

文件上传表单FileManagedType类中 表单的事件监听器代码中, 我们手动创建了一个SimpleFile对象, 并为对象属性进行了赋值。这样就类写死了, 就没有灵活性了。

这节课我们额外学习一下Symfony Form表单的一些用法。

在FileManagedType类的buildForm()中, 有第二个参数\$options, 这是创建表单时为表单行设置的选项数据。dd(\$options); 查看CommentType类中为文件表单行设置的选项。

我们可以手动添加一个option, 指定当前文件表单要转换为哪个文件类。覆盖configureOptions()方法。

```
//FileManagedType

public function configureOptions(OptionsResolver $resolver)
{
    // setDefined定义一个option名称, 这样我们可以在创建表单行时为这个名称设置
    值。
    // 为了防止漏用这个option, 添加另外一个设置 setRequired , 这
    样'file_class'选项就是必填项。
    // 使用setDefaults设置选项的默认值
    $resolver
        ->setDefined('file_class')
        ->setRequired('file_class')
    ;
}
```

在 buildForm() 方法中获取file_class, 再创建对应的文件对象。注释数组形式的表单监听器, 我们需要使用闭包方法的方式来完成数据的转换。

```
//FileManagedType

public function buildForm(FormBuilderInterface $builder, array
$options)
{
    $fileClass = $options['file_class'];
    // $builder->addEventListener(FormEvents::SUBMIT, [$this,
    'onFormSubmit']);

    $builder->addEventListener(FormEvents::SUBMIT, function (FormEvent
$event) use ($fileClass) {
        /**@var UploadedFile $data */
        $file = $event->getData();
        $originName = $file->getClientOriginalName();
        $fileName = pathinfo(htmlspecialchars($originName),
        PATHINFO_FILENAME) . '-' . $file->getFilename() . '.' . $file-
        >getClientOriginalExtension();
        // $uploadPath = $this->parameterBag-
        >get('teebb.upload_dir');//$this->getParameter('base_path');

        $mimeType = $file->getMimeType();
    });
}
```

```
        $filesize = $file->getSize();

//        $file->move($uploadPath, $fileName);
        $file->move($this->uploadDir, $fileName);

        $uploadedFile = new $fileClass();
        $uploadedFile->setName($fileName);
        $uploadedFile->setMimeType($mimeType);
        $uploadedFile->setOriginName($originName);
        $uploadedFile->setSize($filesize);

        $event->setData($uploadedFile);
    });
}
```

我们在文件类中保存了常用的4个属性，但是当用户扩展文件类增加其他的属性时，这样又无法满足。我们可以使用另一个Symfony的核心组件 **EventDispatcher**，我们在文件对象赋值后，定义一个事件，把上传的文件 **\$file** 和创建的文件对象 **\$uploadedFile** 都添加到事件对象里。这样，你可以根据需要添加自己的事件监听器或者订阅器自由的扩展 **\$uploadedFile** 的属性了。