```
 Weather Forecast Data
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2 entries, 0 to 1
Data columns (total 5 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   date           2 non-null      object
 1   temp_min       2 non-null      float64
 2   temp_max       2 non-null      float64
 3   condition      2 non-null      object
 4   precipitation  2 non-null      float64
dtypes: float64(3), object(2)
memory usage: 212.0+ bytes
None
         date  temp_min  temp_max condition  precipitation
0  2024-04-23      15.2      21.4        흐림            0.0
1  2024-04-24      12.3      20.1        맑음            0.2

 News Data
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2 entries, 0 to 1
Data columns (total 4 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   title        2 non-null      object
 1   link         2 non-null      object
 2   description  2 non-null      object
 3   pubDate      2 non-null      object
dtypes: object(4)
memory usage: 196.0+ bytes
None
                                               title  ₩
0            Global food prices rise to record high
1  Biden warns Russia of 'severe consequences' if...

                                                link  ₩
0        https://www.bbc.com/news/business-61183070
1  https://www.bbc.com/news/world-us-canada-61182951

                                         description  ₩
0  The Food and Agriculture Organization of the U...
1  US President Joe Biden has warned Russia of "s...

                         pubDate
0  Mon, 22 Apr 2024 14:32:00 GMT
1  Mon, 22 Apr 2024 13:58:00 GMT
```

```
 News Data
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2 entries, 0 to 1
Data columns (total 4 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   title        2 non-null      object
 1   link         2 non-null      object
 2   description  2 non-null      object
 3   pubDate      2 non-null      object
dtypes: object(4)
memory usage: 196.0+ bytes
None
                                                title  ₩
0             Global food prices rise to record high
1  Biden warns Russia of 'severe consequences' if...

                                                 link  ₩
0         https://www.bbc.com/news/business-61183070
1  https://www.bbc.com/news/world-us-canada-61182951

                                          description  ₩
0  The Food and Agriculture Organization of the U...
1  US President Joe Biden has warned Russia of "s...

                          pubDate
0  Mon, 22 Apr 2024 14:32:00 GMT
1  Mon, 22 Apr 2024 13:58:00 GMT
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
Book Data
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3 entries, 0 to 2
Data columns (total 4 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   title   3 non-null      object
 1   author  3 non-null      object
 2   price   3 non-null      float64
 3   genre   3 non-null      object
dtypes: float64(1), object(3)
memory usage: 228.0+ bytes
None
                title         author  price    genre
0    Pride and Prejudice     Jane Austen  19.99  romance
1  To Kill a Mockingbird      Harper Lee  14.99  fiction
2  The Lord of the Rings  J.R.R. Tolkien  34.99  fantasy
```

```
## 3. Weather XML 처리
tree_weather = ET.parse(file_weather)
root_weather = tree_weather.getroot()
forecast_days = root_weather.find('forecast').findall('day')

weather_data = []
for day in forecast_days:
  temp = day.find('temperature')
  row = {
      'date' : day.attrib['date'],
      'temp_min' : float(temp.attrib['min']),
      'temp_max' : float(temp.attrib['max']),
      'condition' : day.find('condition').text,
      'precipitation' : float(day.find('precipitation').text)
  }
  weather_data.append(row)


df_weather =pd.DataFrame(weather_data)
print("₩n Weather Forecast Data")
print(df_weather.info())
print(df_weather.head())
```

```
## 2. News XML 처리
tree_news = ET.parse(file_news)
root_news = tree_news.getroot()
news_items = root_news.find('channel').findall('item')

news_data = []
for item in news_items :
  row = {
      'title' : item.find('title').text,
      'link' : item.find('link').text,
      'description' : item.find('description').text,
      'pubDate' : item.find('pubDate').text
  }
  news_data.append(row)

df_news = pd.DataFrame(news_data)
print("₩n News Data")
print(df_news.info())
print(df_news.head())
```

```
## 1. Book XML 처리
tree_book = ET.parse(file_book)
root_book = tree_book.getroot()
book_elements = root_book.findall('book')

book_data = []
for element in book_elements:
  row = {
      'title': element.find('title').text,
      'author': element.find('author').text,
      'price' : float(element.find('price').text),
      'genre' : element.find('genre').text
  }
  book_data.append(row)

df_book = pd.DataFrame(book_data)
print("Book Data")
print(df_book.info())
print(df_book.head())
```

```
import pandas as pd
import xml.etree.ElementTree as ET

from google.colab import drive
drive.mount('/content/drive')

file_book = "/content/drive/MyDrive/xmlData/book.xml"
file_news = "/content/drive/MyDrive/xmlData/news.xml"
file_weather ="/content/drive/MyDrive/xmlData/weather.xml"
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3 entries, 0 to 2
Data columns (total 5 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   symbol  3 non-null      object
 1   name    3 non-null      object
 2   price   3 non-null      float64
 3   change  3 non-null      float64
 4   volume  3 non-null      int64
dtypes: float64(2), int64(1), object(2)
memory usage: 252.0+ bytes
None
  symbol           name    price  change    volume
0   AAPL     Apple Inc.   152.34    1.23  12345678
1   GOOG  Alphabet Inc.  2234.56   -0.78  34567890
2   TSLA     Tesla Inc.   789.10    3.21  98765432
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3 entries, 0 to 2
Data columns (total 5 columns):
 #   Column    Non-Null Count  Dtype 
---  ------    --------------  ----- 
 0   id        3 non-null      int64 
 1   name      3 non-null      object
 2   price     3 non-null      int64 
 3   brand     3 non-null      object
 4   category  3 non-null      object
dtypes: int64(2), object(3)
memory usage: 252.0+ bytes
None
    id                name    price    brand category
0  101  iPhone 13 Pro Max  1299000    Apple     스마트폰
1  102   Galaxy S22 Ultra  1199000  Samsung     스마트폰
2  103         LG Gram 17  1999000       LG      노트북

RangeIndex: 2 entries, 0 to 1
Data columns (total 5 columns):
 #   Column   Non-Null Count  Dtype 
---  ------   --------------  ----- 
 0   id       2 non-null      int64 
 1   name     2 non-null      object
 2   email    2 non-null      object
 3   phone    2 non-null      object
 4   address  2 non-null      object
dtypes: int64(1), object(4)
memory usage: 212.0+ bytes
None
   id name             email             phone  ₩
0   1  홍길동  hong@example.com  +82-10-1234-5678
1   2  김철수   kim@example.com  +82-10-8765-4321

                              address
0  {'city': '서울', 'country': '대한민국'}
1  {'city': '부산', 'country': '대한민국'}
```

```
import pandas as pd
import json

from google.colab import drive
drive.mount('/content/drive')

# JSON 데이터 3가지 모두 파일 위치를 file1,2,3으로 정의하였습니다.
file1 = '/content/drive/MyDrive/jsonData/custom.json'
file2 = '/content/drive/MyDrive/jsonData/product.json'
file3 = '/content/drive/MyDrive/jsonData/stock.json'

# JSON 파일 3개를 모두 불러와주고 데이터 프레임으로 만들어주 었습니다.
with open(file1, 'r', encoding='utf-8') as f1:
  data1 = json.load(f1)
df1 = pd.DataFrame(data1)

with open(file2, 'r', encoding='utf-8') as f2:
  data2 = json.load(f2)
df2 = pd.DataFrame(data2)

with open(file3, 'r', encoding='utf-8') as f3:
  data3 = json.load(f3)
df3 = pd.DataFrame(data3)

# Dataframe 정보 출력
print(df1.info())
print(df1.head())

print(df2.info())
print(df2.head())

print(df3.info())
print(df3.head())
```

```python
import json
import pandas as pd

from google.colab import drive
drive.mount('/content/drive')

file_path = '/content/drive/MyDrive/plidd-org-sakura/tokenizer.json'

# JSON 파일 열기
with open(file_path, "r", encoding="utf-8") as f:
    data = json.load(f)

# 올바른 키 사용
added_tokens = data["added_tokens"]

# DataFrame 변환
df_json = pd.DataFrame(added_tokens)
print(df_json.head())
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
      id content  single_word  lstrip  rstrip  normalized  special
0      0   [PAD]        False   False   False       False     True
1      1   [CLS]        False   False   False       False     True
2      2   [SEP]        False   False   False       False     True
3      3   [UNK]        False   False   False        True     True
4 128000  [MASK]        False   False   False       False     True
```