

Oppgaver 3

October 6, 2025

```
[ ]: renv::autoload()
```

1 Oppgaver 3

1.1 Hent inn pakken `tidyverse` som skal brukes i oppgavene med funksjonen `library()`

```
[ ]:
```

1.2 Last inn filen “`befolkning_per_kommune.parquet`” som ligger i mappen `data` kursmateriellet. Kall objektet for `befolkning_per_kommune`.

```
[ ]:
```

1.3 Print de 6 første radene i `befolkning_per_kommune`

```
[ ]:
```

1.4 Hvilke variabeltyper har kolonnene i filen?

```
[ ]:
```

1.5 Hva er gjennomsnittet og medianen til kolonnen `value`?

```
[ ]:
```

1.6 Bruk funksjonen `rename()` til å endre navnet på kolonnene `Region` til `komm_nr` og `value` til `personer`.

```
[ ]:
```

1.7 Bruk funksjonen `select()` til å fjerne variabelen `Tid` og endre rekkefølgen på kolonnene slik: `komm_nr`, `kommunenavn`, `personer`.

[]:

1.8 Hvor mange personer bor det i Bergen kommune?

[]:

1.9 Bruk funksjonen `filter()` for å beholde kommuner med befolkning på lavere enn eller lik 1000 personer. Hvor mange kommuner blir det?

[]:

1.10 Hvilke to kommuner har flest og færrest antall innbyggere? Hvor mange innbyggere har disse kommunene?

[]:

1.11 Bruk funksjonen `mutate()` til å opprette en ny variabel som heter `fylke_nr` som inneholder fylkesnummeret til hver kommune. Hint: fylkesnummeret er de to første sifrene i kommunenummeret.

[]:

1.12 Bruk funksjonene `group_by()` og `summarise()` for å beregne summen og gjennomsnittet av befolkning per fylke. Lagre resultatene i et nye objekter som heter `befolkning_per_fylke_sum` og `befolkning_per_fylke_gjennomsnitt` og kall de nye variablene for `sum` og `gjennomsnitt`

[]:

1.13 Koble sammen datasettene `befolkning_per_fylke_sum` og `befolkning_per_fylke_gjennomsnitt` og kall det nye objektet for `befolkning_per_fylke_2`

[]:

1.14 Last inn filen "../data/fylkesinndeling.csv" med funksjonen `read.csv2()` og print ut den første raden og kolonnenavnene i filen (filen er lagret med encoding "latin1"). Gjør deretter det samme, men legg til argumentet `header = FALSE` i funksjonen `read.csv2()`. Hva blir forskjellen?

[]:

1.15 Last inn filen "../data/fylkesinndeling.csv" og kall objektet `fylkesinndeling`. Endre navn på kolonnen V1 til `fylke_nr` og legg til ledende null med funksjonen `str_pad()`

[]:

1.16 Legg til navn på fylke ved å koble sammen `befolkning_per_fylke_2` og `fylkesinndeling` med funksjonen `full_join()`. Kall det nye objektet `befolkning_per_fylke_3`

[]:

1.17 Fjern rader med missing (NA) på variabelen `sum`.

[]:

1.18 Restrukturer datasettet `befolkning_per_fylke_3` fra "bredt" til "langt" format. Hint: kolonnene `sum` og `gjennomsnitt` skal slås sammen til én kolonne.

[]:

1.19 Last inn filen `befolkning` fra statistikkbanken ved å kjøre kodesnutten nedenfor og beregn `befolkning` (totalt) for hele landet, per kommune og per fylke i tre separate objekter. Behold kun aktive fylker og kommuner (dvs. som har innbyggere i 2024)

Hint + Region: 0 = hele landet, to siffer = fylke, fire siffer = kommune + Kjønn: 1 = menn, 2 = kvinner

```
[ ]: befolkning <- PxWebApiData::ApiData(07459,
                                         ContentsCode = T,
                                         Region = T,
                                         Kjønn = T,
                                         Alder = T,
                                         Tid = "2024")[[2]]
```

[]:

- 1.20 Bruk befolkning fra forrige oppgave og opprett en ny variabel som heter `aldersgruppe` der alder er gruppert etter følgende inndeling: “0-15”, “16-24”, “25-34”, “35-44”, “45-54”, “55-64”, “65-74”, “75+”

Beregn deretter befolkning per aldersgruppe for hele landet, per kommune og per fylke i tre separate objekter.

[]:

- 1.21 Bruk funksjonen `bind_rows()` for å legge sammen de tre objektene du opprettet i forrige oppgave til ett nytt objekt som heter `befolkning_per_aldersgrupper`.

[]:

- 1.22 Lagre objektet `befolkning_per_aldersgrupper` i mappen “../data/” i .csv-format og les deretter inn filen for å kontrollere at filen ble lagret riktig. Lagre også filen i .xlsx og .parquet-format (og les inn filene etterpå)

[]:

[]:

- 1.23 Åpen oppgave: Les inn et datasett du jobber med og bruk funksjonene `rename()`, `select()`, `filter()`, `group_by()`, `summarise()` osv. til å gjøre gjøre oppgaver du vanligvis må løse.

[]:

[]: