

23차시	총10문제		연습: <input type="checkbox"/>	과제 : <input checked="" type="checkbox"/>	평가 : <input type="checkbox"/>
1. JSON 파일을 Python 객체로 변환하는 함수는 무엇인가요?					
a) json.load()					
b) json.loads()					
c) json.read_json()					
d) json.parse_json()					
2. matplotlib에서 X축과 Y축의 라벨을 설정하는 함수는 무엇인가요?					
a) plt.xlabel(), plt.ylabel()					
b) plt.x_label(), plt.y_label()					
c) plt.set_xlabel(), plt.set_ylabel()					
d) plt.axis_label_x(), plt.axis_label_y()					
3. Python의 re 모듈에서 특정 패턴과 일치하는 문자열을 찾는 함수는 무엇인가요?					
a) re.match()					
b) re.search()					
c) re.findall()					
d) 위의 모든 옵션					
4. PdfPages 객체에 플롯을 저장하는 함수는 무엇인가요?					
a) savefig()					
b) add_page()					
c) write_pdf()					
d) insert_fig()					
5. Python에서 REST API 요청을 보내는 데 가장 많이 사용되는 라이브러리는 무엇인가요?					

a) requests

b) urllib

c) http.client

d) fetch

6. Python의 json 모듈을 사용하여 다음 데이터를 JSON 파일(data.json)로 저장하세요.

```
{  
    "name": "홍길동",  
    "age": 25,  
    "city": "서울"  
}
```

The screenshot shows a terminal window with the following content:

```
python > #!/usr/bin/python  
      1 import json  
      2  
      3  
      4 python_dict = {  
      5     "name": "홍길동",  
      6     "age": 25,  
      7     "city": "서울"  
      8   }  
      9  
     10  
     11   with open('data.json', 'w', encoding='utf-8') as file:  
     12     json.dump(python_dict, file, indent=4, ensure_ascii=False)  
     13  
     14 print("data.json 파일이 깔끔하게 저장되었습니다.")
```

The terminal window has a dark background and light-colored text. The code is written in Python, using the json module to dump the dictionary into a file named 'data.json'. The terminal also shows the current directory as 'C:\Users\hone\AppData\Local\Programs\Python\Python33\python.exe' and the command 'data.json'.

## 7. requests 모듈을 사용하여 JSONPlaceholder API

(<https://jsonplaceholder.typicode.com/todos/1>)에서 데이터를 가져오세요.

```
import requests

api_url = "https://jsonplaceholder.typicode.com/todos/1"

try:
    response = requests.get(api_url)

    response.raise_for_status()

    data = response.json()

    print("API에서 가져온 데이터:")
    print(data)

except requests.exceptions.RequestException as e:
    print(f"오류가 발생: {e}")
```

PS C:\rokey> & C:/Users/hoone/AppData/Local/Programs/Python/Python313/python.exe c:/rokey/python/dd.py  
API에서 가져온 데이터:  
{'userId': 1, 'id': 1, 'title': 'delectus autem', 'completed': False}  
PS C:\rokey>

## 8. <https://randomuser.me/api/>에서

랜덤 유저 정보를 가져오고, name 값을 출력하세요.

```
import requests

api_url = "https://randomuser.me/api/"

try:
    response = requests.get(api_url)
    response.raise_for_status()

    data = response.json()

    user_name = data['results'][0]['name']

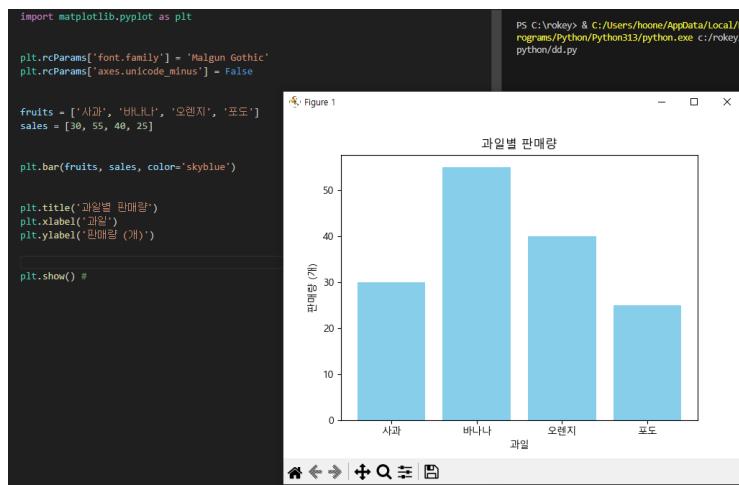
    print("랜덤 유저의 이름 정보:")
    print(user_name)

    print(f"\n- 호칭: {user_name['title']}")
    print(f"- 이름: {user_name['first']}")
    print(f"- 성: {user_name['last']}")

except requests.exceptions.RequestException as e:
    print(f"API 요청 중 오류가 발생: {e}")
```

PS C:\rokey> & C:/Users/hoone/Programs/Python/Python313/python python/dd.py  
랜덤 유저의 이름 정보:  
{'title': 'Mr', 'first': 'Blake', 'last': 'Wang'}  
- 호칭: Mr  
- 이름: Blake  
- 성: Wang  
PS C:\rokey>

9. matplotlib을 사용하여 과일별 판매량을 나타내는 막대 그래프를 그려세요.



10. logfile.log에서 2025-03-30 날짜의 로그만 추출하세요.

# 예제 로그 파일 생성 코드

```
path = r"homework\logfile.log"
with open(path, "w", encoding="utf-8") as file:
    file.write("2025-03-30 12:00:01 INFO 서버 시작됨\n")
    file.write("2025-03-30 12:05:12 ERROR 데이터베이스 연결 실패\n")
    file.write("2025-03-30 12:10:35 WARNING 응답 속도 저하\n")
    file.write("2025-03-30 12:15:45 ERROR 사용자 인증 실패\n")
```

The figure shows a Jupyter Notebook cell with the following Python code:

```
import os

dir_path = "homework"
os.makedirs(dir_path, exist_ok=True)

file_path = os.path.join(dir_path, "logfile.log")

with open(file_path, "w", encoding="utf-8") as file:
    file.write("2025-03-30 12:00:01 INFO 서버 시작됨\n")
    file.write("2025-03-30 09:30:00 INFO 사용자 로그인()\n") # 다른 날짜 로그 추가
    file.write("2025-03-30 12:05:12 ERROR 데이터베이스 연결 실패\n")
    file.write("2025-03-30 12:10:35 WARNING 응답 속도 저하\n")
    file.write("2025-03-30 10:00:00 INFO 서버 접속 완료()\n") # 다른 날짜 로그 추가
    file.write("2025-03-30 12:15:45 ERROR 사용자 인증 실패\n")

print(f"{file_path} 예제 파일 생성이 완료되었습니다.\n")

target_date = "2025-03-30"
print(f"{file_path}에서 '{target_date}' 날짜의 로그:")

try:
    with open(file_path, "r", encoding="utf-8") as file:
        for line in file:
            if line.startswith(target_date):
                print(line.strip())
except FileNotFoundError:
    print(f"오류: '{file_path}' 파일을 찾을 수 없습니다.")
```

The terminal output shows the log file was created and contains the specified entries for March 30, 2025. The code then filters these entries.

