# Lab: Media Files

# Pythons Wiki (with Image Upload)

**Refactor** the project from the previous lecture, so instead of **link** to an image of a python, it works with **file upload**.

### Steps

* Create a folder called **'media'** and a folder **'images'** in it
* Add the needed **configuration** for the media files to work
* Add **enctype="multipart/form-data"** to your form
* Change the **image field** in the Python **model** to **ImageField()**
* Add appropriate **styling** to the field using bootstrap (as shown below)

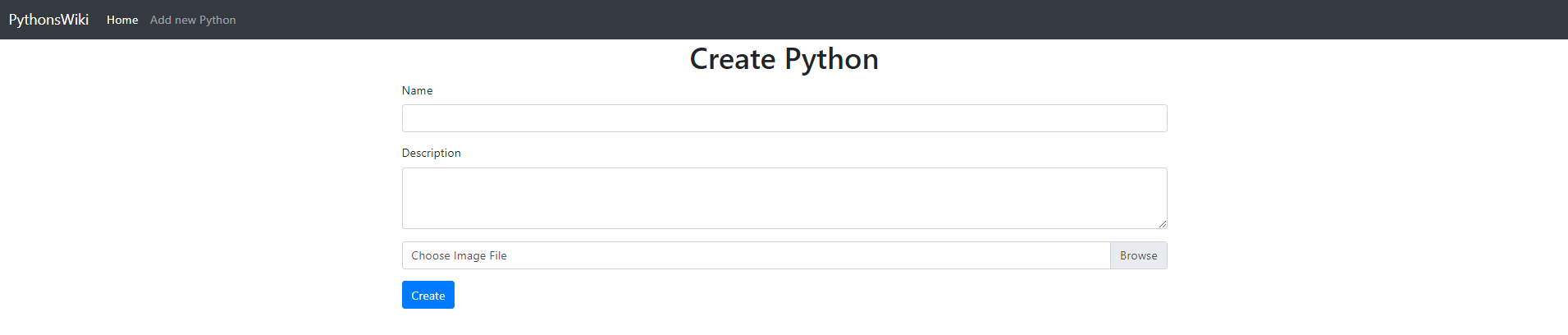
<**div class="custom-file"**>  
 <**label class="custom-file-label" for="image"**>Choose Image File</**label**>  
 {{ **form**.**image** }}  
</**div**>  
<**input type="submit" value="Create" class="btn btn-primary mt-3"**>

**class** PythonCreateForm(forms.ModelForm):  
 **class** Meta:  
 model = Python  
 widgets = {  
 **'name'**: forms.TextInput(attrs={**'class'**: **'form-control'**}),  
 **'description'**: forms.Textarea(attrs={**'class'**: **'form-control'**, **'rows'**: **'3'**}),  
 **'image'**: forms.FileInput(attrs={**'class'**: **'custom-file-input'**})  
 }  
 fields = **'\_\_all\_\_'**

* Update the **'python\_detail.html'** file, so the image displays properly

<**div class="card m-3" style="width**: 18**rem**;**"**>  
 <**img class="card-img-top" src="**{{ **python**.**image**.**url** }}**" alt="Card image cap"**>  
 <**div class="card-body"**>  
 <**h5 class="card-title"**>{{ **python**.**name** }}</**h5**>  
 <**p class="card-text"**>{{ **python**.**description** }}</**p**>  
 </**div**>  
</**div**>

At the end, the **form** should look something like this:



And the **home page** should look like this:

