

# 3주차-1

## 1. 오늘은 무엇을 배우는가?

### 2. Seaborn

#### 2.1. 패키지 불러오기

#### 2.2. Seaborn 테마 설정하기

##### 2.2.1. 테마 설정 - 스타일

##### 2.2.2. 테마 설정 - 팔레트

#### 2.3. 데이터 불러오기

#### 2.4. 그래프 - relplot

##### 2.4.1. scatterplot

##### 2.4.2. lineplot

#### 2.5. relplot으로 여러 변수 비교하기

## 1. 오늘은 무엇을 배우는가?

1. 패키지 불러오기
2. Seaborn 테마 설정하기
3. 데이터 불러오기
4. 그래프 - relplot
  - a. scatterplot
  - b. lineplot
5. relplot으로 여러 변수 비교하기

## 2. Seaborn

- Seaborn
  - 파이썬을 통해 데이터 시각화를 도와주는 도구입니다.

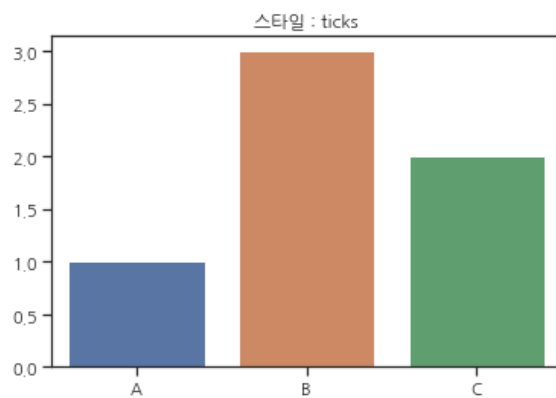
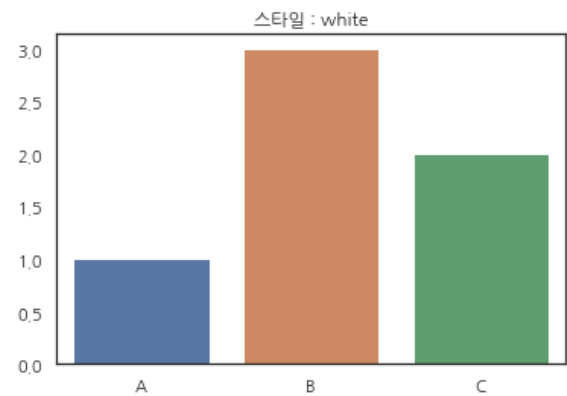
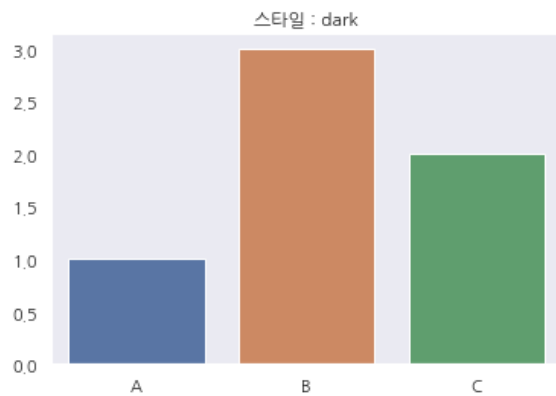
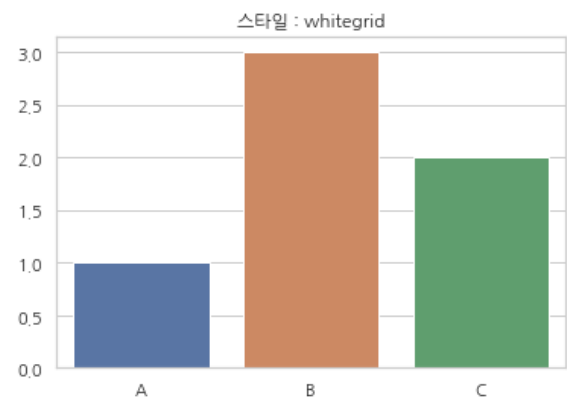
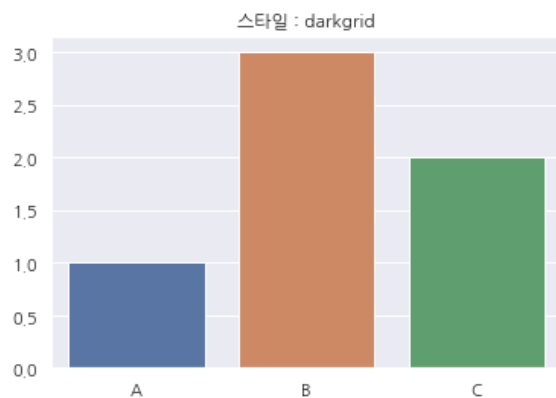
### 2.1. 패키지 불러오기

```
import pandas as pd
import seaborn as sns
```

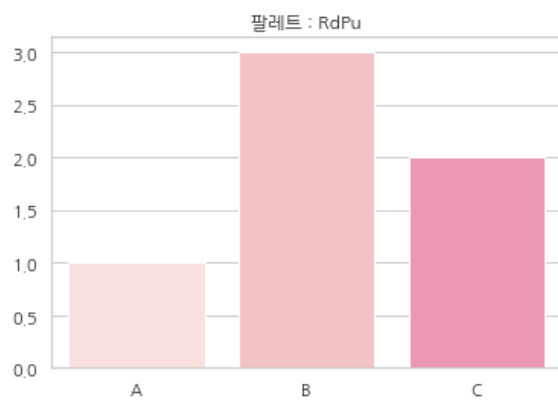
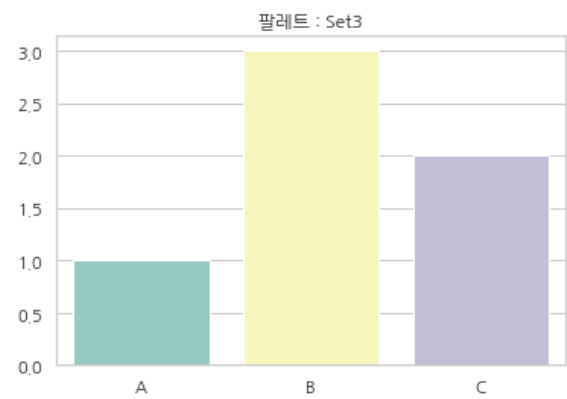
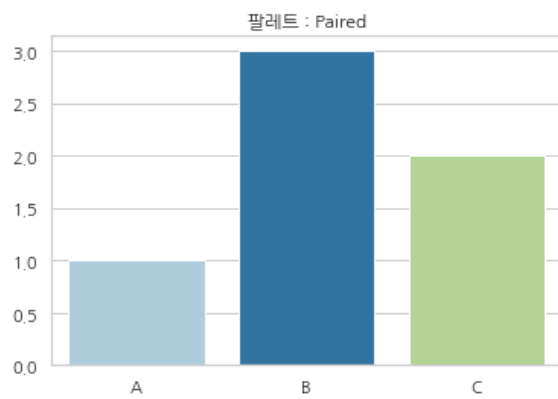
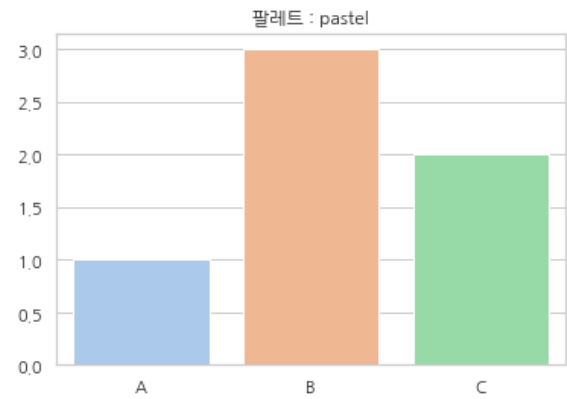
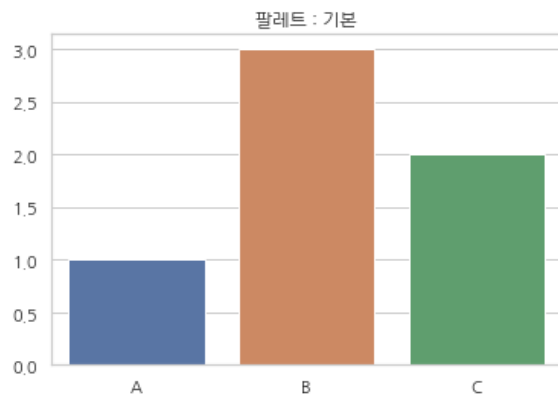
## 2.2. Seaborn 테마 설정하기

### 2.2.1. 테마 설정 - 스타일

```
sns.set_theme(style='whitegrid', # ["darkgrid", "whitegrid", "dark", "white", "ticks"]  
             palette="pastel", # 기본값, [pastel, Paired, Set3, RdPu]  
             font="NanumGothic", # 폰트 설정  
             rc={"axes.unicode_minus":False}, # 마이너스 기호가 출력 시 깨지는 문제 방지  
             )
```



## 2.2.2. 테마 설정 - 팔레트



## 2.3. 데이터 불러오기

- seaborn에서 테스트용 데이터로 제공하는 연비 데이터를 사용합니다.

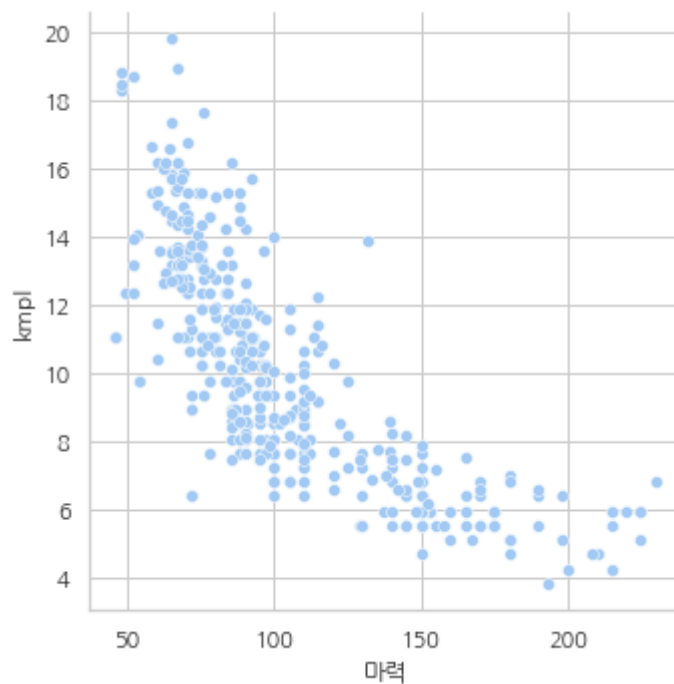
```
연비 = pd.read_excel("연비.xlsx")
```

## 2.4. 그래프 - relplot

### 2.4.1. scatterplot

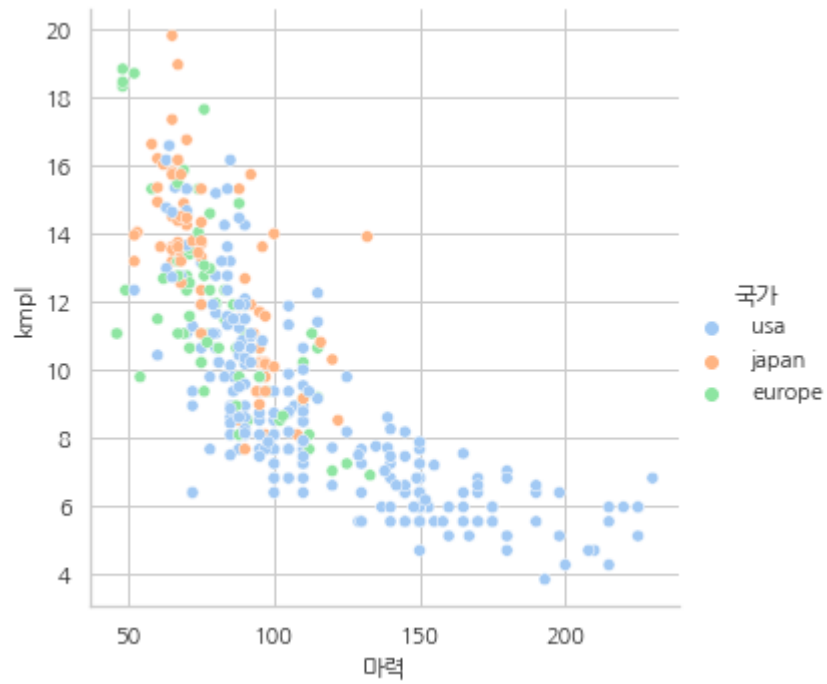
- 기본적으로 X축, Y축, 데이터의 3개 파라미터가 필요합니다.

```
X축 = "마력"  
Y축 = "kmpl"  
데이터 = 연비  
  
sns.relplot(x=X축, y=Y축, data=데이터)
```



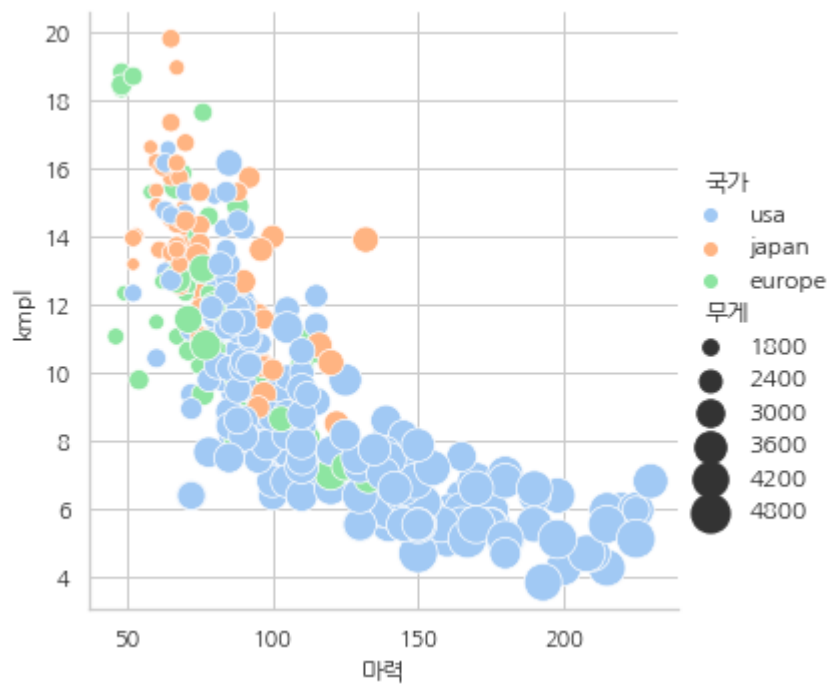
- hue 파라미터를 추가하여 데이터를 색으로 구분할 수 있습니다.

```
색구분기준 = "국가"  
sns.relplot(x=X축, y=Y축, data=데이터, hue=색구분기준) # 카테고리별 색깔 지정
```



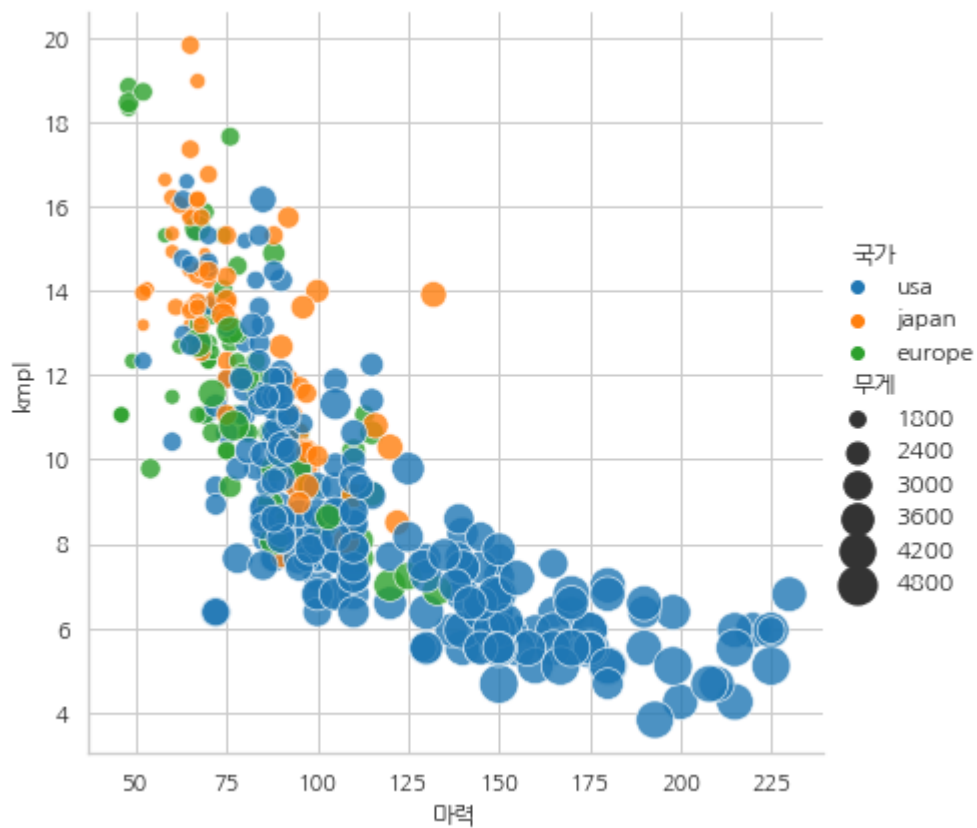
- size, sizes 파라미터를 추가하여 데이터의 크기를 시각적으로 표현할 수 있습니다.
  - sizes에 들어가는 숫자는 점의 최소 크기, 최대 크기를 의미합니다.  
 sizes=(40, 400)일 경우 주어진 데이터의 최소값은 점의 크기가 40, 최대값은 점의 크기가 400이 됩니다.  
 이해가 잘 가지 않는다면 sizes=(1, 900)처럼 아주 작은 값과 큰 값으로 입력해보면 점의 크기가 눈에 띄게 차이나는 걸 확인할 수 있습니다.

```
sns.relplot(x=X축, y=Y축, data=데이터, hue=색구분기준, size="무게", sizes=(40, 400)) #
점의 사이즈 지정
```



- alpha로 데이터의 투명도, height로 그래프의 크기, palette로 데이터의 색을 지정할 수 있습니다.

```
sns.relplot(x=X축, y=Y축, data=데이터, hue=색구분기준, size="무게", sizes=(40, 400), alpha=.5, height=6, palette = "tab10")
```

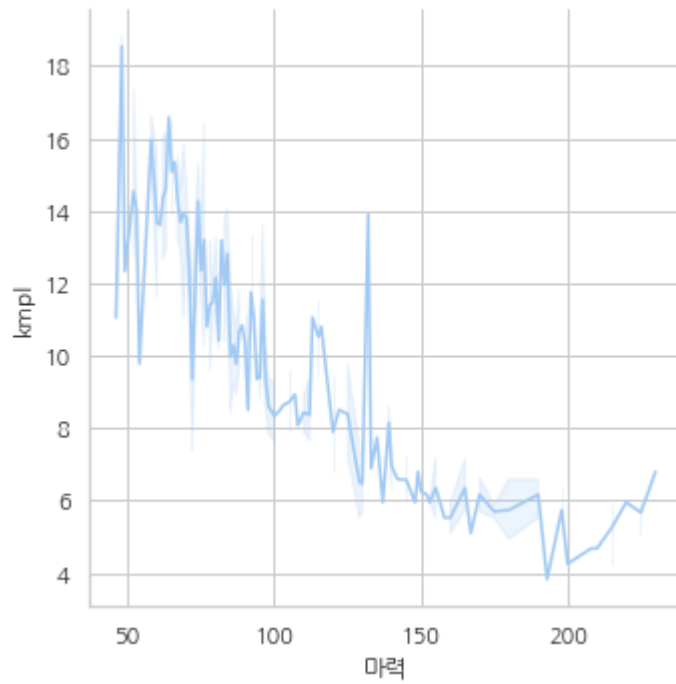


## 2.4.2. lineplot

- lineplot은 relplot 함수에서 kind="line" 파라미터를 추가하여 표현할 수 있습니다. 기본적으로 선을 표현하되, 데이터의 95%만큼을 표현하기 위해 신뢰구간을 추가로 그려줍니다.

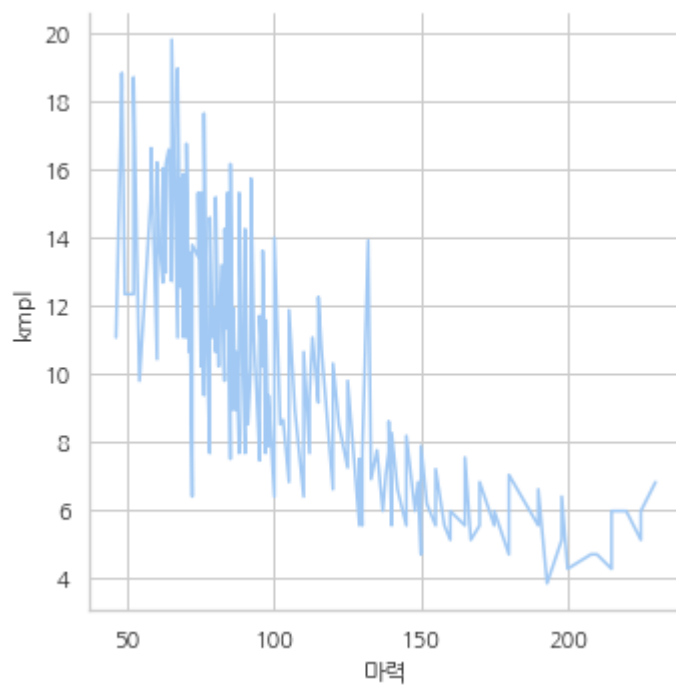
```
X축 = "마력"
Y축 = "kmpl"
데이터 = 연비
```

```
sns.relplot(x=X축, y=Y축, data=데이터, kind="line") #
```



- 신뢰구간을 따로 표현하고 싶지 않은 경우 `estimator=None`을 입력합니다.

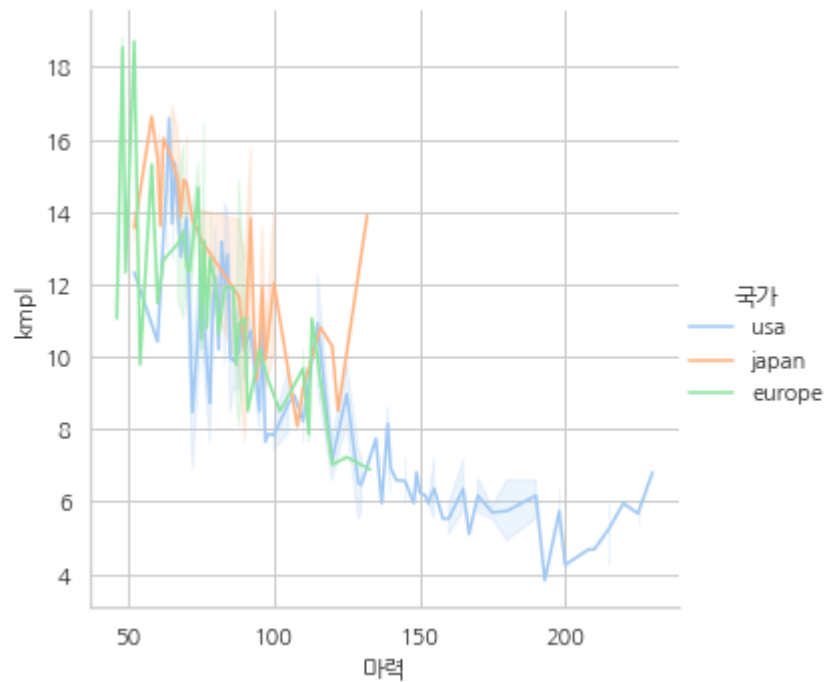
```
sns.relplot(x=X축, y=Y축, data=데이터, kind="line", estimator=None)
```



- `scatterplot`에서 사용했던 파라미터를 마찬가지로 입력할 수 있습니다.



```
sns.relplot(x=X축, y=Y축, data=데이터, kind="line", hue=색구분기준)
```

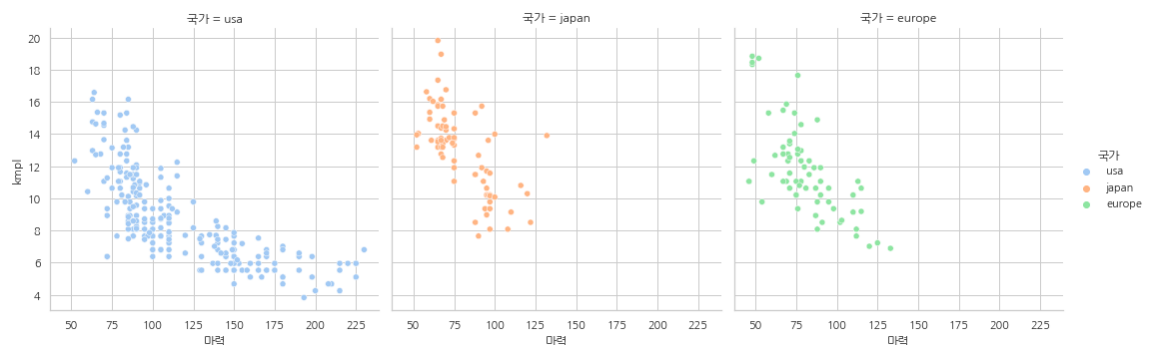


## 2.5. relplot으로 여러 변수 비교하기

- col 파라미터를 통해 여러 그래프를 동시에 볼 수 있습니다.

```
X축 = "마력"
Y축 = "kmp1"
데이터 = 연비
```

```
sns.relplot(x=X축, y=Y축, data=데이터, hue="국가", col="국가")
```



- row, col 파라미터를 통해 여러 그래프를 가로, 세로로 확장해서 여러 변수에 대한 데이터 분포를 확인할 수 있습니다.

```
연비["배기량_100단위"] = 연비["배기량"] // 100
연비["배기량_100단위"] = 연비["배기량_100단위"].astype(int)

sns.relplot(x=X축, y=Y축, data=데이터, hue="국가", col="국가", row="배기량_100단위")
```

