## [10주차 - 강의 예제]

● 수업 깃허브 경로

https://github.com/topmentor/RPA2024

- 실습파일 제출 요령
- 1) 주차별 폴더(10week)를 만들고 폴더를 VSCode로 열어서 실습
- 2) 학번 디렉토리 이동 --> github 푸시

```
cd C:\200000
git add .
git commit -m "10주차 실습파일"
git pull
git push -u origin main
```

```
실습 1 - 회귀분석 실습
```

종속변수 weight와 독립변수 중 food에 대한 회귀분석을 실시하시오 'ch5-1.csv' 데이터 위치 ---> <a href="https://github.com/topmentor/RPA2024/tree/main/예제파일">https://github.com/topmentor/RPA2024/tree/main/예제파일</a> --> ex1.py

라이브러리 설치 pip install scipy statsmodels

import pandas as pd

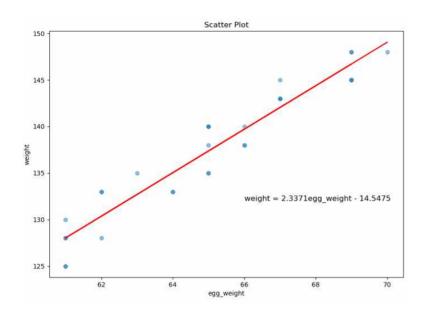
```
import matplotlib.pyplot as plt
import statsmodels.formula.api as smf

w = pd.read_csv('ch5-1.csv')
w_n = w.iloc[:,1:5]

model_lm = smf.ols(formula = 'weight ~ egg_weight', data = w_n)
result_lm = model_lm.fit()
result_lm.summary()
```

## 실습 2 - 회귀분석 실습 시각화 --> ex1.py에 이어서

```
plt.figure(figsize = (10,7))
plt.scatter(w.egg_weight, w.weight, alpha = .5)
plt.plot(w.egg_weight, w.egg_weight*2.3371 - 14.5475, color = 'red')
plt.text(66, 132, 'weight = 2.3371egg_weight - 14.5475', fontsize = 12)
plt.title('Scatter Plot')
plt.xlabel('egg_weight')
plt.ylabel('weight')
plt.show()
```



실습 3 - 실습 1에서 독립변수 중 food에 대한 회귀분석을 실시하시오 (종속변수 weight)

--> ex2.py

```
1. 클라이언트 코딩
: static 폴더 안에 filedownload.html 파일 생성
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>File Download</title>
</head>
<body>
   <h2>Download File</h2>
   <input type="text" id="filename" placeholder="Enter filename">
   <button onclick="downloadFile()">Download File</button>
<script>
    function downloadFile() {
        const filename = document.getElementById('filename').value;
        if (filename.length > 0) {
           const url = \files/${filename}\;
           const a = document.createElement('a');
           a.href = url;
           a.download = filename;
           document.body.appendChild(a);
           a.click();
           document.body.removeChild(a);
       } else {
           alert('Please enter a filename');
       }
    }
</script>
</body>
</html>
```

- 2. 서버 코딩: Form 요청에 대응하는 API
- 9주차 fapil.py 코드를 복사해서 추가 편집
- 'from fastapi.staticfiles import StaticFiles' 윗 줄에 코딩

from fastapi import File, UploadFile
import shutil
from pathlib import Path
from fastapi.responses import FileResponse

@app.get("/files/{filename}")
async def get\_file(filename: str):
 file\_path = Path("static/uploads") / filename
 if file\_path.is\_file():
 return FileResponse(path=file\_path, filename=filename)
 else:
 raise HTTPException(status\_code=404, detail="File not found")