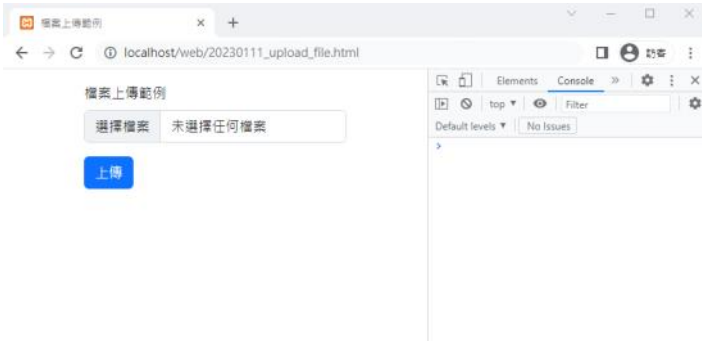


11-2. 檔案上傳

1. 設定上傳畫面

```
<div class="container">
  <div class="row justify-content-center">
    <div class="col-5">
      <div class="my-3">
        <label for="fileupload" class="form-label">檔案上傳範例</label>
        <input class="form-control" type="file" id="fileupload" name="fileupload">
      </div>
      <button class="btn btn-primary" id="upload-button">上傳</button>
    </div>
  </div>
</div>
```



2. 預覽畫面

預覽畫面預設為隱藏

```
<div class="my-3">
  <label for="fileupload" class="form-label">檔案上傳範例</label>
  <input class="form-control" type="file" id="fileupload" name="fileupload">
  
</div>
```

確定上傳格式為jpeg或png後顯示預覽畫面
並利用flag_upload紀錄是否允許上傳的狀態

```
$("#fileupload").change(function () {
  console.log(fileupload.files[0].type);
  if (fileupload.files[0].type == "image/jpeg" || fileupload.files[0].type == "image/png") {
    $("#prevImg").removeClass("d-none");
    $("#prevImg").attr("src", URL.createObjectURL(fileupload.files[0]));
    flag_upload = true;
  } else {
    console.log("jpeg or png");
    $("#prevImg").addClass("d-none");
    flag_upload = false;
  }
});
```



上傳格式不符合規定，不顯示預覽效果!



3. 後端程式

參考<https://www.php.net/manual/en/reserved.variables.files.php>

\$_FILES["file"]["name"]：上傳檔案的原始名稱。

\$_FILES["file"]["type"]：上傳的檔案類型。

\$_FILES["file"]["size"]：上傳的檔案原始大小。

\$_FILES["file"]["tmp_name"]：上傳檔案後的暫存資料夾位置。

\$_FILES["file"]["error"]：如果檔案上傳有錯誤，可以顯示錯誤代碼。

先判斷有檔案傳遞再判斷格式是否為jpeg或png

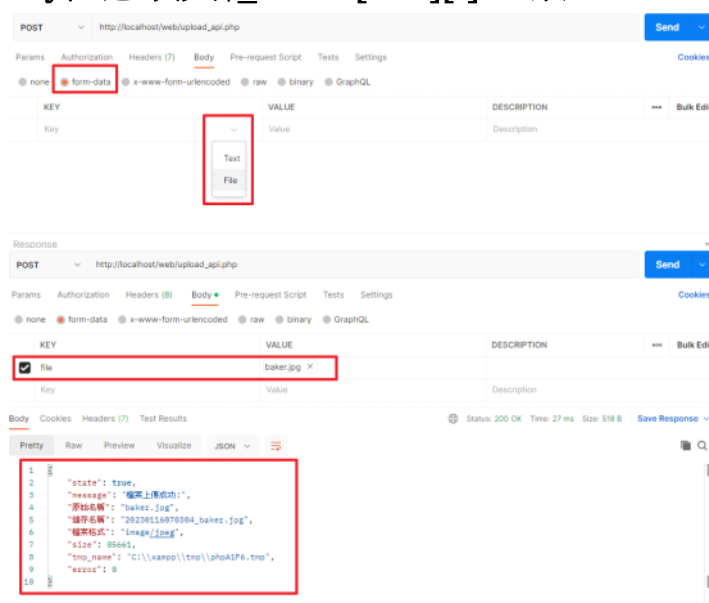
```
<?php
error_reporting(0); // 不顯示錯誤
if(isset($_FILES['file']['name']) && $_FILES['file']['name'] != ""){
    if($_FILES['file']['type'] == 'image/jpeg' || $_FILES['file']['type'] == 'image/png'){
        // 重新命名或使用原本檔名
        // $filename = $_FILES['file']['name']; // 使用原本檔名
        $filename = date("YmdHis") . "_" . $_FILES['file']['name']; // 重新命名

        $location = "upload/" . $filename;

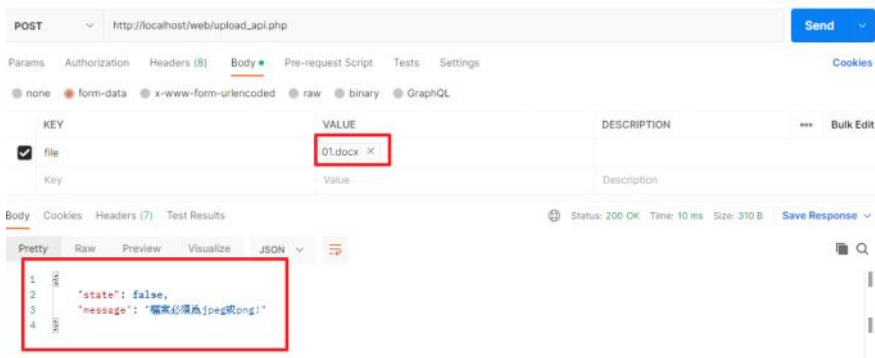
        if (move_uploaded_file($_FILES['file']['tmp_name'], $location)) {
            $datainfo = array();
            $datainfo["state"] = true;
            $datainfo["message"] = "檔案上傳成功!";
            $datainfo["原始名稱"] = $_FILES['file']['name'];
            $datainfo["儲存名稱"] = $filename;
            $datainfo["檔案格式"] = $_FILES['file']['type'];
            $datainfo["size"] = $_FILES['file']['size'];
            $datainfo["tmp_name"] = $_FILES['file']['tmp_name'];
            $datainfo["error"] = $_FILES['file']['error'];
            echo json_encode($datainfo);
        } else {
            $errinfo = array();
            $errinfo["state"] = false;
            $errinfo["message"] = "檔案上傳失敗!";
            echo json_encode($errinfo);
        }
    } else {
        echo '{"state":false, "message":"檔案必須為jpeg或png!"}';
    }
} else {
    echo '{"state":false, "message":"檔案不存在!"}';
}
?>
```

使用postman驗證

key值必須與\$_FILES[file][]一致



測試非規定格式(jpeg或png)



4. ajax傳遞檔案至後端 確認格式是否符合?

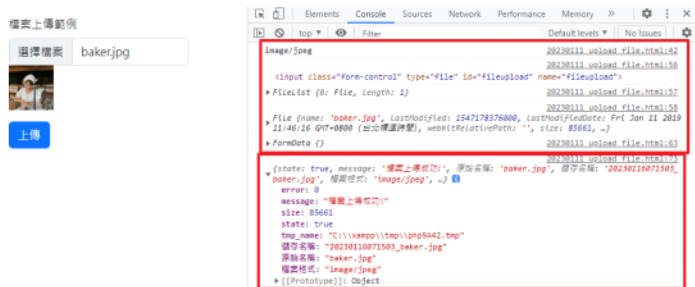
```
var flag_upload = false;
$(function () {
    $("#upload-button").bind("click", function () {
        if (flag_upload) {
            uploadFile();
        } else {
            alert("檔案必須為jpeg or png");
        }
    });
    $("#fileupload").change(function () { ...
});
```

將檔案放置formData 並以ajax傳遞至後端!

```
function uploadFile() {
    console.log(fileupload);
    console.log(fileupload.files);
    console.log(fileupload.files[0]);

    var formData = new FormData();
    //配合後端API命名$_FILES['file']['']
    formData.append("file", fileupload.files[0]);
    console.log(formData);
    $.ajax({
        type: "POST",
        url: "upload_api.php",
        data: formData,
        dataType: "json",
        cache: false,
        contentType: false,
        processData: false,
        success: function (data) {
            console.log(data);
        },
        error: function (data) {
            // alert("error"+data);
            console.log("error" + data);
        }
    });
}
```

圖片符合規格



圖片不符合規格

