

Aby połączyć dwa ramki danych korzysta się z funkcji **merge()**

np.

```
merge(x=objekt_1, y=objekt_2, by='wspolny_element')
```

```
merge(x=objekt_1, y=objekt_2, by.x='element.x', by.y='element.y')
```

```
merge(x=objekt_1, y=objekt_2, by.x='element.x', by.y='element.y', all=TRUE)
```

1. Połącz podane ramki danych na różne sposoby:

- wyświetlając wszystkie wiersze, które występują tylko w jednym lub tylko w drugim data frame,
- wyświetlając wiersze istniejące w make_df bez pasującego w brand_df oraz istniejące w brand_df bez pasującego w make_df,
- dodając suffix do kolumn is_Europe: _make lub _brand.

```
make_df <- data.frame(name=c('Toyota','Ford','Fiat','Nissan'),
                      country=c('Japan','USA','Italy','Japan'),
                      president=c('Akio Toyoda','William Clay Ford Junior','John Elkann',NA),
                      is_Europe=c(F,F,T,F))
```

```
brand_df <- data.frame(
  name=c('Corolla','Auris','Mondeo','Sedici','Cinquecento','Tipo','Astra'),
  make=c('Toyota','Toyota','Ford','Fiat','Fiat','Fiat','Opel'),
  available=c(F,T,F,F,F,T,T),
  mileage=c(56,13,400,30,84,120,10),
  is_Europe = c(T,T,F,T,T,T,T))
```

2. Wynik z ostatniego zadania w poprzednim poleceniu zapisz w zmiennej result, a następnie wyświetl tylko nazwę make, nazwę brand i mileage.

Do ramki danych możemy dodawać dodatkowe wiersze i kolumny za pomocą odpowiednich funkcji **cbind()** i **rbind()**

3. Do podanej ramki danych dodaj nową kolumnę, która będzie druga ramka danych zawierająca wektor location=c('Toyota','Dearborn','Torino','Jokohama').

```
make1 <- data.frame(name=c('Toyota','Ford','Fiat','Nissan'),
                    country=c('Japan','USA','Italy','Japan'),
                    president=c('Akio Toyoda','William Clay Ford Junior','John Elkann',NA),
                    is_Europe=c(F,F,T,F))
```

4. Utwórz kolejną ramkę danych.

```
make2 <- data.frame(name=c('Mercedes','Renault','Nissan'),
                    country=c('Germany','France','Japan'),
                    president=c(NA,NA,NA),
                    is_Europe=c(T,T,F))
```

Ćwiczenia zrealizowane w RStudio zapisz w oddzielnym pliku (File -> Save as -> Save) o nazwie Lab_5 i prześlij go za pomocą modułu „Zadanie” w MS Teams.

Pomocne w filtrowaniu danych w ramce danych mogą być funkcje: **union()**, **intersect()**, **setdiff()**, **setequal()**

5. Pracując na kolumnie name z make1 i make2:
 - wyświetl nazwy marek występujące w obu kolumnach,
 - wyświetl nazwy marek występujące i w jednej i w drugiej kolumnie,
 - wyświetl nazwy marek, które występują w make1, ale nie występują w make2,
 - wyświetl informację o tym, czy dane w obu kolumnach są takie same.
6. Wczytaj plik fruits.csv do obiektu owoce.
7. Uporządkuj dane z kolumny „protein..g.” (średni wynik) według zawartości wody (kolumna „water..g.”). Wykorzystaj funkcję aggregate()
8. Wyświetl owoce, które mają zawartość wody większą bądź równą 90 g. Użyj funkcji subset()
9. Uporządkuj dane z obiektu owoce według zawartości witaminy C. Wykorzystaj funkcję split()
10. Wyświetl liczebność owoców według danych z kolumny „carbohydrates..g.”. Wykorzystaj funkcję table()

DLA CHĘTNYCH

11. Wczytaj plik wheather_madrid.csv do obiektu pogoda_Madryt.
12. Dokonaj filtrowania wybranych danych, używając następujących funkcji:
 - aggregate()
 - table()
 - subset()
 - split()

Ćwiczenia zrealizowane w RStudio zapisz w oddzielnym pliku (File -> Save as -> Save) o nazwie Lab_5 i prześlij go za pomocą modułu „Zadanie” w MS Teams.