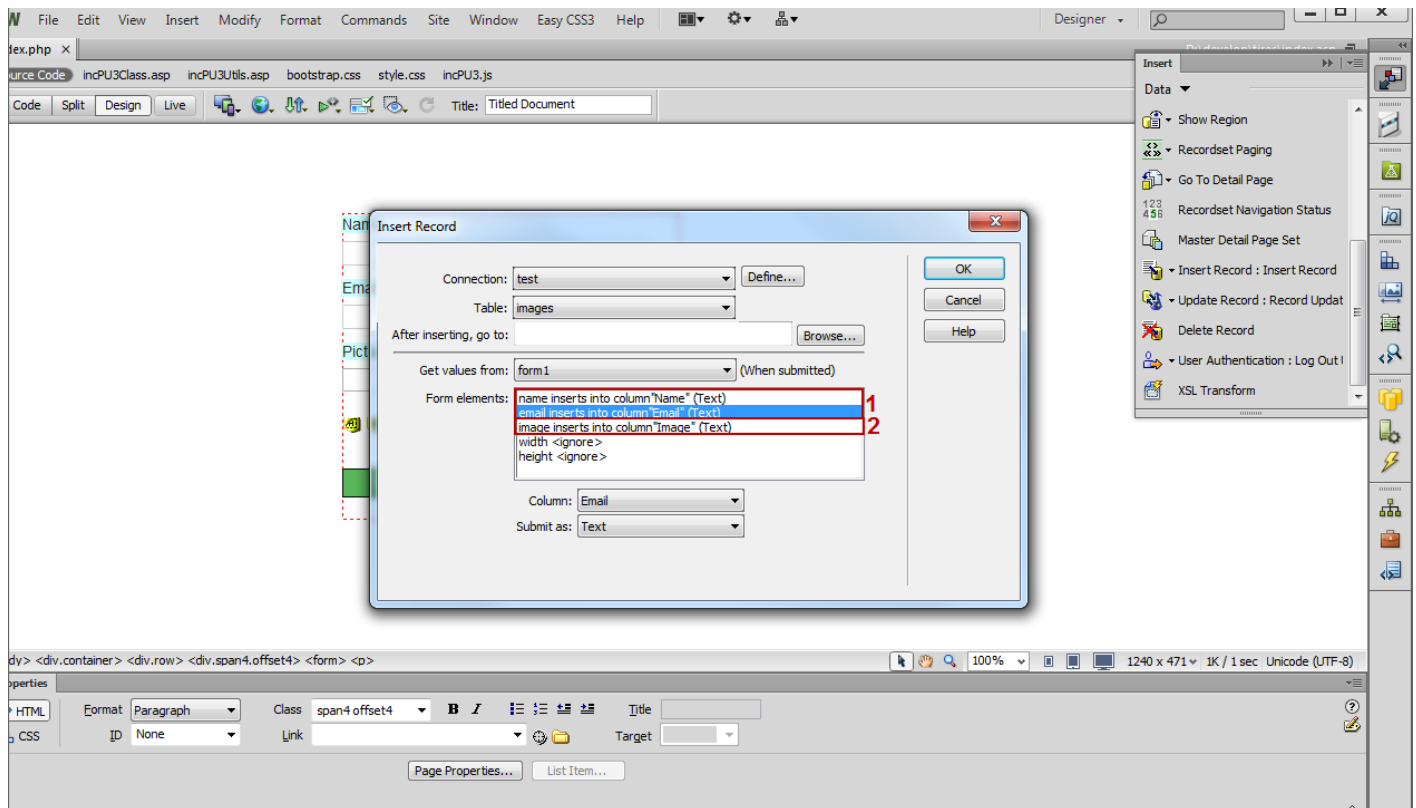




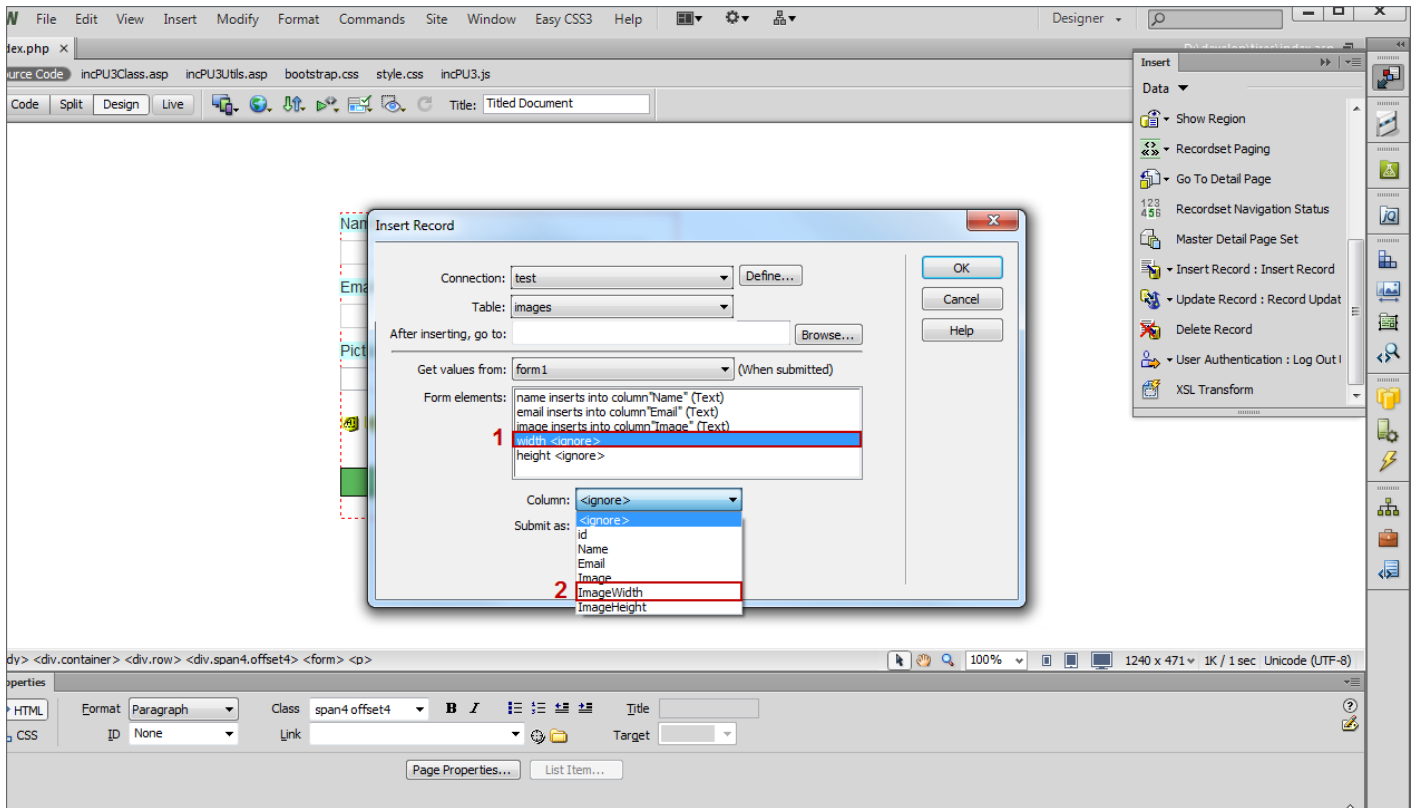
10. Now, we're going to apply the **Insert record behavior**. From the **Insert menu (1)**, choose the **Data menu item (2)** and select **Insert Record (3)**.



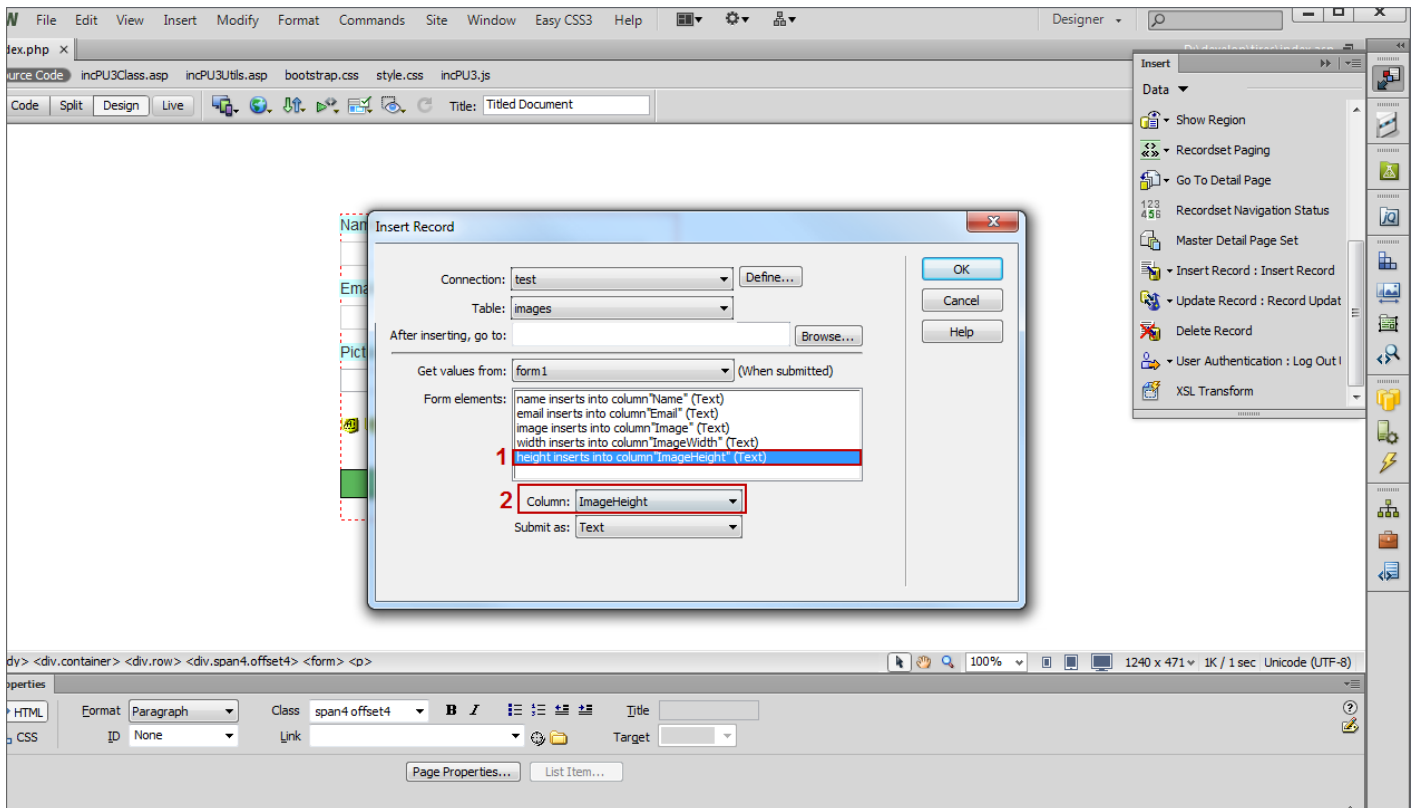
11. In the **Insert record window**, we set the right database columns for our **text fields (1)** and store the **Image name** into a database column named Image **(2)**.



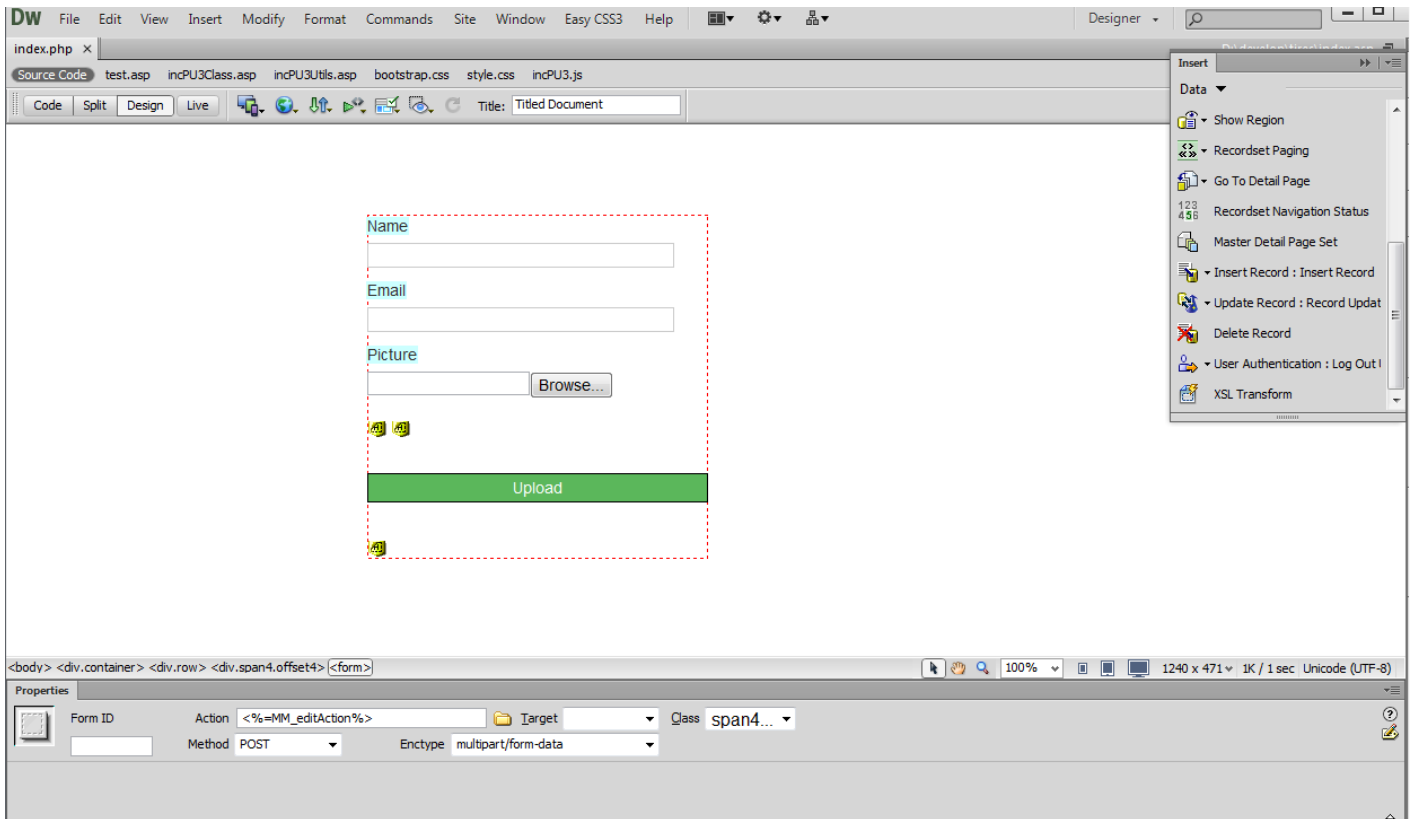
12. In the next step we are going to store the **width hidden field (1)** value into the **ImageWidth database column (2)**.



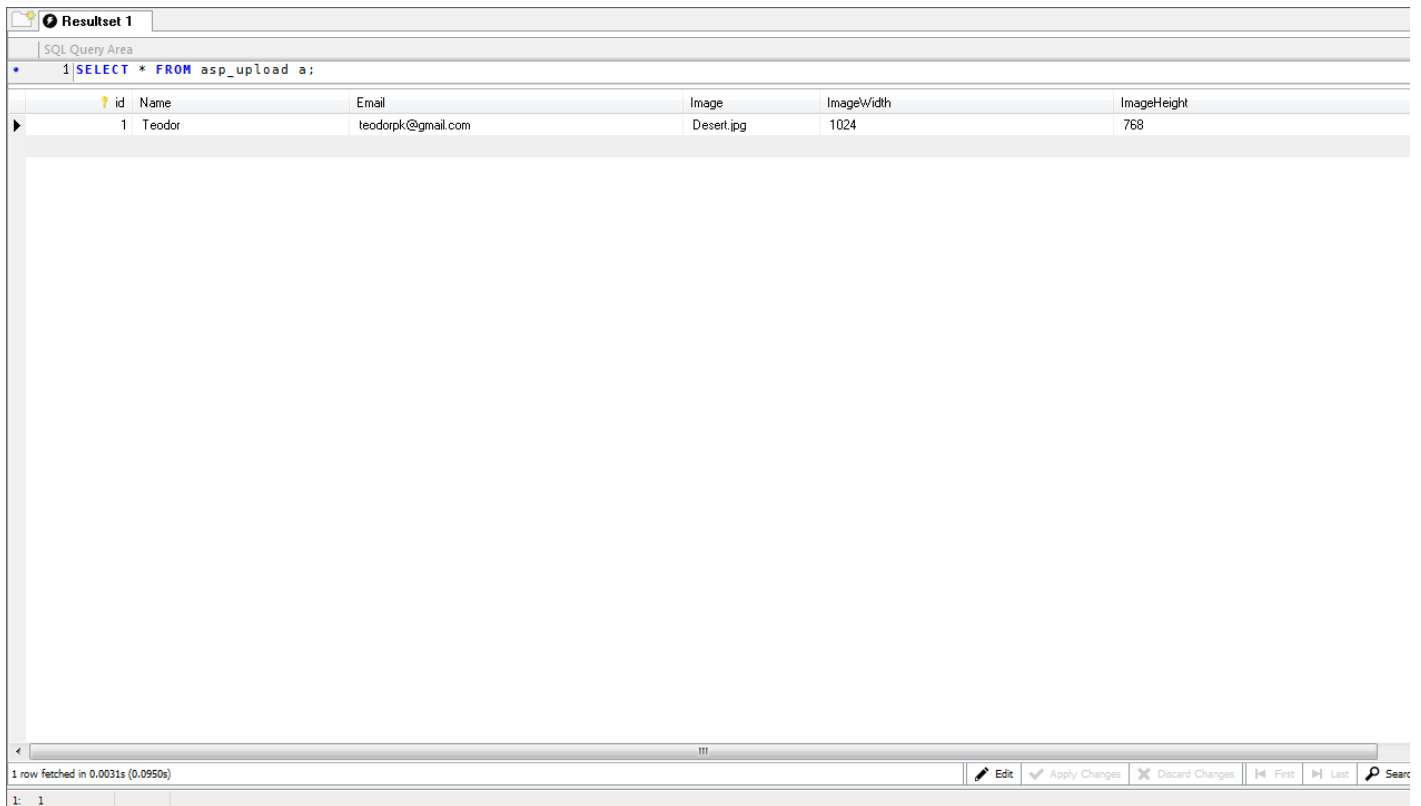
13. We do the same for the **height hidden field**.



14. Now you can save your work and preview the result in the browser.



15. That's how you can upload images and save their names, width and height into a database.



The screenshot shows a database query result set. The SQL query is `SELECT * FROM asp_upload a;`. The result set contains one row with the following data:

id	Name	Email	Image	ImageWidth	ImageHeight
1	Teodor	teodorpk@gmail.com	Desert.jpg	1024	768

At the bottom of the interface, it indicates "1 row fetched in 0.0031s (0.0950s)".

## Advanced: Using Pure PHP Upload 3 with Update Record

In this tutorial we will show you how to use [Pure PHP Upload 3](#) with Update Record server behavior.

\*You can also check out the [video](#).

Name

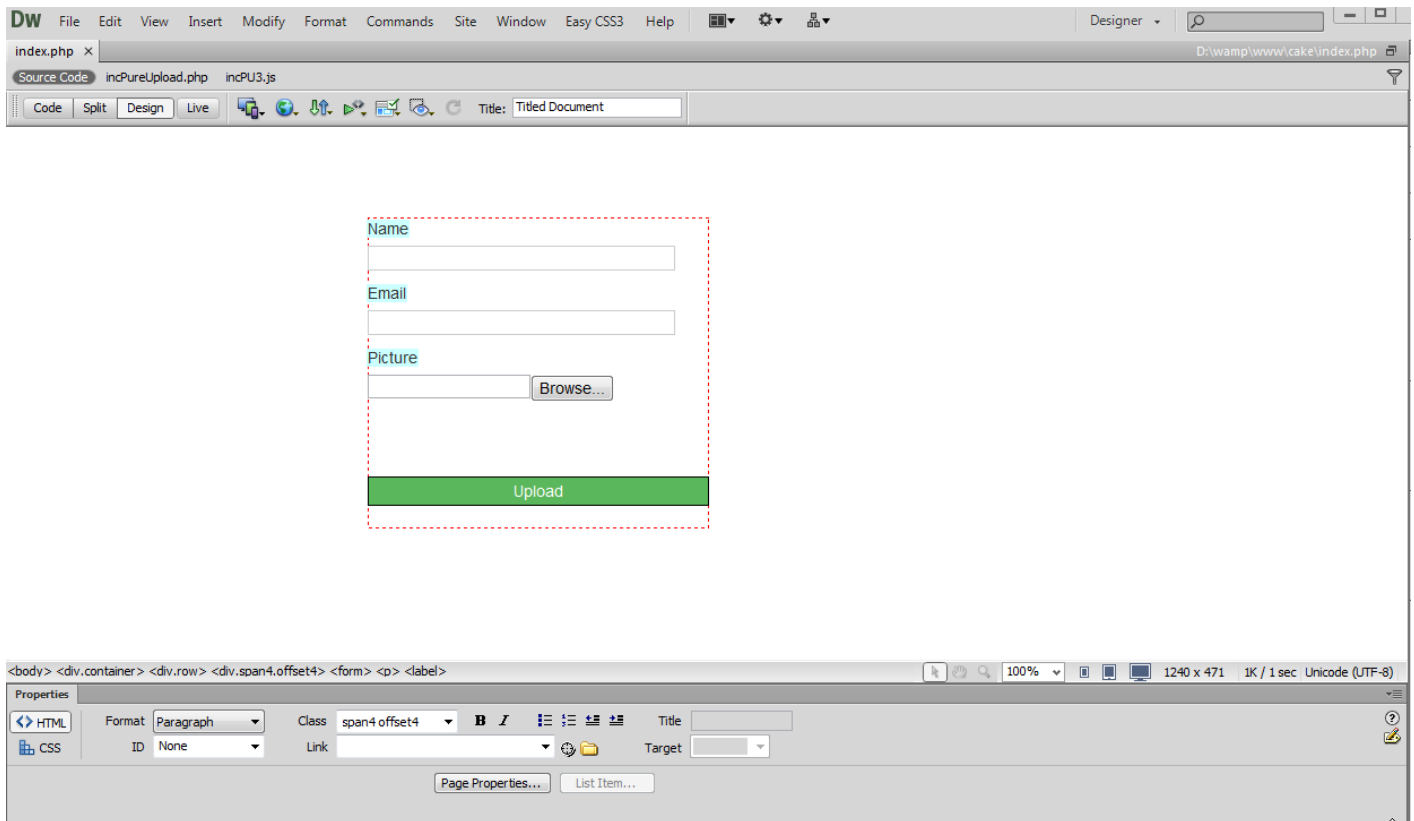
Email

Picture



## How to do it

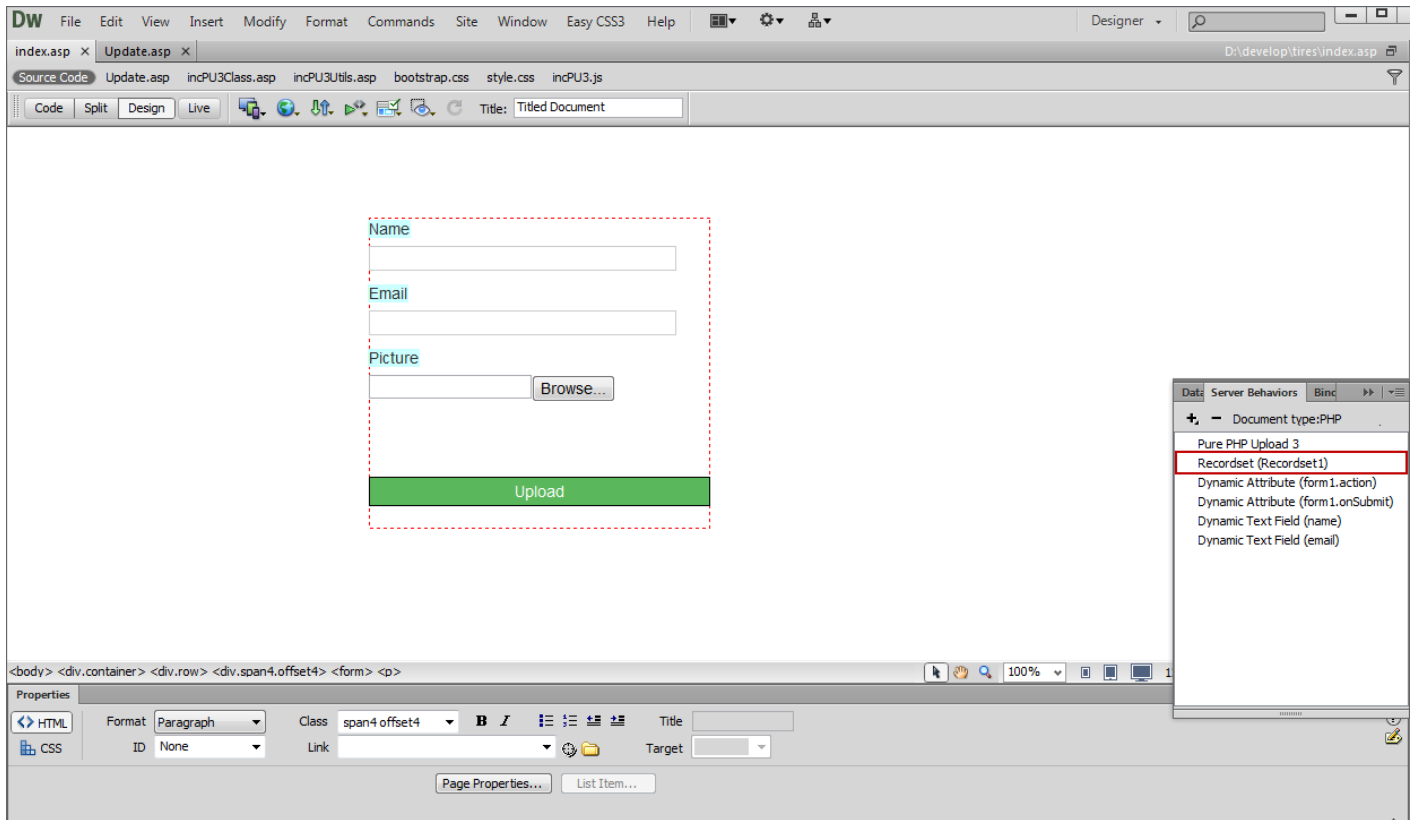
1. We've already created a form that contains three fields – name, email and picture. Now, we're going to **apply the Update record behavior** to this page so we can update the fields' info.



The screenshot displays the Adobe Dreamweaver interface. The main workspace shows a web form with three input fields: "Name", "Email", and "Picture". The "Picture" field includes a "Browse..." button. Below these fields is a green "Upload" button. A red dashed border highlights the entire form area. The top toolbar shows various editing tools, and the bottom Properties panel displays the HTML structure: `<body> <div.container> <div.row> <div.span4.offset4> <form> <p> <label>`. The Properties panel also shows styling options like Format (Paragraph), Class (span4 offset4), and ID (None).



- We've already applied **Pure PHP Upload 3** to this form and created a **standard Dreamweaver recordset**.



3. In the **Recordset** window, we're filtering the recordset by an URL parameter named **ID**.

The screenshot shows the Dreamweaver interface with the Recordset dialog box open. The dialog box is titled "Recordset" and has the following settings:

- Name: Recordset1
- Connection: test
- Table: images
- Columns: All (selected)
- Filter: id = [URL Parameter] id
- Sort: None, Ascending

The Properties panel at the bottom shows the Recordset properties for "Recordset1":

- Connection: Update
- SQL: SELECT \* FROM test.asp\_upload WHERE id = MMCoIPa

The Properties panel also shows a list of dynamic attributes for the Recordset:

- Pure PHP Upload 3
- Recordset (Recordset1)
- Dynamic Attribute (form1.action)
- Dynamic Attribute (form1.onSubmit)

- Next, select the **Name field (1)** and click the **Bindings button (2)**, choose the **Name binding (3)** and click the **Bind button (4)**.

The screenshot shows a web designer interface with a form containing the following elements:

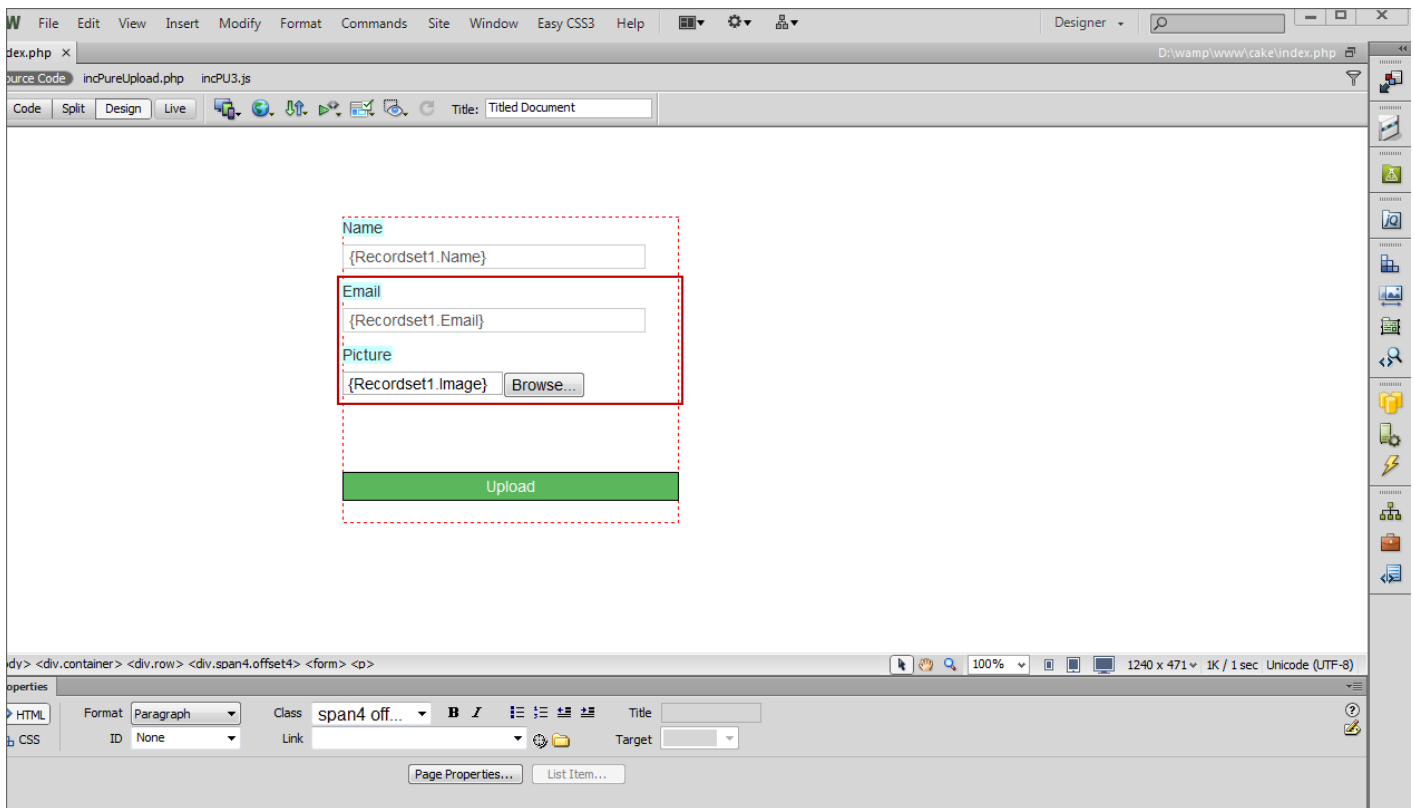
- Name field:** A text input field with a red dashed box around it, labeled '1'.
- Email field:** A text input field.
- Picture field:** A text input field with a 'Browse...' button.
- Upload button:** A green button labeled 'Upload'.

The Bindings panel on the right shows a Recordset (Recordset1) with the following fields:

- id
- Name** (highlighted with a red box and labeled '3')
- Email
- Image
- ImageWidth
- ImageHeight
- [first record index]
- [last record index]
- [total records]

The Bind to: input.value dropdown is set to 'Bind' (labeled '4').

5. We do the same for **the Email and Picture fields**, choosing the corresponding bindings - email and image.



6. Now we are going to apply the **Update Record** behavior. Click the **Server behavior** button (1), select the **Add item** button (2) and choose **Update record** menu item (3).

The screenshot shows a web design tool interface. The main workspace displays a form with three input fields: "Name" (containing "{Recordset1.Name}"), "Email" (containing "{Recordset1.Email}"), and "Picture" (containing "{Recordset1.Image}" and a "Browse..." button). Below these fields is a green "Upload" button. A red dashed box highlights the form fields. The "Server Behaviors" panel on the right is open, showing a list of behaviors. A red box labeled "1" highlights the "Add" button (+) at the top of the panel. A red box labeled "2" highlights the "Add" button (+) at the top of the "Recordset (Query)" section. A red box labeled "3" highlights the "Update Record" menu item in the list.

Code: incPureUpload.php incPU3.js

Code Split Design Live

Title: Titled Document

Name  
{Recordset1.Name}

Email  
{Recordset1.Email}

Picture  
{Recordset1.Image} Browse...

Upload

Server Behaviors

Document type:PHP

Recordset (Query)

Command

Repeat Region

Recordset Paging

Show Region

Dynamic Text

Go To Detail Page

Go To Related Page

Insert Record

Update Record

Delete Record

Dynamic Form Elements

User Authentication

DMXzone

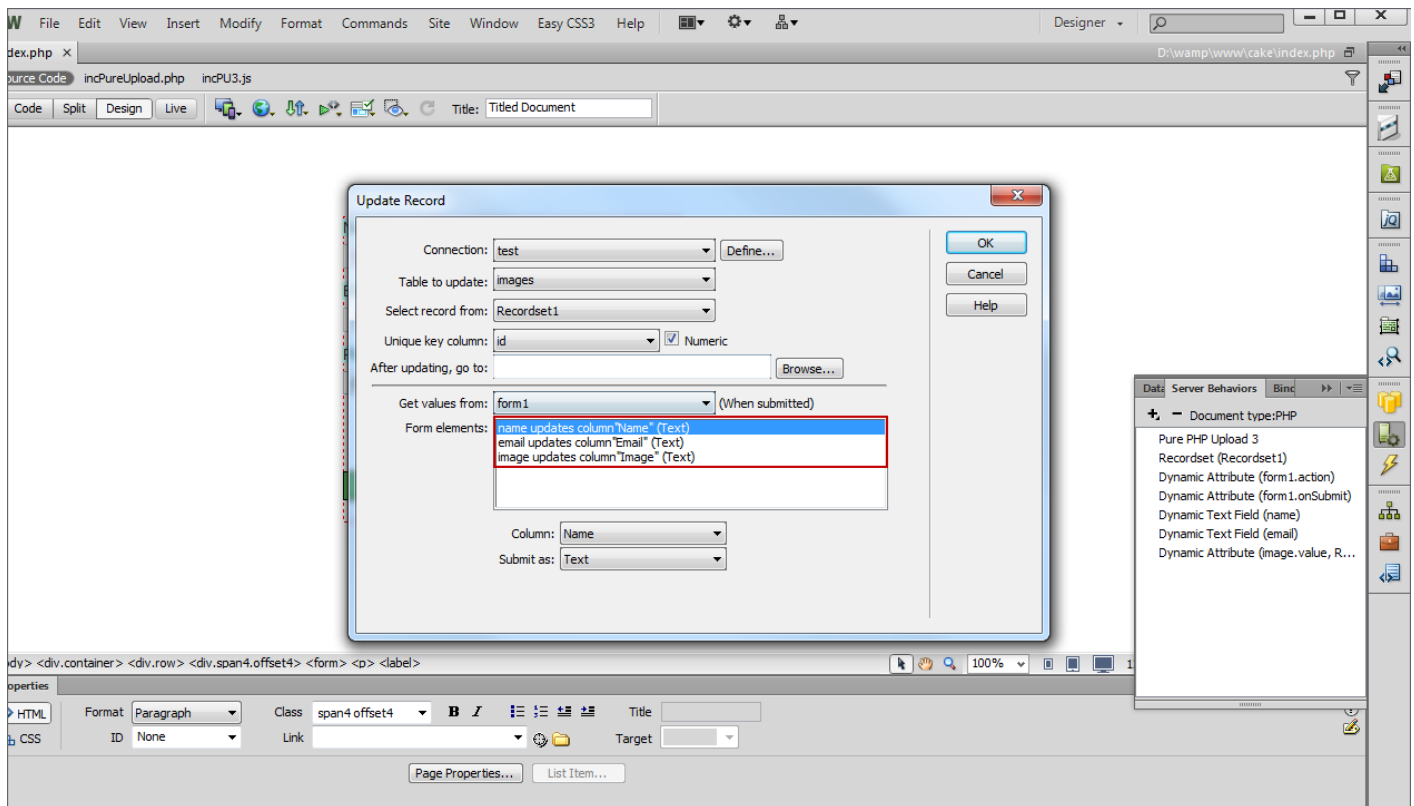
XSLT Transformation

HTML Format Paragraph Class span4 offset4 B I

CSS ID None Link

Page Properties... List Item...

7. In the Update record window, make sure that your form fields are assigned to the corresponding fields in your database and click the OK button.



8. Now you can save your work and preview the result in the browser.

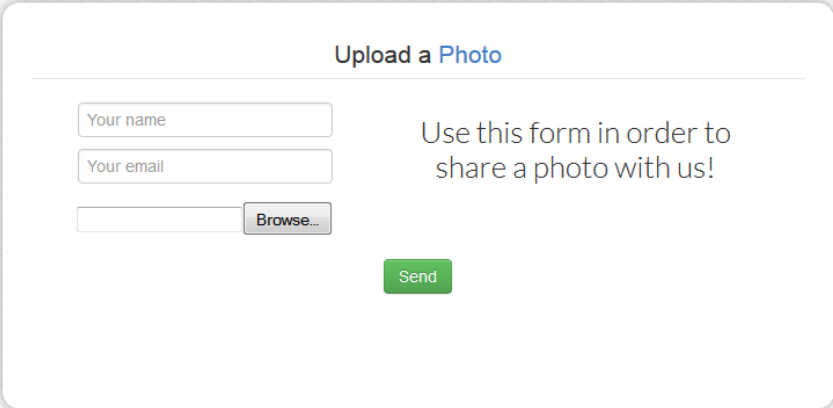
The screenshot displays a web design application window. The main workspace shows a form with three input fields: "Name" containing the dynamic text "{Recordset1.Name}", "Email" containing "{Recordset1.Email}", and "Picture" containing "{Recordset1.Image}" and a "Browse..." button. A green "Upload" button is positioned below these fields. A red dashed box highlights the form elements. The application's interface includes a menu bar (File, Edit, View, Insert, Modify, Format, Commands, Site, Window, Easy CSS3, Help), a toolbar, and a properties panel at the bottom. The properties panel shows a "Recordset" named "Recordset1" with a connection of "Update" and an SQL query: "SELECT \* FROM test.asp\_upload WHERE id = MMC0Pa". A right-hand sidebar contains a "Server Behaviors" panel with a tree view listing: "Pure PHP Upload 3", "Recordset (Recordset1)", "Command (Recordset1\_cmd)", "Update Record (Recordset1, form \*...", "Dynamic Text Field (name)", "Dynamic Text Field (email)", and "Dynamic Attribute (image.value, R...".

## Advanced: Creating an Ajax Upload Page

In this tutorial we will show you how to use [Pure PHP Upload 3](#) with [DMXzone Ajax Form](#) in order to create an Ajax Upload Page.

\*You can also check out the [video](#).

SHARE YOUR PHOTO



Upload a **Photo**

Your name

Your email

Use this form in order to share a photo with us!

Don't have an account yet?

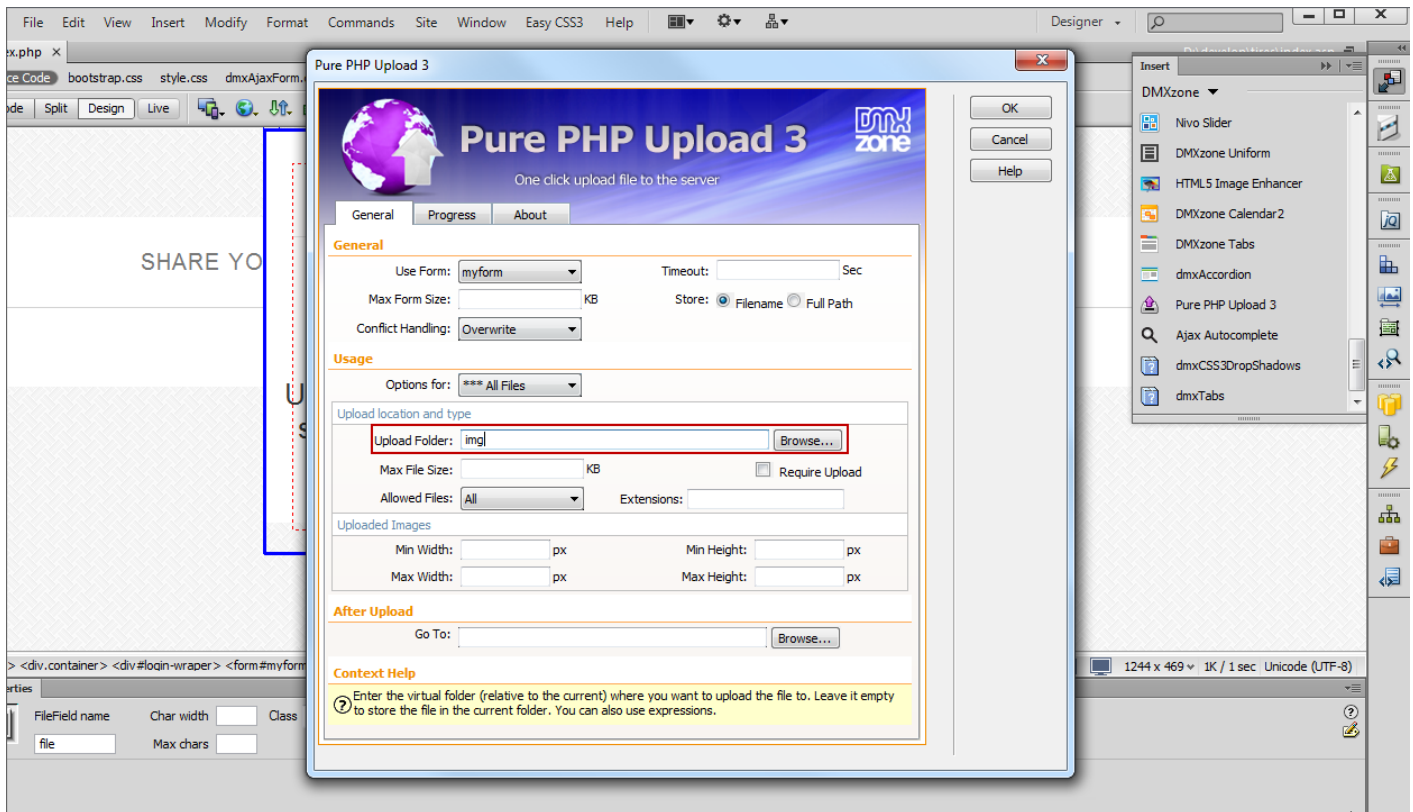


## How to do it

1. We created a simple form, containing two text fields and one file field. First we are going to insert **Pure PHP Upload 3**. Click the **Insert button (1)** and choose the **Pure PHP Upload 3 icon (2)**.

The screenshot shows a web design tool interface. The main workspace displays a form titled "Upload a Photo" with a "Send" button. The form contains two text input fields and a file input field with a "Browse" button. The text "Use this form in order to share a photo with us!" is positioned below the input fields. The "Insert" panel on the right side of the tool shows a list of widgets, with "Pure PHP Upload 3" highlighted. The top menu bar includes options like File, Edit, View, Insert, Modify, Format, Commands, Site, Window, Easy CSS3, and Help. The bottom status bar shows the current HTML structure: `<div.container> <div#login-wrapper> <form#myform.form.login-form> <div.row> <div.span4> <input#file>`.

2. In the Pure PHP Upload 3 window, **set your upload folder** and click the Ok button.



3. Now, **select your form (1)**, click the **Insert button (2)** and choose the **DMXzone Ajax Form (3)**.

The screenshot shows a web design software interface with a central canvas and a right-hand 'Insert' panel. The canvas displays a form titled 'Upload a Photo' with a text input, a 'Browse...' button, and a 'Send' button. A red dashed box highlights the form, and a blue solid box highlights the 'Insert' panel. The 'Insert' panel lists various widgets, with 'Pure PHP Upload 3' selected and highlighted in red. The 'Properties' panel at the bottom shows the form's configuration: Form ID 'myform', Action '<%=DMX\_uploadAction%>', Method 'POST', and Enctype 'multipart/form-data'.

DMXzone

- Online Page Editor Add-on
- Advanced HTML Editor 3
- dmxSlidingPanels
- Sliding Billboard
- HTML5 Video Player
- HTML5 3D Flipping Gallery
- Pure PHP Upload 3**
- DMXzone Ajax Form
- Ajax Autocomplete
- dmxCSS3DropShadows

Upload a Photo

USE YOUR PHOTO

Use this form in order to share a photo with us!

Send

<div.container> <div#login-wrapper> <form#myform.form.login-form>

Properties

Form ID	Action	Target	Class
myform	<%=DMX_uploadAction%>		form logi...

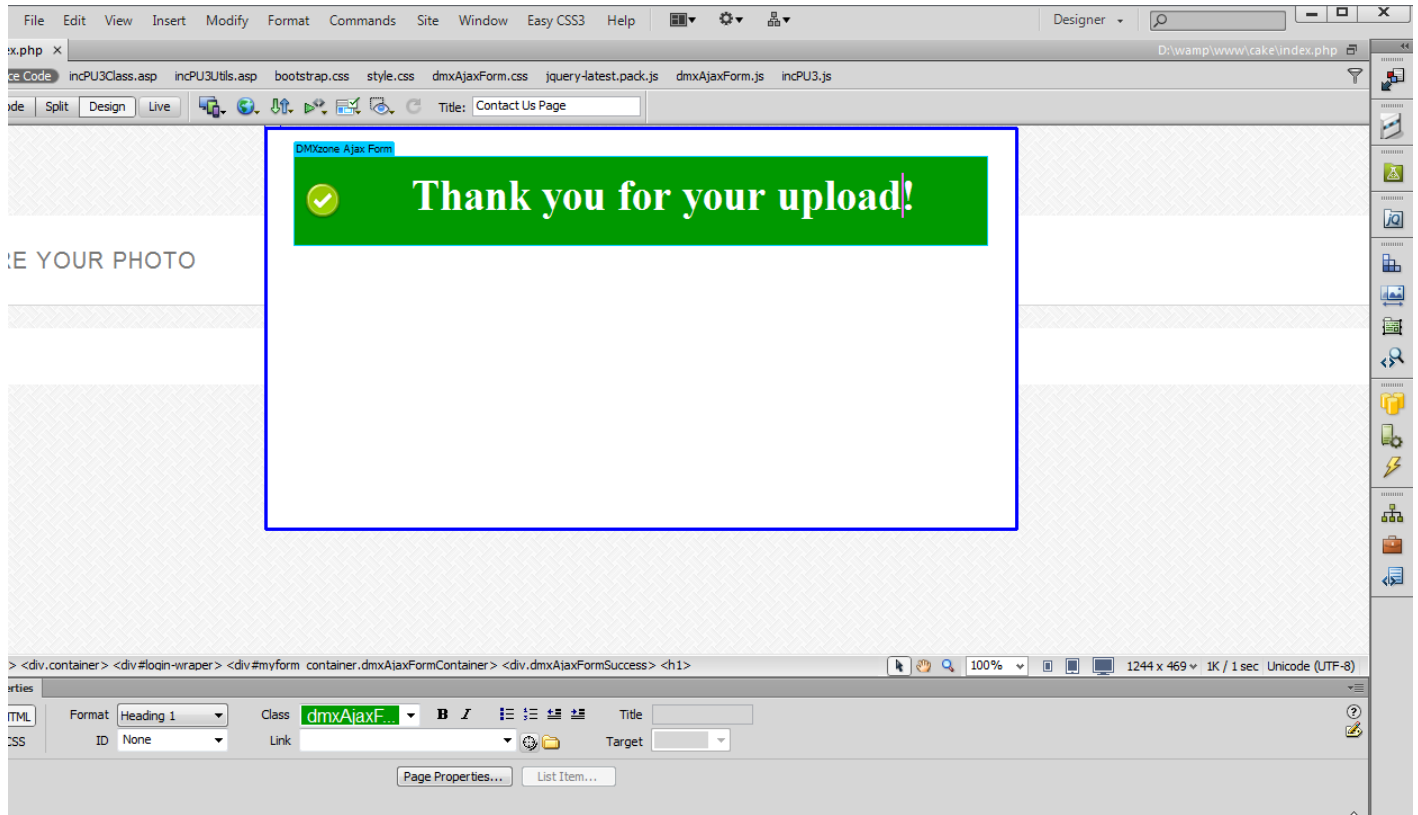
Method: POST  
Enctype: multipart/form-data

4. We are going to change the default Success message. **Mark it from the select section to edit (1)** and **write your custom message (2)**.

The screenshot shows a web designer interface with a form titled "DMXzone Ajax Form". The form is displayed on a page with the text "YOUR PHOTO". The form has a green success message "Thank you!" with a checkmark icon. A red box highlights the message, and a number "2" is next to it. A tooltip shows the form's properties: ID: myform\_container, Class: dmxAjaxFormContainer, Tag: div, Float: Top, Position: Top, Right: Right, Bottom: Bottom, Left: Left, Width: Width, Height: Height, Overflow: Overflow.

At the bottom, the configuration panel for the "DMXzone Ajax Form" is visible. It includes fields for Action, Enctype, Form Method, and On Submit. A red box highlights the "Select Section to Edit" dropdown menu, which has "Form" selected. A number "1" is next to the dropdown. The configuration panel also includes "Advanced" options like Set Target, Get Response, Hide Form, and Clear Form.

5. Now you can save your work and preview the results in your browser.



## Advanced: Uploading Files to a Dynamic Folder

In this tutorial we will show you how to upload files to a Dynamic folder using [Pure PHP Upload 3](#).

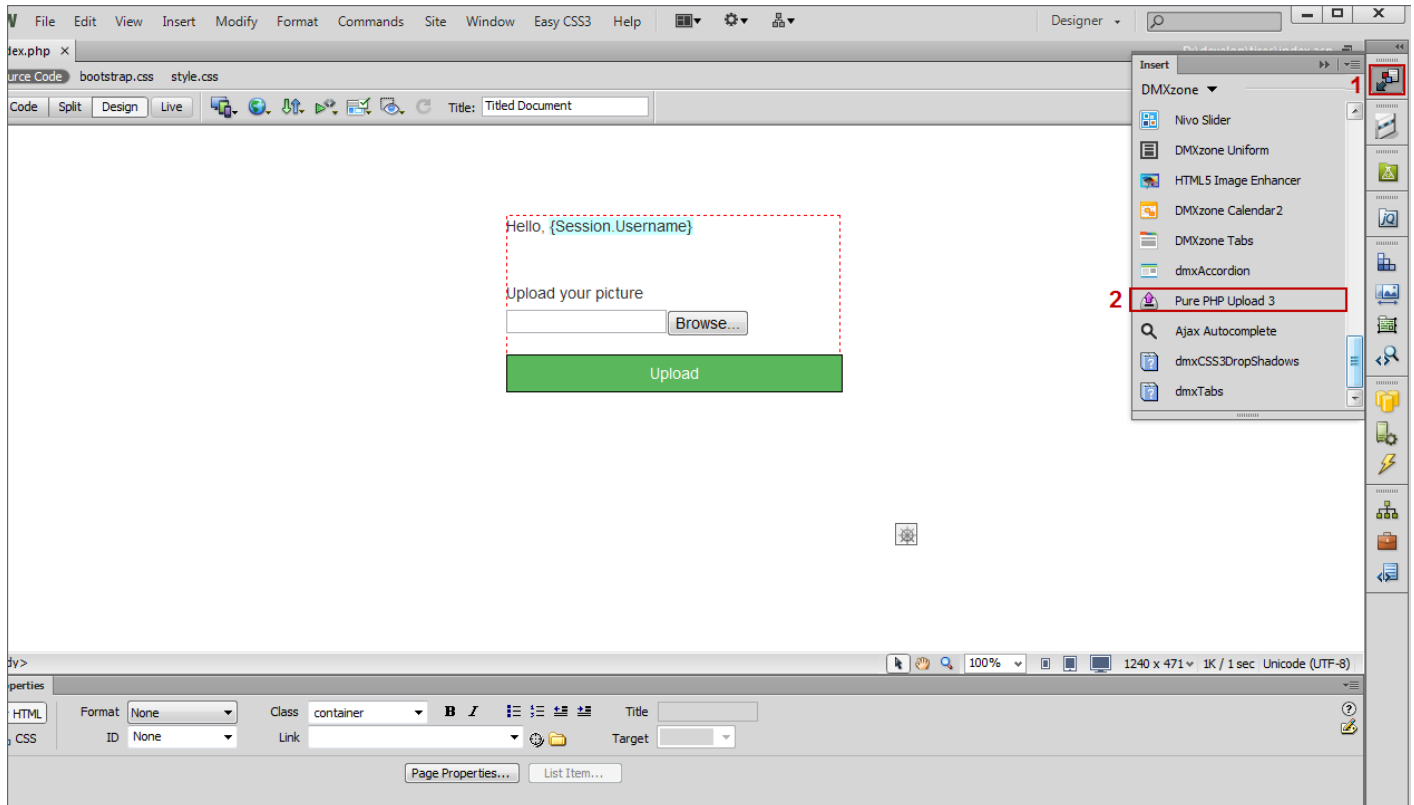
\*You can also check out the [video](#).

Hello, **Teodor**

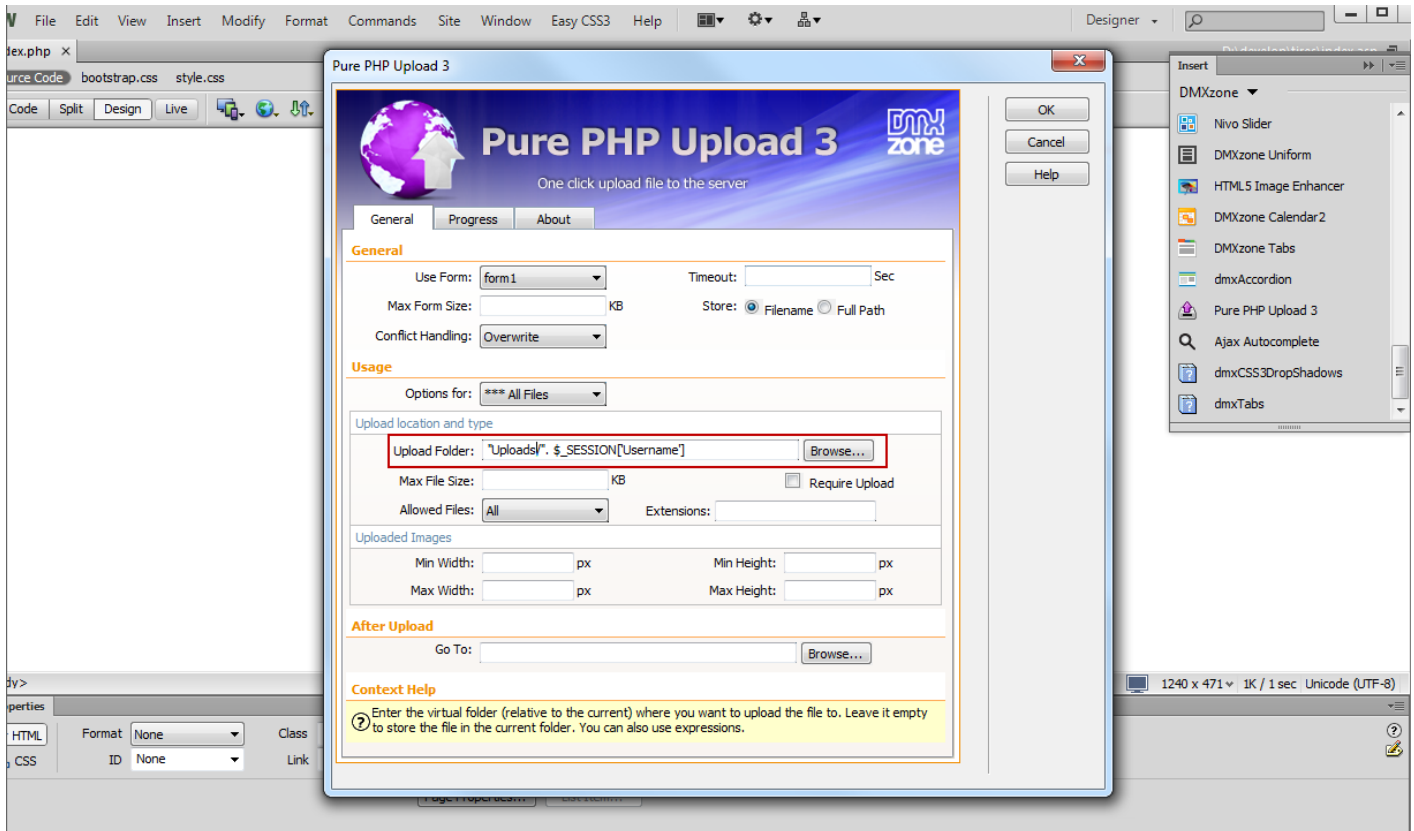
Upload your picture

## How to do it

1. For this tutorial we use a **session variable** that stores a username. We are going to use the username as a folder name. Click the **Insert button (1)** and select the **Pure PHP Upload 3 icon (2)** to open its UI.

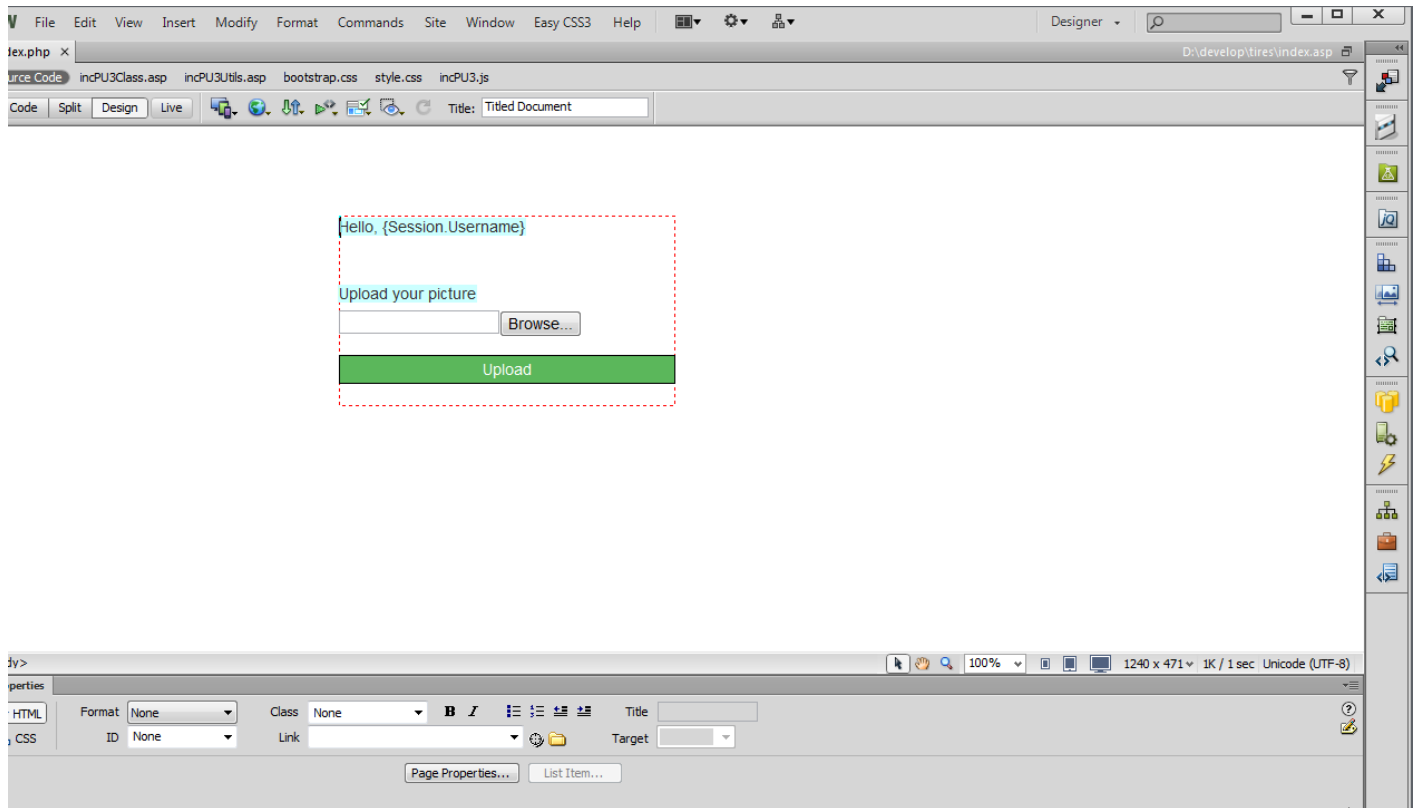


- Our upload folder is named **"Uploads"** and we want to create a folder for each user inside it. Enter: **"Uploads/" . \$\_SESSION ('Username')**. Username is the session variable that stores the username. This will create a new folder for every user that tries to upload. Non-existing folders will be auto-created





3. Now you can save your work and preview the result in the browser.



## Advanced: Uploading Large Files

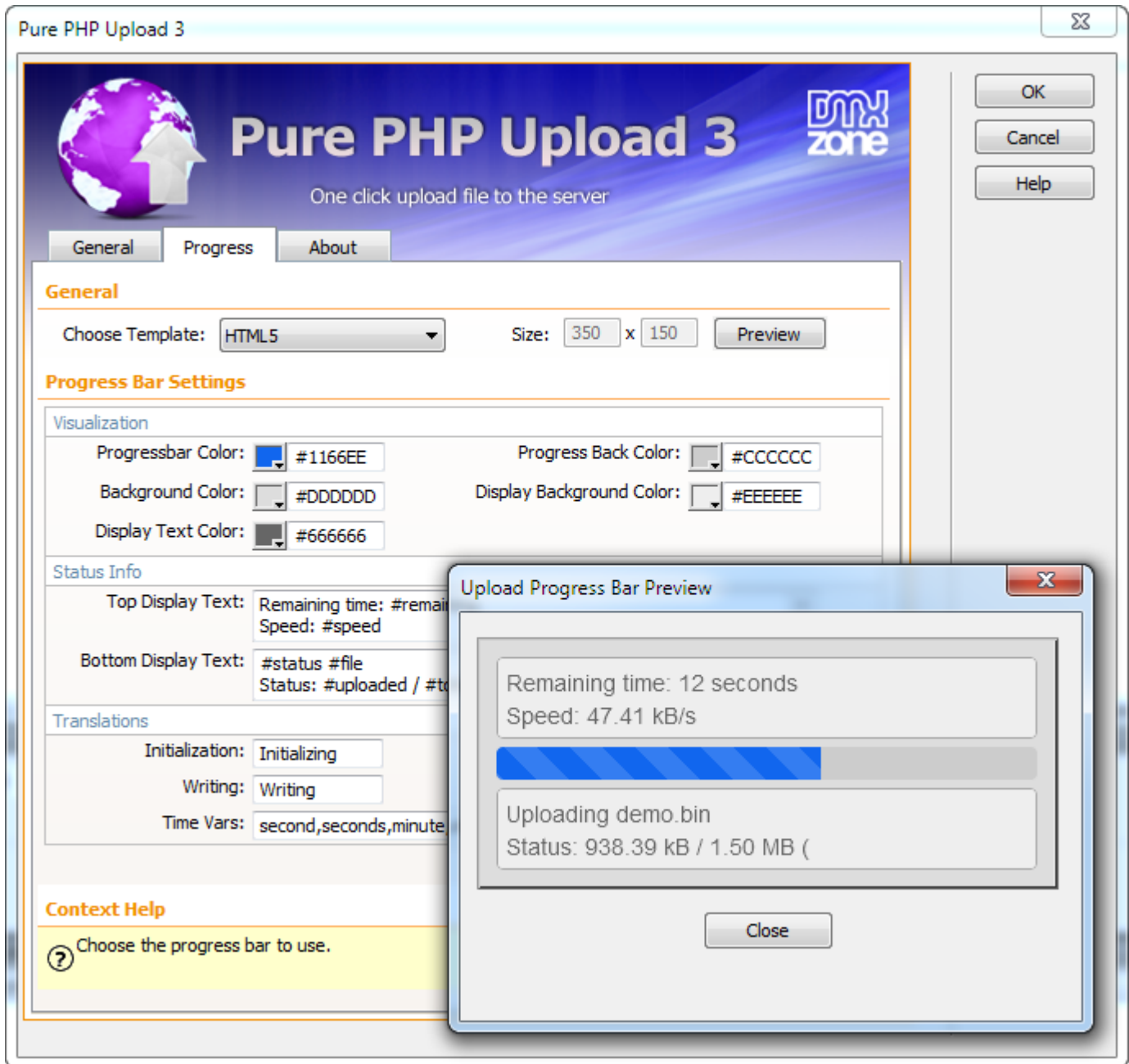
There are some limitations that are set in the **php configuration (PHP.INI)**. In this article we will show you common pitfalls.

## How to do it

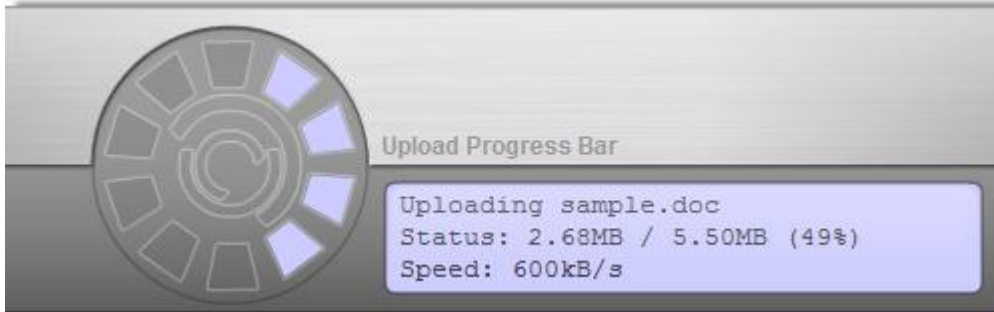
- The **MAX\_FILE\_SIZE** item cannot specify a file size greater than the file size that has been set in the `upload_max_filesize` ini-setting. The default is 2 Megabytes.
- If **memory limit** is enabled, larger `memory_limit` may be needed. Make sure to set `memory_limit` large enough.
- If **max\_execution\_time** is set too small, script execution may be exceeded the value. Make sure to set `max_execution_time` large enough.
- If **post\_max\_size** set too small, large files cannot be uploaded. Make sure to set `post_max_size` large enough.
- Please note that the CERN httpd seems to strip off everything starting at the first whitespace in the content-type mime header it gets from the client. As long as this is the case, CERN httpd will not support the file upload feature.
- Due to the large amount of directory listing styles we cannot guarantee that files with exotic names (like containing spaces) are handled properly.

## Reference: Customizable Progress Bar

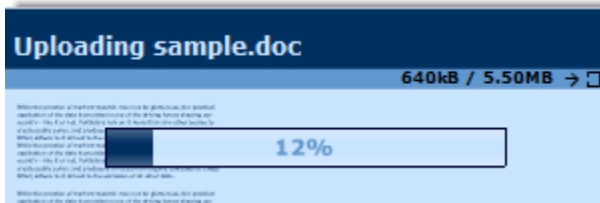
The build-in Dreamweaver menu enables you to select multiple predefined Progress bars and customize them. You can preview your progress bar design directly in Dreamweaver by clicking the “preview” button.



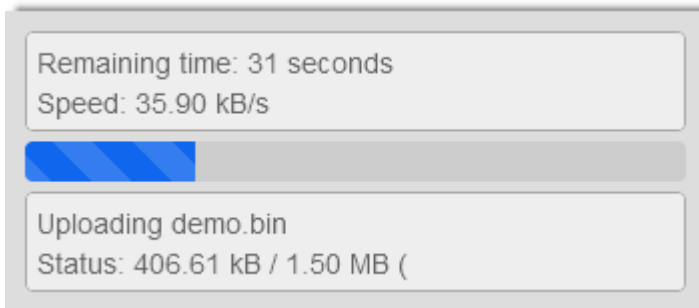
## Circle



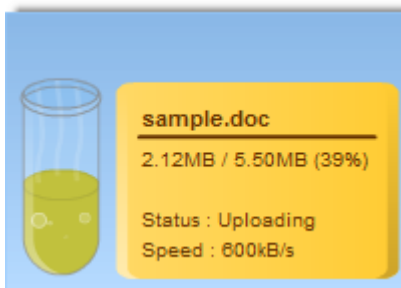
## Classic Pure Upload 2



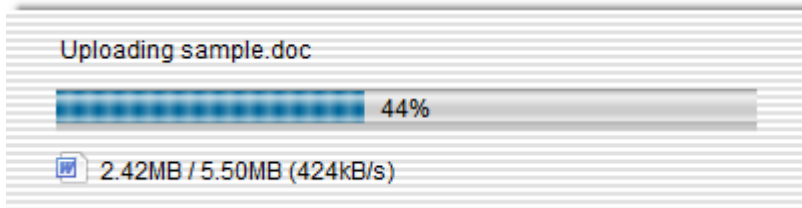
## HTML5



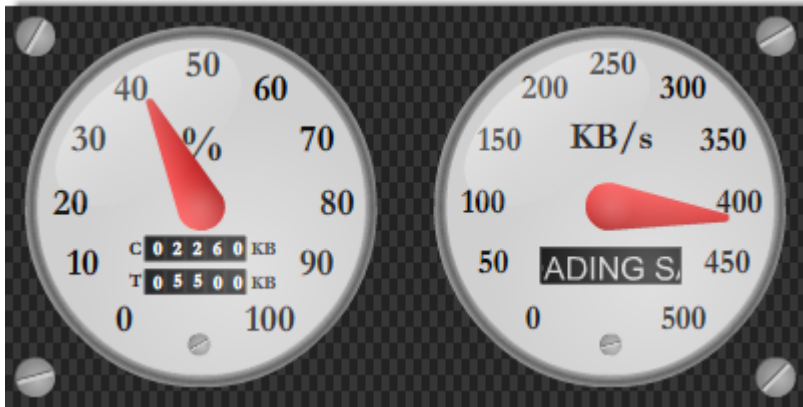
## Laboratory



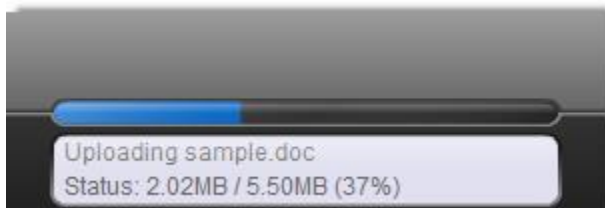
## Mac Look



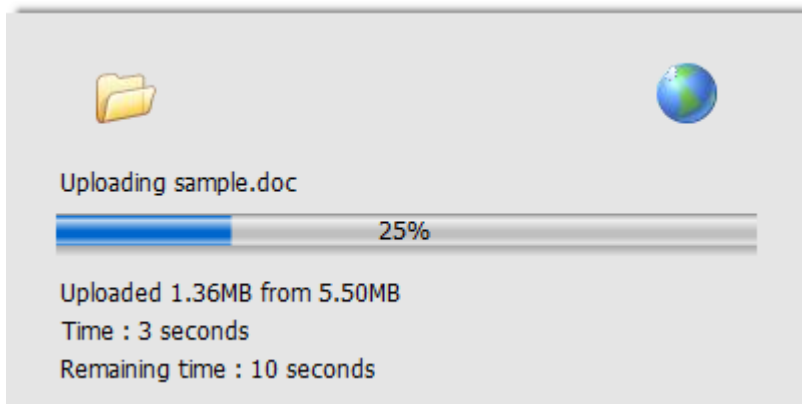
## Speedometer



## Vista Look



## Windows Look



## Video: Uploading Files with Pure PHP Upload 3



With this [movie](#) we will show you how to upload files using [Pure PHP Upload 3](#).

## Video: Using Pure PHP Upload 3 with Insert Record



With this [movie](#) we will show you how to upload images and save their names, width and height into a database.

## Video: Using Pure PHP Upload 3 with Update Record



With this [movie](#) will show you how to use [Pure PHP Upload 3](#) with Update Record server behavior.

## Video: Uploading Files to a Dynamic Folder



With this [movie](#) we will show you how to upload files to a Dynamic folder using [Pure PHP Upload 3](#).

## Video: Creating an Ajax Upload Page



With this [movie](#) we will show you how to use [Pure PHP Upload 3](#) with [DMXzone Ajax Form](#) in order to create an Ajax Upload Page.