

case 1 – Lasse Plov Vejl Madsen, Stine Carlslund Nielsen og Safia
Abdukadir Hersi

Introduktion

Den case, som vi arbejder med i denne her opgave handler om virksomheden Climaiden, der blev stiftet i år 2018. Virksomheden arbejder med at hjælpe virksomheder og private med at nedbringe deres CO2 niveau. De har udviklet en app, hvor man ud fra ens forbrugsmønstre se hvor meget CO2 man udleder om året. Den hjælper virksomheder og private med at blive opmærksomme på, hvor de udleder meget CO2 og derved hvor de på sigt kan nedbringe deres CO2 niveau. Vi har fået udleveret Climaiden's regnskab og et Excel ark med anonymiseret data fra en af de virksomheder, som de arbejder sammen med.

I løbet af denne opgave laver vi nogle UML diagrammer og sekvens diagrammer, som ligger til grund for det, som vi til sidst i opgaven koder i Python for at lave vores program. I vores program laver vi en log in funktion med grafer. Man kan logge in og se graferne online og sammenligne forskellige tal i forhold til mængden af CO2, som man udleder. Derudover kigger vi på regnskabet for virksomheden Climaiden. Vi kigger på hvordan deres resultatopgørelse bliver til ved at indsætte deres tal i Excel og lave resultatopgørelsen. Når resultatopgørelsen er lavet, så kigger vi på opsætningen af balancen. Vi kigger på de forskellige aktiver og passiver. Som en del af det ser vi på, hvordan balancen ser ud, når vi tilføjer en app på 1 million i balancen. Balancen skal altid stemme. Det vil sige at der skal stå det samme under aktiver som under passiver for at balancen stemmer.

Som en del af projektet kigger vi på, hvad skal være en del af projektet, og hvad der ikke er altså, hvad er inden for og udenfor scope. Det er også kaldet in og out i forhold til scope. Derudover har vi nogle unresolved ting, som vi gerne vil løse, men det er et spørgsmål om mængden af tid til at løse de enkelte dele. Vi har udfyldt inception deck i bilag, som en del af vores tilgang til at starte projektet. Her har vi snakket om valg af sprog, brugervenlighed, miljø og hvad der kan gå galt.

Regnskab

1) Vis den beregning der resulterer i, at egenkapitalen ultimo 2020 er -361.917 kr.

Taget hensyn til at der kun er indskudt kapital til virksomheden og et overført overskud/underskud fra årets regnskab. Dette forbliver i minus, hvis virksomheden ikke genererer nok omsætning til at dække deres totale udgifter i resultatopgørelsen.

Grundet et overført underskud på -401916 kr. er egenkapitalen negativ.

2) Afkastningsgraden viser, hvor stor en procent resultatet før finansielle omkostninger udgør af de samlede aktiver.

Beregn den for de første 2 regnskabsår, og knyt et par kommentarer til de 2 tal.

Afkastningsgraden kendes som et nøgletal, som viser om virksomheden kan generere et overskud med den indskudte kapital. På engelsk kaldes dette for **ROI - Return of Investments** - og bliver holdt nøje øje med af investorer, som potentielt kunne tænke sig at investere i virksomheden for at få lidt profit / renter for deres lån til virksomheden. Dette beregnes på følgende måde:

Ud fra de to regnskabsår, har vi følgende afkastningsgrader:

Vi kan inkludere nogle ekstra beregninger ift. Egenkapitalens forrentning, da den kan bruges til at se om virksomheden taber på sin fremmedkapital. Dette kommer vi ikke nærmere på i opgaven.

3) Soliditetsgraden (el. solvensgraden) viser, hvor stor en procent egenkapitalen udgør af de samlede aktiver.

Beregn den for de første 2 regnskabsår, og knyt et par kommentarer til de 2 tal.

Soliditetsgraden viser hvor solvent en virksomhed er - om de er i stand til at bære et større tab. Forenklet beregner den hvor meget kapital virksomheden kan miste uden at røre fremmedkapitalen. Høj soliditetsgrad = store tab kan klares. Lille soliditetsgrad = meget ustabilt og kan ikke tåle store tab.

Soliditetsgrad på engelsk er **Solvency ratio**.

Virksomheden kan altså ikke tåle store tab uden at skulle røre ved fremmedkapitalen.

De skal være opmærksomme på deres poster på *andre tilgodehavender*. Eftersom vi ikke ved hvor de tilgodehavender skulle komme fra, kan vi ikke vide med sikkerhed om de kan få dem tilbage. Firmaskatten er hvad de har til gode hos skat pga. de har haft negative resultater. Det kan man snildt overføre til andre år, så man som virksomhed ikke får et stort skattesmæk på et senere tidspunkt. Dog vil jeg være opmærksom på posten *penge i bank eller fysisk*, da vi ikke kan se, hvor pengene skulle stamme fra. Hvis de ikke er genereret og registreret på en resultatopgørelse, hvor stammer pengene så fra?

4) Kom med et bud på, hvad værdien af selskabet er, hvis der havde været en anmærkning om "going concern" i regnskabet.

Hvis et firma har fået anmærkningen om "going concern" i regnskabet, skal ledelsen beslutte sig om det er en idé at lade driften fortsætte eller ej. Det er en Hovedsagelig negativ ting at se i et regnskab, da det er en formel advarsel fra revisoren, at hvis virksomheden ikke allokerer sine prioriteter i virksomheden til problematikkerne, vil deres firma gå ultimativt konkurs. Ergo vil værdien af selskabet være negativt.

5) Hvilke informationer i noterne kan give os bedre forståelse af ovenstående 4 spørgsmål?

Beskrevet på spg. 4.

6) Hvilke informationer er ikke til stede i regnskabet, men som ville være gode at have? - og hvorfor?

- Faste aktiver (*anlægsaktiver* og *omsætningsaktiver*) : *transportation (hvis de fx skal køre ud til en virksomhed, et foredrag et andet sted eller noget helt tredje), inventar (bruger de deres egne private computere eller har de inventar i et kontor?) Har de eventuelt investeret i en bygning, så de evt. kunne slippe for at leje sig ind på et kontor?* Det ville hjælpe dem med at spare penge i det lange løb, men jeg tvivler stærkt på om de kan få lov til at låne til at eje et sted på stående fod med deres to forgangne resultater. Har de eventuelle likvider, som kunne hjælpe med at betale af på fx en kommende eller uforudset gæld - ergo et større tab. Det ser ikke sådan ud ift. soliditetsgraden. Materielle anlægsaktiver er fx biler, inventar, computere, investeringsejendomme, etc.

- Gældsforpligtelser - både langfristede (5+ år) og kortfristede gældsforpligtelser (indenfor 0 - 5 år)
- Kreditorer - skylder de penge andre steder.
- Hvad var deres omsætning før vareforbrug? Kunne godt være at man havde tjent meget, men har brugt for mange varer til at have en god **dækningsgrad**.

7) Antag, at virksomhedens app er værdi sat til 1 mio.kr. Hvordan skal dette registreres i regnskabet?

Det kommer kraftigt an på om appen er solgt, hvor der er betalt for appen - 'solgt' og virksomheden har nogle penge som tilgodehavende eller om det er en værdi som et aktiv. Vi forholder os til at det er et aktiv i virksomheden – altså, at de har udviklet appen og bruger den til at generere omsætning.

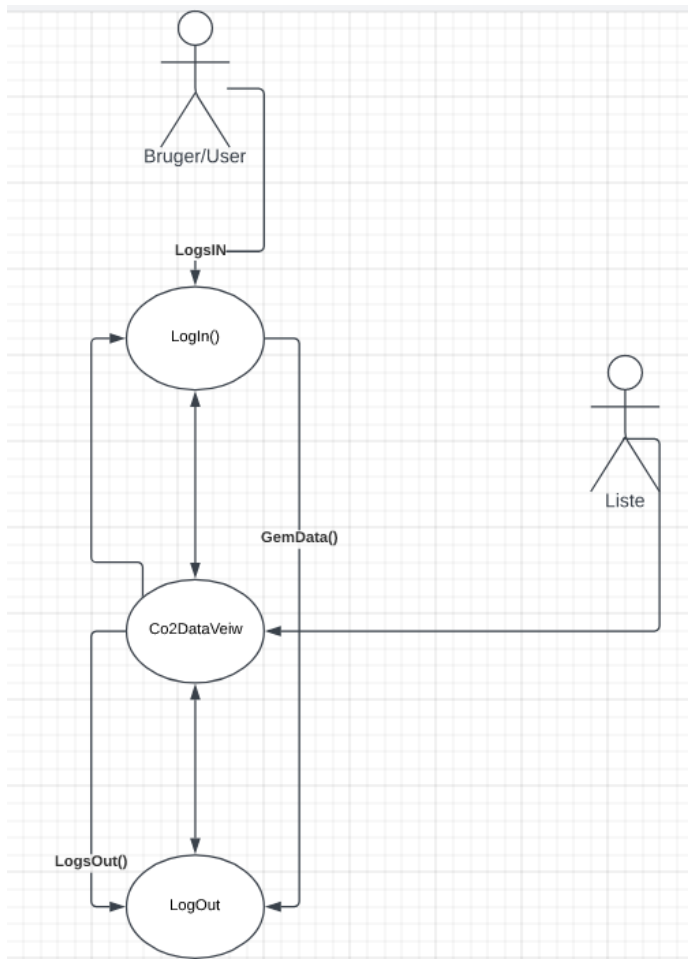
Eftersom virksomheden er udviklere af appen og nok gerne vil tage imod betalinger ift. betalingsløsninger fra deres brugere, er det et immaterielt anlægsaktiv. Det er en form for anlægsaktiver, som er skattet til brug i virksomheden, uden de er fysiske eller finansielle. Eksempler på dette er Goodwill, varemærker, patenter, mv. Appen er et produkt som virksomheden har som patent! Dvs. De må bruge det til deres finansielle fordel - de bestemmer om det skal forblive brugt som det er nu eller om de vil sælge denne app. Debet på immaterielle anlægsaktiv og kredit på egenkapitalen

8) Hvordan vil svaret i spørgsmål 7 påvirke Afkastningsgraden, Soliditetsgraden og værdien af selskabet (spørgsmål 2,3,4)

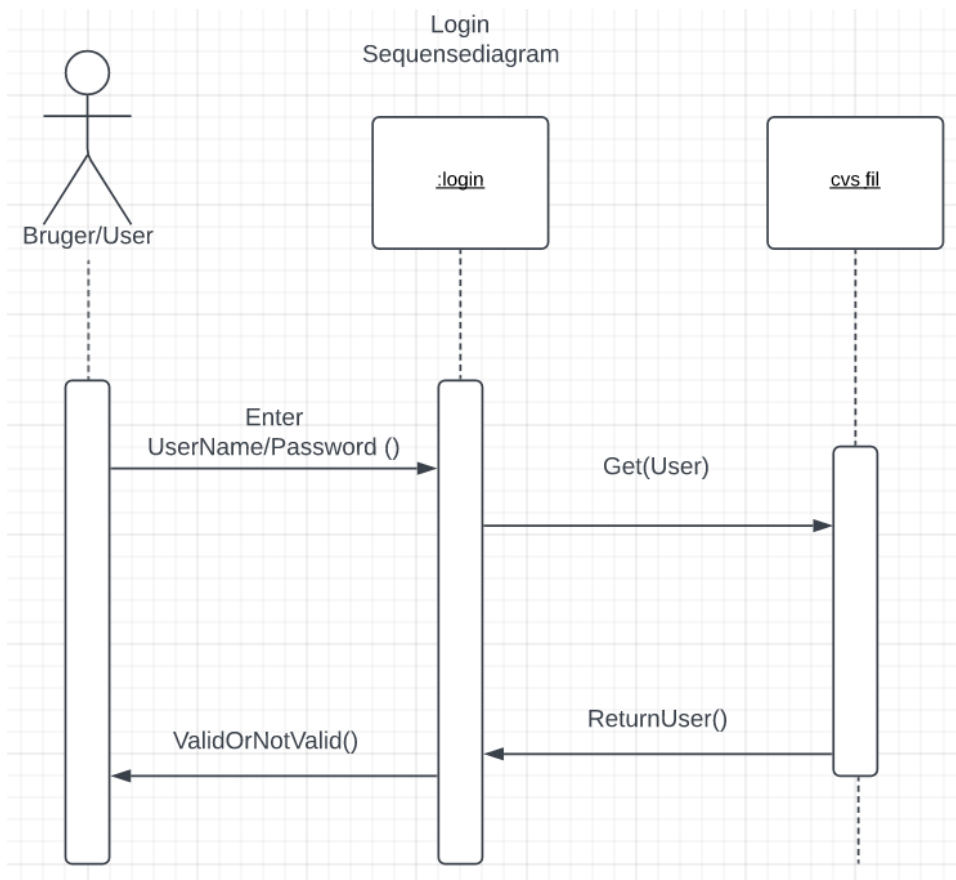
Med appens værdi på 1. mio. kr. ville det påvirke virksomhedens afkastningsgrad, soliditetsgrad og værdien af selskabet på følgende metode med udgangspunkt i 2020:

Virksomheden ville være bedre stillet, hvis det var realiteten at appen havde en værdi til 1 mio. kr., men deres afkastningsgrad og soliditetsgrad er stadig ikke tilfredsstillende. Selvom virksomheden har hjertet det gode sted, skal de prioritere hvorvidt de vælger at udvikle deres firma. Der er ingen tvivl om, at der er efterspørgsel for denne slags app med den stigende mængde af klimaentusiaster, så om de skal prioritere deres betalingsløsninger, sælge procentdele af appens værdi til investorer der vil kunne se appens værdi stige markant om et par år, eller noget helt tredje - er noget ledelsen skal tage stilling til.

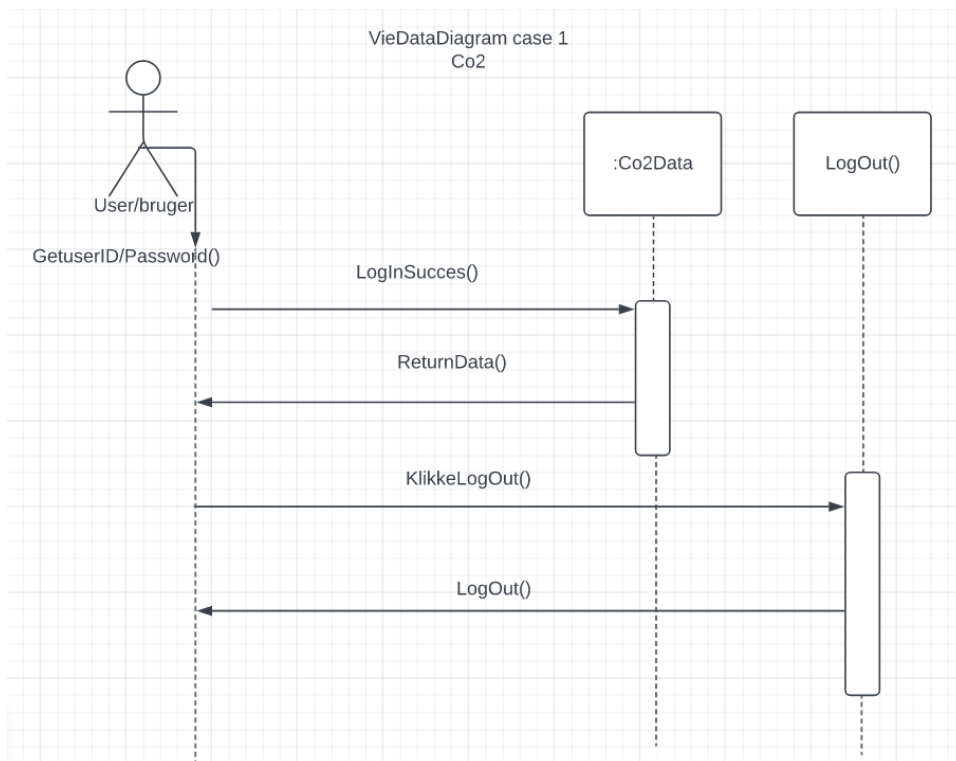
Vi beskriver nu, hvad vi har tænkt os at programmere først ved hjælp af et flowdiagram og efterfølgende to sekvens diagrammer. Flowdiagrammet viser, hvordan brugeren logger på systemet, hvorefter vedkommende ser dataen og kan se data og grafer. Log af funktionen er tilgængelig hele vejen igennem så snart vedkommende er logget på.



Sekvensdiagrammet viser klasserne som login og csv fil. Derudover viser det de handler, der sker i systemet, når vedkommende vil logge på



Det efterfølgende sekvensdiagram viser de handlingerne, der sker efter og i forbindelse med at Log in er en succes. Vi ser handlingerne fra Log in er en succes indtil man laver beregninger for grafer og logger af igen.



Python programmering

For at kunne køre koden installeres Flask og Render_Templet til at kunne gøre det til læselig data.

Programmeringsdelen med at lave grafer

Først importeres modulerne, som vi skal bruge. Det drejer sig om modulerne Pandas, numpy og matplotlib. Herefter indlæser vi Excel arket, som vi har fået til casen med data om CO2.

```
# Excel fil indlæses i Python
df = pd.read_excel('Modul1.1CaseVirksomhedsberegning.xlsx', sheet_name='Beregning', usecols="A,B,F,G,H,I,J,K,L,M,O,P,Q,R,S,T,U,V")
```

For at kunne bruge dataen i nogle af kolonnerne, så er vi nødt til at ændre nogle af kolonnenavnene. Vi ændrer mellemrummet til _.

```
# Denne kode ændrer på kolonne navnene
df = df.rename(columns={'Udledning faktor': 'Udledning_faktor',
                        'Total enheder for året': 'Total_enheder_for_året'})
```

Herefter grupperer vi dataen hvorefter vi kalder enkelte kolonner ved hjælp af

```
Ttit = df.loc[:, 'Titel'].values  
Tot = df.loc[:, 'Total_enheder_for_året'].values  
Udledning = df.loc[:, 'Udledning'].values
```

Vi bruger de enkelte kolonner senere, når vi laver grafer. Eftersom vi ikke kan lave grafer med strings, så laver vi en sampleList

```
sampleList = [i for i in range(41)]
```

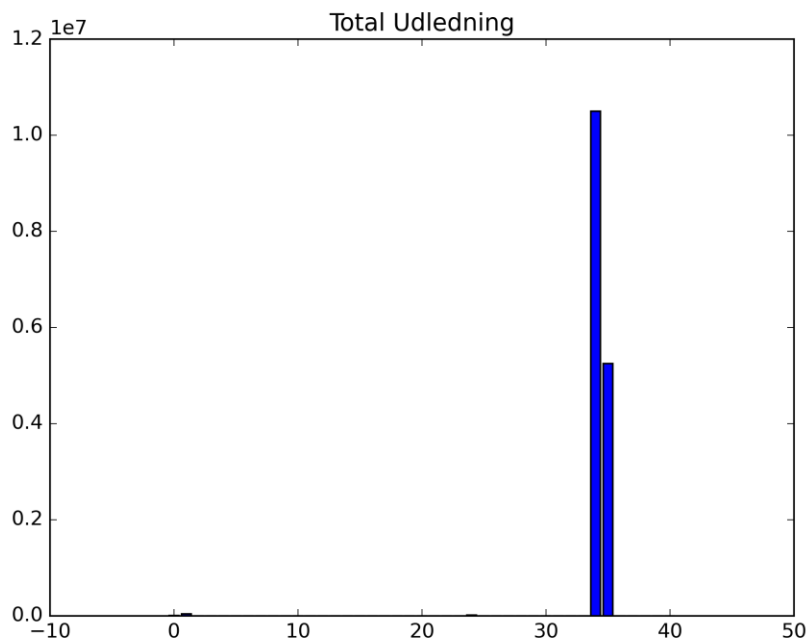
Den gør at vi har antallet af tal fra 0 til 41, da Python 0 indexerer. Det betyder at en talrække starter ved 0 og går op ad og ikke som vi normalt tænker fra 1.

Herefter laver vi en tom liste for tit (titel) for så bagefter at fylde den op. Det giver en liste at arbejde med væk fra Excel arket, som vi så kan begynde at lave nogle grafer med. For at være sikre på at der er tal i alle rækker, så kan man bruge

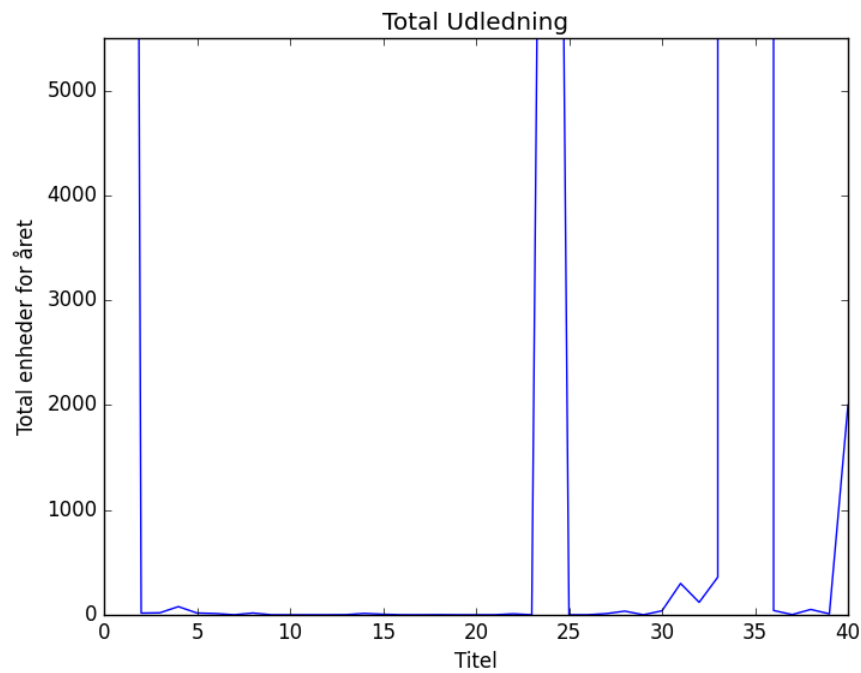
```
df.info()
```

Det giver derudover informationer om antallet af kolonner, hvilken datatype har sammen med navnene på kolonnerne og antallet af rækker, der er i hver kolonne.

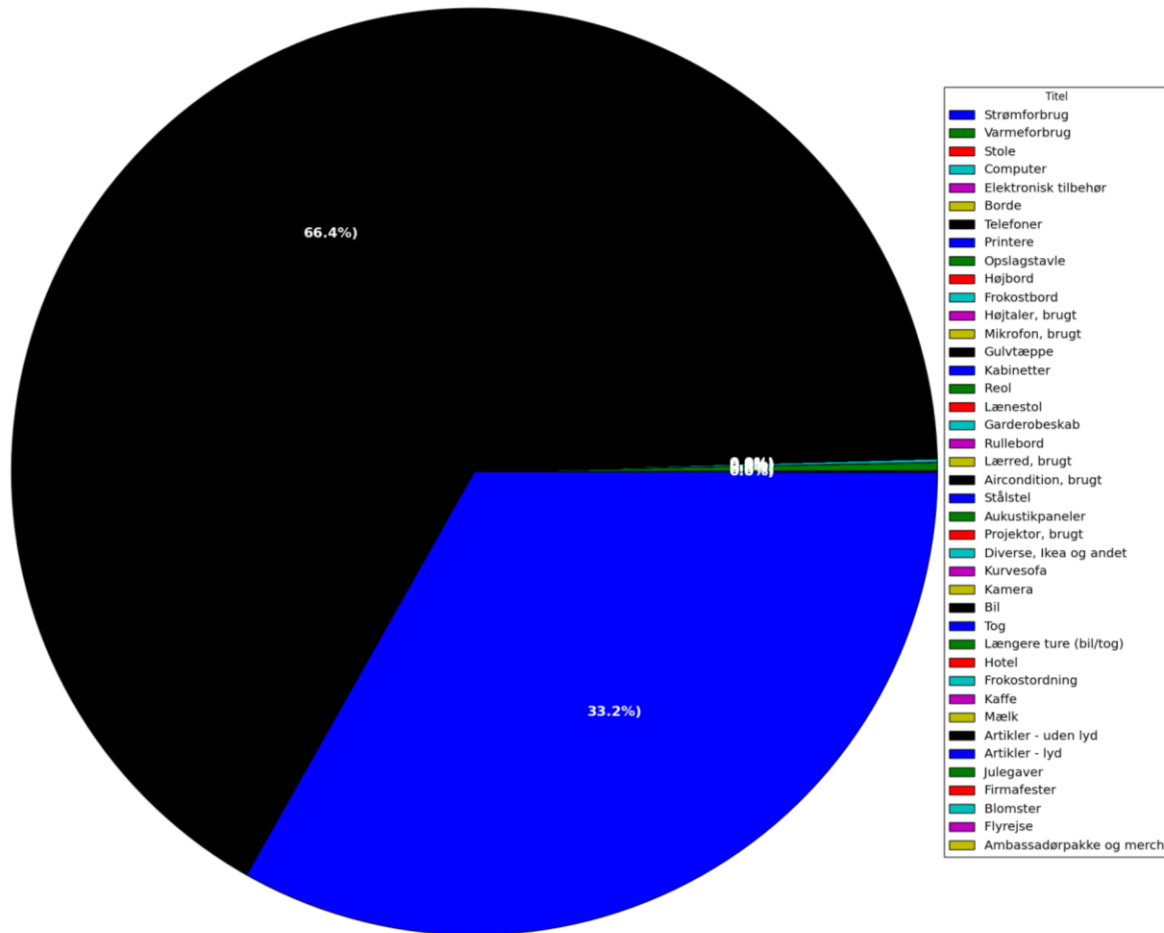
Nu kan vi så begynde at lave grafer.



Her laver vi et søjlediagram der viser den totale udledning. Det er så totale enheder for året, og her kan man også se der er artikler-uden lyd der er højest, og artikler-lyd er næst højest.



Den her graf viser det samme som den første men den er bare mere zoomet ind og er en anden type graf.



De to største områder er "Artikler - uden lyd" og "Artikler - lyd". Grafen viser totalenheder for året. Og den er så . Den sorte er den højeste. Det er "artikler - uden lyd". Den næste højeste det er den blå, og det er så "artikler - lyd". Og de andre er lidt svær og se fordi de er meget små i forhold til de store.

Vi kan konkludere at det er lykket at lave noget kode, hvor vi har arbejdet med en loginfunktion, der virker sammen med indlæsning af Excel i kode

Hvis vi havde har lidt mere tid var planen at finpudse vores grafer lidt mere og tilføje flere grafer. Derudover ville vi have arbejdet noget mere med flask.

```

    tor for en route så den får Kategori Transport
    te("/Transport")
    sport():
    df = pd.read_excel('Modul1.1CaseVirksomhedsberegning.xlsx', sheet_name='Beregning', usecols="A,B,F,G,H,I,J,K,L,M,O,P,Q,R,S,T,U,V")
    print(df)
    return( '<center><h1>Transport</h1></center> '
            '<br /><br /><a href="/">Transport<button>Se Transport</button></a>'
            '<a href="/Transport"><button>Se Transport</button></a>'
            '<a href="/Grafer"><button>Se Grafer</button></a>'
            '<br /><br /><a href="/"><button>Tilbage til forsiden</button></a>'
            '<br /><br />' + render_template('htmlTable.html', data = df ))

```

Her kan man se lidt kode på en knap i vores system, som kan bruges til at fører videre til en graf, i dette tilfælde er Co2 Udledningen for transport fra vores excel.

```

# For at køre koden direkte
if __name__ == '__main__':
    app.run()

```

Denne funktion bruger vi til at kalde på vores Flask, og kører Koden på en hjemmeside. Vi har også noget login() men vi konkluderede, at det var out of scope.

Referencer

Dyson, J. R., & Franklin, E. (n.d.). *ACCOUNTING for Non-Accounting Students*.

Jonathan. (n.d.). *What Readers Are Saying About The Agile Samurai*.

Lone Hansen,

Milliken, C. P. (2020). Python Projects for Beginners. In *Python Projects for Beginners*. Apress.

Erhvervsøkonomi til akademiuddannelserne. (n.d.). Retrieved March 25, 2022, from <https://erhv-oekon-akademi.hansreitzel.dk/>

Bilag:

Case_1.py

BeregningerAfleveret.py

Regnskab Excel ark – Excel ark, hvor tallene fra det udleverede regnskab er blevet sat ind i Excel for at vise beregningerne af de forskellige tal i først resultatopgørelsen og herefter balancen og regne beregningerne for soliditetsgrad og afkastningsgrad.