

Opgave 9.1

Lav et array med en række person objekter. Hver person skal have et navn, en alder og et mobilnummer. Løs nedenstående vha. arrays højere ordens metoder.

Find person med et bestemt mobilnummer.

Find den mindste alder.

Modificer arrayet med personer, så personerne nu får et fortløbende id.

Generer en tekststreng, der indeholder personernes navne – kommasepareret og i sorteret rækkefølge.

Generer et array med navn og mobilnummer på de personer, der er yngre end 30.

(Du kan få brug for dette link:

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Functions/Arrow_functions under advanced syntax)

Opgave 9.2

Definer en *compareSort* funktion, der tager en *compare* funktion som parameter og returnerer en sort funktion, der bruger denne *compare* funktion til at sortere et array af strenge.

Brug *compareSort* funktionen til at generere en *lenSort* og en *ignoreCaseSort* funktion, der bruger *compareLen* hhv. *compareIgnoreCase* fra opgave 2.3.

Opgave 9.3

Modificer *compareSort* funktionen, så den returnerede sort funktion nu validerer sin parameter.

Opgave 9.4

- Lav en function *Compose*(f1, f2, a), der tager to funktioner og et tal som parametre og returnerer f1(f2(x)). Altså beregner f2(x) og beregner f1(resultatet).
- Lav *compose* om, så den ikke modtager x som parameter, men kun f1 og f2. Den skal så returnere en funktion, der beregner f1(f2(x)). Denne kaldes også f1 o f2.
- Lav *Compose* om (eller lav en ny), der tager et array af funktioner [f1, f2, ...fn] og returnerer f1 o f2 o ... o fn

Opgave 9.5

Implementer et *Observer pattern* på følgende måde:

- Observers* implementeres som funktioner, der kaldes, når de "notifies".
- Et *Subject* defineres som en *subject* funktion, der har et array med *observers* som lokal variabel
- Funktionen *subject* skal desuden definere funktionerne *registerObserver* og *notifyObservers* som indre funktioner og returnere dem som metoder i et object.

Afprøv implementationen med et par observers.