

Oblig 3 – Programmering 1

TEORI:

- Teorioppgave 1: *Dictionary. Beskriv med egne ord hva en dictionary er i python og forklar forskjellen på disse og vanlige lister. Trekk frem fordeler og ulemper ved begge.*

Svar: Dictionaries er enda en måte i Python som vi kan lagre data på. Det jeg tenker på først angående forskjeller på vanlig liste og dictionary er at dictionary lagrer informasjonen som key/value par. Altså, vi har en key, f.eks frukt1 og så har vi da en verdi knyttet til denne key, f.eks. «Banan». Dataen i lister lagres «som den er», uten en key som har en spesifikk value. For å nå dataen i lister bruker vi indeksplasingen til det lagrede elementet vi ønsker å få ut, men i dictionaries bruker vi en kjent key for å hente ut dataen vi trenger. En key i dictionaries er unik, vi kan ikke lage to keys med samme navn, mens med lagring av data i lister så er det mulig å ha duplikater. Dictionaries var en unordered lagringsmåte før python 3.7 men etter dette er denne nå også ordered. Det vil si at dataen vil beholde samme rekkefølge/struktur som når den ble definert i dict.

- Teorioppgave 2: *Funksjoner. Funksjoner er ekstremt nyttige i alle former former for programmering. Diskuter hvorfor dette er tilfellet. Fokuser på hva funksjoner gjør og hvordan de kan benyttes, og hvilke effekter bruken av disse har på hvordan en programmerer kan skrive, lese og opprettholde kode.*

Svar: Funksjoner gjør at vi kan bruke kode om igjen. Det gjør at vi kan skrive mindre kode når vi har en funksjon som utfører en spesiell oppgave. Funksjonen kan gjenbrukes så mange ganger det er nødvendig, både i et nåværende prosjekt, men også gjenbrukes videre i andre grener av samme prosjekt, eller potensielt i helt andre prosjekt.

Funksjoner gjør det mulig å samle koden du ønsker i mindre «deler», som er mer oversiktlig og leselig for oss. Når du har definert en funksjon som fungerer så kan dette også være med på å minske risikoen for feil i senere kode da du gjenbraker funksjonen du vet fungerer.

Funksjonene gjør det også mye enklere å skulle endre på noe til ev. ny funksjonalitet uten å behøve å skrive alt om igjen.

Funksjonene er også fine på den måten at vi kan lage lokale variabler som kan brukes i akkurat den koden der og da. Funksjonen vil ikke bli påvirket av globale variabler, noe som igjen kan minske risikoen for å gjøre feil/modifisere (globale) variabler slik at funksjonen ikke fungerer/har samme funksjonalitet lenger.

Til slutt, funksjoner hjelper oss å skrive penere, renere kode som er lett å håndtere (som nevnt før) da det strukturerer koden på en god og oversiktlig måte.