**上传文件**

**单个文件：**

示例代码：

func main() {

router := gin.Default()

// 为 multipart forms 设置更小的内存限制 (默认是 32 MB)

// router.MaxMultipartMemory = 8 << 20 // 8 MB

router.POST("/upload", func(c \*gin.Context) {

// single file

file, \_ := c.FormFile("file")

log.Println(file.Filename)

// Upload the file to specific dst.

// c.SaveUploadedFile(file, dst)

c.String(http.StatusOK, fmt.Sprintf("'%s' uploaded!", file.Filename))

})

router.Run(":8088")

}

运行上述代码启动服务器，然后在终端窗口通过 curl 模拟文件上传进行测试：



**多个文件**

示例代码：

func main() {

router := gin.Default()

// 为 multipart forms 设置更小的内存限制 (默认是 32 MB)

// router.MaxMultipartMemory = 8 << 20 // 8 MB

router.POST("/upload", func(c \*gin.Context) {

// Multipart form

form, \_ := c.MultipartForm()

files := form.File["upload[]"]

for \_, file := range files {

log.Println(file.Filename)

// Upload the file to specific dst.

// c.SaveUploadedFile(file, dst)

}

c.String(http.StatusOK, fmt.Sprintf("%d files uploaded!", len(files)))

})

router.Run(":8088")

}

通过 curl 模拟文件上传进行测试：

