

# Python



盛岡情報ビジネス&デザイン専門学校

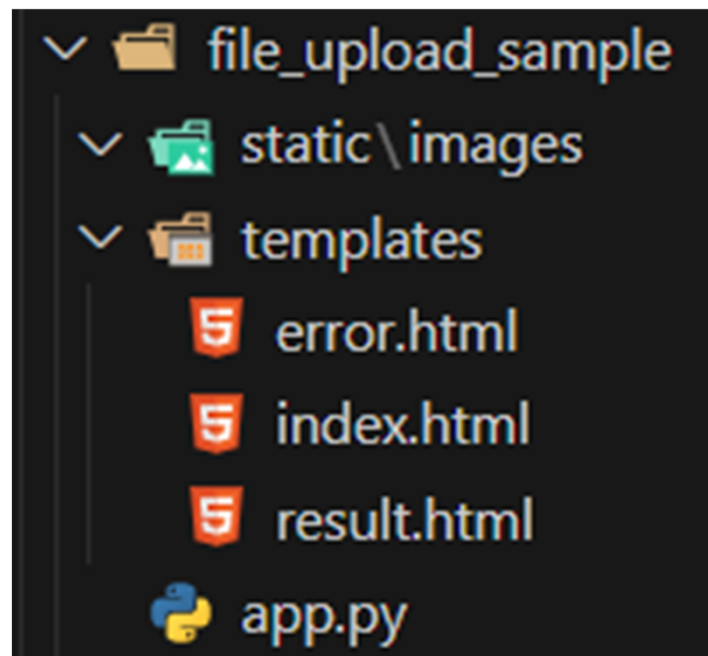
# ★☆☆★本日の内容★☆☆★

## 1. ファイルアップロード



## ◆ファイルアップロード

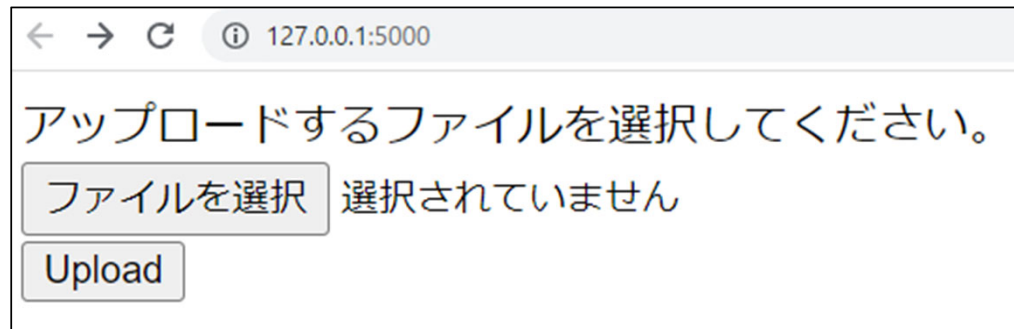
簡単なファイルアップロードアプリを作りながら学習していきましょう。  
まずは以下のように構成でフォルダとファイルを作成します。



## ◆ファイルアップロード

まずは index.html にアップロード画面を作成します。

```
<body>
  アップロードするファイルを選択してください。
  <form action="{{url_for('uploads')}}" method="post" enctype="multipart/form-data">
    <input type="file" name="file"> <br>
    <button type="submit">Upload</button>
  </form>
</body>
```



A screenshot of a web browser window. The address bar shows the URL 127.0.0.1:5000. The page content displays the text "アップロードするファイルを選択してください。" followed by a file selection interface. This interface includes a button labeled "ファイルを選択" (Select File) and the text "選択されていません" (Not selected). Below this is an "Upload" button.



## ◆ファイルアップロード

残りの HTML も作っていきます。

### result.html

```
<body>  
  登録成功  
    
</body>
```

### error.html

```
<body>  
  処理に失敗しました。やり直してください。 <br>  
  <a href="{{url_for('top_page')}}">戻る</a>  
</body>
```



## ◆ファイルアップロード

最後に app.py を作っていきます。

まずは、必要な Flask アプリとしての骨組みと import を書いていきます。

定数 UPLOAD\_FOLDER は static/images の絶対パスを指定します。

```
from flask import Flask, render_template, request, redirect, url_for
from werkzeug.utils import secure_filename
import os

app = Flask(__name__)

# ファイルアップロード先の絶対パス
UPLOAD_FOLDER = 'C://sample-python/flask-sample/file_upload_sample/static/images'

# ここにルーティングの処理を記述(以降のスライド)

if __name__ == '__main__':
    app.run(debug=True)
```



## ◆ファイルアップロード

先に index.html と error.html へのルーティングを書きます。

```
@app.route('/')  
def top_page():  
    return render_template('index.html')  
  
@app.route('/error')  
def error():  
    return render_template('error.html')
```



## ◆ファイルアップロード

最後にアップロードの処理を書きます。

```
@app.route('/uploads', methods=['POST'])
def uploads():
    # 何らかの理由で request にファイルが存在しない場合のエラー処理
    if 'file' not in request.files:
        return redirect(url_for('error'))

    # アップロードされたファイルをrequestから取得
    file = request.files['file']

    # ファイル名が空(何もアップロードされていない)場合のエラー処理
    if file.filename == '':
        return redirect(url_for('error'))

    # ../../a.jpgみたいな名前のファイルへの対策(ディレクトリトラバーサル攻撃)
    # 不要な文字列を削除します。(なお、日本語も削除されちゃいます....)
    name = secure_filename(file.filename)

    # ファイルを保存
    file.save(os.path.join(UPLOAD_FOLDER, name))

    return render_template('result.html', name='images/' + name)
```