

# Oppgave 2B

Oppgave 2 er en app som returnerer 10 linjer med 2 ord, hvor det andre ordet begynner med 3 bokstaver av slutten av første ordet.

Siden begge ord skal være lengre enn 6 bokstaver har jeg brukt `words.filter` til å filtrere ordene i listen og laget ny liste `randomWords`

`randomNumber` er en funksjon som tar en liste som verdi og returnerer et number basert på listens lengde.

`randomWord` bruker `randomNumber` med `randomWords` som en parameter for å gi oss et number basert på `randomWords` som vi kan bruke som indeks til å returnere et tilfeldig ord.

`generateWordPuzzle` starter en løkke med `wordCount` som teller starter på 0 og løkken løper så lenge `wordCount` er mindre enn 10. Inne i løkken sjekker vi om `drawPuzzle()` med `randomWord` funksjonen som en parameter har returnert true, øk `wordCount` med 1, ellers løper løkken igjen uten å øke `wordCount`.

`drawPuzzle` tar imot et ord som parameter og starter en løkke som går over alle ord i `randomWords` listen. Inne i løkken har vi en if som sjekker at `findMatch` returnerer true med ordet som er gitt som parameter og ordet fra løkken. Hvis `findMatch` er sant, blir det laget en variabel `firstPartOfWord` med 3 bokstaver av første ordet og variabel `lastPartOfSecondWord` med 3 bokstaver trekket fra begynnelsen av andre ordet. Vi legger til bodyen med `puzzleHtml`, gir den første ordet, andre ordet og de 2 variablene som parameter, og returnerer true.

`findMatch` tar 2 ord som parametre. Den lager en variabel `lastPartOfWord` med 3 siste bokstaver av første ord som verdi. Vi sjekker at første ord er ikke lik andre ordet og at andre ordet starter med variabelen `lastPartOfWord` med if og returnerer true.

`puzzleHtml` tar imot 4 parametre og returnerer en div med `firstPart` og `secondPart` som verdi og en knapp som `onClick` bytter ut html med `firstWord` og `secondWord`.

Ved å bruke refactoring har koden blitt mer lesbar og ansvaret har blitt fordelt på funksjonene. Dette gjør at hver funksjon bare har en jobb, som gjør det lettere å forstå og feilsøke hvis koden feiler.