

# EGENREFLEKSJON OG RAPPORT FOR YFF

## FYLLES UT VED OPPSTART

### Problemstilling

Jeg ønsker å lære hvordan jeg kan lage visuelle applikasjoner i Python ved bruk av Tkinter. Det betyr at jeg må lære meg Tkinter og alt innover der.

### Kompetansemål?

I dette prosjektet så vil jeg planlegge alt mye av det jeg skal gjøre på forhånd. Sette meg godt inni Tkinter og eksperimentere litt med koden min. Være effektiv sånn at jeg rekker alt jeg skal gjøre før tiden er oppe.

### Om prosjektet

I dette prosjektet så skal jeg prøve å lage en mail app ved bruk av Python kodespråket. Jeg skriver koden inni VS Code, og skal bruke Tkinter/CustomTkinter for å lage selve visuelle applikasjonen.

### Mitt utgangspunkt

Jeg har hatt noen tidligere prosjekter med Python, men jeg har aldri jobber med Tkinter modulen før, så det blir noe nytt å lære seg.

### Plan for gjennomføring

#### Gang 1

Dette er den første gangen og målet mitt er å laste ned Tkinter og sette opp en innloggingsmeny.

#### Gang 2

Denne gangen så har jeg planlagt å legge til de forskjellige menyene i programmet og å sette opp en liste med de mottat epostene fra en epostadresse.

#### Gang 3

Denne siste gangen så skal jeg mulighvis fikse litt på koden, og gjøre sånn at det kjører bedre. Hvis det er noe som mangler eller kan være bedre så fikser jeg på det. Prøve å vise eposten visuelt.

## SKRIVES HVER ØKT

### Logg / fremgangsmåte

*Skriv logg etter hver økt hvor du beskriver hva du har gjort.*

*Hvilke steg har du tatt for å komme der du nå er i kodingen/prosjektet?*

*Legg inn bilder/skjermdump som viser prosessen.*

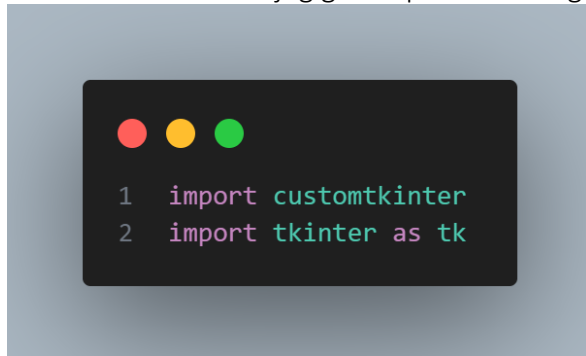
## Logg uke 1

For å starte denne uken så måtte jeg laste ned selve Tkinter. Jeg tror den burde være lastet ned med Python men det denne gangen var den ikke det. Da gikk jeg inn i ledeteksten på Windows og skrev inn disse kommandoene for å laste ned Tkinter og CustomTkinter.

```
C:\Users\Olive>pip install Tkinter
```

```
C:\Users\Olive>pip install Customtkinter
```

Etter dette så kunne jeg gå inn på VSCode og starte med å kode.



Jeg la til moduelene som jeg akkurat lastet ned inni VSCode med koden over. Etter dett så kunne jeg starte å kode på Tkinter. Den første tingen som føltes viktig var å lære moduelen, og for å gjøre dette så koder jeg en knapp.

Denne knappen kan mann trykke på men ingenting skjer. Jeg kodet på måten under, som du kan se er egentlig ikke så avansert som man hadde trodd. Først må jeg definere knappen, og så bruke «Pack» for å sette knappen inn sånn at man kan se den.

```
import customtkinter
import tkinter as tk

Kinter = tk.Tk()
Kinter.title("Hallo")
Maxwidth = Kinter.winfo_screenwidth()
MaxHeight = Kinter.winfo_screenheight()
Kinter.minsize(1280, 720)

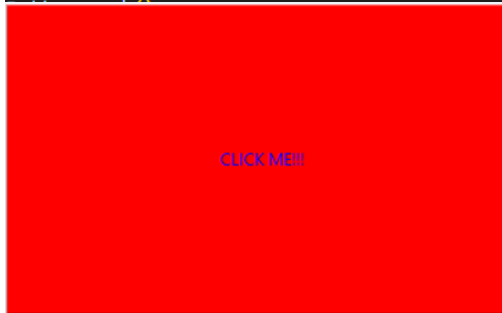
Kinter.maxsize(Maxwidth, MaxHeight)
Button = tk.Button(Kinter, text="CLICK ME!!!", pady=100, padx=150, )
Button.pack()
Kinter.mainloop()
```

Jeg endret størrelsen av knappen ved å skrive «pady» som er for y aksen og «padx» for x aksen.



Etter dette så la jeg på en rød bakgrunn og skiftet fargen på teksten fra svart til blå. Dette igjen krevde lite kode for å gjøre.

```
Kinter.maxsize(Maxwidth, MaxHeight)
Button = tk.Button(Kinter, text="CLICK ME!!!", pady=100, padx=150, background="Red", foreground="Blue")
```



Det jeg gjorde etter dette var bare å legge på noen til «entryer», og noe som heter «labels». Man skriver disse inn på lik måte og labler oppfører seg om tekst som kan være ved siden av eller over en knapp. Entrier er bokser hvor man kan skrive inn ting som i denne oppgaven passord og epost adresse.

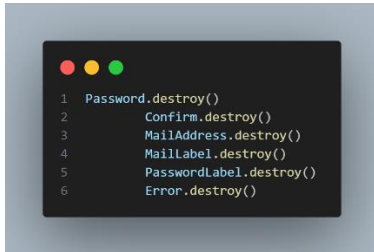
Jeg kan sette opp et passord system ved å bruke funksjoner. Den jeg har under finner hva passordet og epost adressen er og da ser om det er riktig. Over her igjen setter jeg bare bakgrunnsfarger til hvit som er den allerede når man starter et Tkinter vindu.

```
1 Kinter.configure(bg="White")
2
3 def Check():
4     PassCheck = Password.get()
5     AddressCheck = MailAddress.get()
6     if PassCheck == "Dog":
7         print("Working")
8
9
10 Kinter.maxsize(Maxwidth, MaxHeight)
11 MailLabel = tk.Label(text="Email Address")
12 MailAddress = tk.Entry()
13 PasswordLabel = tk.Label(text="Password")
14 Password = tk.Entry()
15 Confirm = tk.Button(text="Log in", background="Red", command=Check)
16 MailLabel.pack()
17 MailAddress.pack()
18 PasswordLabel.pack()
19 Password.pack()
20 Confirm.pack()
21
22
23 Kinter.mainloop()
```

Det var det for denne uken. Den eneste tingen som jeg ikke nevnte var at jeg gjorde hvis vinduet ble større en maks oppløsningen så løste vinduet seg.

## Logg uke 2

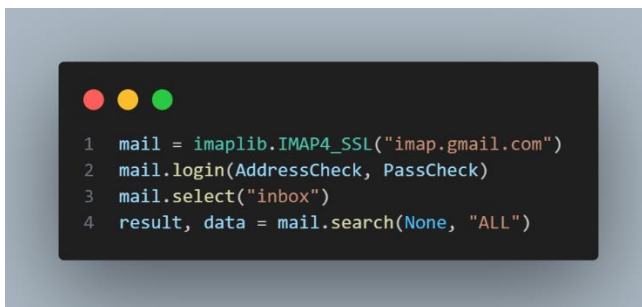
Dette var den andre uken av dette prosjektet, og veldig mye skjedde. Hva jeg gjorde på starten var å gjøre sånn at når man var ferdig med å skrive inn eposten og passordet så ble man sendt til en ny side.



Da måtte jeg slette alle elementene som lå på skjermen som jeg gjør med koden over. Etter dette så endret jeg sånn at man ikke trengte å ta inn det riktig eposten og passordet som jeg satt opp. Da til slutt skiftet jeg fargen av siden til blå for å vise at man var på en ny side.



Før jeg gjorde noe annet så følte jeg at det hadde vært best å gjøre den vanskeligste delen med en gang som var å skaffe alle epostene. Da lagde jeg en funksjon hvor jeg kunne putte inn alt koden som skulle hente epostene.



De første linjene i funksjonen finner hvilket type epost du har, og fordi jeg skal prøve det med min gmail så velger jeg det. Etter det så bruker vi den tidligere innlogginginformasjonen for å logge inn på gmail. Etter det så går vi inn på inbox, og da putter vi alle epostene som ble funnet inni variabel. Jeg bruker modulen imaplib for å gjøre alt dette.

```

1  if result == "OK":
2      email_message = message_from_bytes(message_data[0][1])
3
4      sender = email_message.get("From", "Unknown Sender")
5      subject = email_message.get("Subject", "No Subject")
6      date = email_message.get("Date", "Unknown Date")
7
8
9      email_data = {
10         "ID": id,
11         "Sender": sender,
12         "Subject": subject,
13         "Date": date
14     }
15     all_emails_data.append(email_data)
16     id += 1
17
18     with open("JSON.json", "w") as file:
19         json.dump(all_emails_data, file)
20
21     print(f"Sender: {sender}")
22     print(f"Subject: {subject}")
23     print(f>Date: {date}")
24     print("-----")

```

Målet mitt med denne gangen var å alle epostene mine kategorisert inni en JSON fil. Den holder datasett, så jeg kan ha navnet på senderen, hva det er om, og sånt der. Da for å få dette til, så må jeg først hente navnet av senderen, hva eposten handler om, og til slutt hvilket dato den ble motatt.

Email message er variabelen over som forteller oss det vi trenger for å finne datoen osv. Jeg bruker også «Message\_from\_bytes» som er fra email pakken i python. Uansett så etter det her så sette jeg opp en variabel som heter «email\_data». Det er denne jeg bruker for JSON.

```

1  for email_id in email_ids:
2      result, message_data = mail.fetch(email_id, "(RFC822)")

```

Løkken over kjører over alle epostene. Dette betyr at email data kan legge på ny sender, dato, osv. Til slutt så blir denne variabelen veldig stor fordi den da inneholder masse informasjon. Så for å si det enkelt så har jeg en løkke som kjører en gang for hver epost i inboxen, og så legges informasjonen om posten på en variabel.

```

1  with open("JSON.json", "w") as file:
2      json.dump(all_emails_data, file)
3
4  print(f"Sender: {sender}")
5  print(f"Subject: {subject}")
6  print(f>Date: {date}")
7  print("-----")

```

Koden etter det lager en JSON dokument, og så dumper alt dataen fra epostene inni dokumentet. Etter det igjen så printer jeg det også.

```

1 {
2   "ID": 37,
3   "Sender": "Team Anima <no-reply@animaapp.com>",
4   "Subject": "How is Anima going for you so far?",
5   "Date": "Fri, 08 Dec 2023 10:00:04 +0000"
6 },
7 {
8   "ID": 38,
9   "Sender": "LALAL.AI <noreply@lalal.ai>",
10  "Subject": "New! Record HQ Audio & Video Enhanced with LALAL.AI Noise Removal",
11  "Date": "Fri, 08 Dec 2023 15:19:32 +0000"
12 },
13 {
14  "ID": 39,
15  "Sender": "\"Kahoot!\" <kahoot@team.kahoot.com>",
16  "Subject": "=?utf-8?B?5GV5LCBoZCJlIGFyZSAzIHdheXQgdG8yY3JlYXIlIGxhc3Rpbmcg?=\\r\\n\\t=?utf-8?B?bWVtb3JpZXMgdGhpcy8ob2xpZGF5IHh1YXNvb1lg8j+0gQ==?=",
17  "Date": "Fri, 08 Dec 2023 18:10:46 +0000"
18 },

```

I JSON filen, så det sånn her ut når det var lagt inn.

```

1 fetch_button = tk.Button(Kinter, text="Fetch Emails", command=display_emails)
2 fetch_button.pack()

```

Koden over lager en knapp som kjører funksjonen. Som du kan tenke så står det at den henter epostene.

Det er en siste ting som er bare at jeg lagde sånn at når man tar feil adresse eller passord så får man en feilmelding, og man må prøve på nytt igjen. Akkurat så kan du logge inn uansett hva du skriver inn men du får en feil i terminalen.

Til slutt så brukte jeg Chat-GPT litt på grunn av at mye dette her er nytt, og jeg trengte litt hjelp med å forstå det. Den også hjalp mye fordi når jeg fikk alt informasjonen fra gmail, så kom det meg masse annet også. Chat-GPT hjalp med å fjerne dette og med å kategorisere epostene.

## Logg uke 3

Nå var det den siste uken av prosjektet og jeg hadde egentlig ikke så fryktelig mye igjen å gjøre. Det første jeg gjorde var å fikse på programmet. Jeg følte at det som var viktigst var at hvis det ikke var riktig passordet eller eposten så får du en feilmelding.

```

1 except Exception as e:
2     print(f"Error: {e}")
3     isloggedin = False

```

Koden over kjører når vi prøver å logge inn men vi får ikke noe svar tilbake igjen. Da skriver vi ut feilen vår, og sier at vi ikke er logget inn med variabelen, som lar at vi kan prøve å logge inn på nytt igjen.

Den neste tingen var å sentrere log inn skjermen. For å oppnå dette så brukte jeg nett eller grid. Da kan jeg endre rekken, og sånne ting. Dette gjør sånn at jeg kan plassere ting hvor jeg vil.

```

1 login_frame = tk.Frame(Kinter, bg="White")
2 login_frame.place(relx=0.5, rely=0.5, anchor=tk.CENTER)
3
4 Login_Text = tk.Label(login_frame, text="LOG IN")
5 MailLabel = tk.Label(login_frame, text="Email Address")
6 MailAddress = tk.Entry(login_frame)
7 PasswordLabel = tk.Label(login_frame, text="Password")
8 Password = tk.Entry(login_frame, show="*")
9 Confirm = tk.Button(login_frame, text="Log in", background="Red", command=Check)
10 Error = tk.Label(login_frame)
11
12 Login_Text.grid(row=0, column=1, pady=5)
13 MailLabel.grid(row=1, column=0, pady=5, sticky=tk.E)
14 MailAddress.grid(row=1, column=1, pady=5, padx=5)
15 PasswordLabel.grid(row=2, column=0, pady=5, sticky=tk.E)
16 Password.grid(row=2, column=1, pady=5, padx=5)
17 Confirm.grid(row=3, column=0, colspan=2, pady=10)
18 Error.grid(row=4, column=0, colspan=2)

```



Bildene over viser mye av koden for å oppnå dette. Bildet til høyre er sånn det ser ut i programmet.

For nå så er denne menyen ferdig lagd, og jeg kan gå over til selve eposten min. Her er målet mitt å vise det visuelt og når du trykker på det så kommer opp noe informasjon om posten.

```

1 email_variables = all_emails_data
2
3
4 email_listbox.delete(0, tk.END)
5
6
7 for email_data in all_emails_data:
8     email_listbox.insert(tk.END, email_data)

```

En av tingene jeg gjorde for å få det til å fungere var å lagre de forskjellige gruppene av informasjonen for hver ID inni Json dokumentet til «dictionaries». Da har jeg veldig lett tilgang til det forskjellige epostene og informasjonen inni dem.

```

1 {
2     "ID": 11,
3     "Sender": "\"Kahoot!\" <kahoot@team.kahoot.com>",
4     "Subject": "Welcome to Kahoot!",
5     "Date": "Thu, 12 Oct 2023 06:48:21 +0000"
6 },
7 {
8     "ID": 12,
9     "Sender": "Freesound NoReply <noreply@freesound.org>",
10    "Subject": "[freesound] Your activation link",
11    "Date": "Fri, 13 Oct 2023 06:50:32 +0000"
12 },
13 {
14    "ID": 13,
15    "Sender": "LALAL.AI <noreply@lalal.ai>",
16    "Subject": "LALAL.AI Orion: New Stem Splitting AI",
17    "Date": "Mon, 16 Oct 2023 18:07:41 +0000"
18 }

```

Inni Json dokumentet ser det sånn her ut.

I bildet over så har jeg en løkke som kjører gjennom hver epost, og legger informasjonen inni en «listbox».

```

1 def display_email_info():
2     selected_index = email_listbox.curselection()
3     if selected_index:
4         email_id = selected_index[0] + 1
5         email_info = email_variables[email_id - 1]
6         messagebox.showinfo("Email Information", email_info)

```

Hva jeg gjør i denne koden her er at når man trykker på en epost inni appen så kommer opp en liten box som viser bare generell informasjon. Hvis videreutviklet så kan dette vise informasjonen inni eposten.

Det siste da var å bare å lage det i visuell form. Koden under er sånn jeg kodet mye av det.

```

1 global email_listbox
2
3 email_listbox = tk.Listbox(Kinter, selectmode=tk.SINGLE, height=20, width=100)
4 email_listbox.pack(fill=tk.BOTH, expand=True)
5
6 scrollbar = tk.Scrollbar(email_listbox, orient=tk.VERTICAL, command=email_listbox.yview)
7 scrollbar.pack(side=tk.RIGHT, fill=tk.Y)
8
9 email_listbox.config(yscrollcommand=scrollbar.set)
10 email_listbox.bind("<Double-Button-1>", lambda event: display_email_info())
11

```

Jeg også la til inni «Listboxen» at du kunne skrolle ned på grunn av det kan være en del eposter i din inbox, og da for å se de, så må du skrolle ned.

D: 1Sender: Google Community Team <googlecommunityteam-noreply@google.com> Subject: =?UTF-8?Q?Your Google Account is live\_=E2=80=93\_now\_help\_your\_busine?= ?UTF-8?Q?ss\_grow?=? Date: Wed, 20 Sep 2023 03:22:34  
D: 2Sender: ElevenLabs <invoice+statements+acct\_1M07hSLmdOdiMXBs@stripe.com> Subject: Your receipt from ElevenLabs #2799-9876Date: Mon, 25 Sep 2023 06:37:54 +0000  
D: 3Sender: Team D-ID <no-reply@d-id.com> Subject: Welcome to the Creative Studio!Date: Mon, 25 Sep 2023 12:19:03 +0300  
D: 4Sender: Team D-ID <no-reply@d-id.com> Subject: Oliver, Your Video is Ready!Date: Mon, 25 Sep 2023 12:21:55 +0300  
D: 5Sender: "Uppbeat" <upbeat@upbeat.io> Subject: Welcome to Uppbeat!Date: Thu, 28 Sep 2023 03:22:31 -0500  
D: 6Sender: "Uppbeat" <upbeat@upbeat.io> Subject: How to avoid copyright claimsDate: Fri, 29 Sep 2023 10:04:32 -0500  
D: 7Sender: "Uppbeat" <upbeat@upbeat.io> Subject: =?utf-8?Q?You haven't downloaded anything yet\_=F0=9F=A2?=? Date: Sun, 01 Oct 2023 10:08:44 -0500  
D: 8Sender: LALALAI <noreply@lala.ai> Subject: Split Audio AND Video Files with LALALAI!Date: Mon, 02 Oct 2023 01:49:04 +0000  
D: 9Sender: LALALAI <noreply@lala.ai> Subject: Split Any File to 10 Stems With LALALAI!Date: Thu, 05 Oct 2023 01:51:02 +0000  
D: 10Sender: LALALAI <noreply@lala.ai> Subject: Get More Out of LALALAI!Date: Sun, 08 Oct 2023 01:53:02 +0000  
D: 11Sender: "Kahoot!" <kahoot@team.kahoot.com> Subject: Welcome to Kahoot!Date: Thu, 12 Oct 2023 06:48:21 +0000  
D: 12Sender: Freesound NoReply <noreply@freesound.org> Subject: [freesound] Your activation linkDate: Fri, 13 Oct 2023 06:50:32 +0000  
D: 13Sender: LALALAI <noreply@lala.ai> Subject: LALALAI Orion: New Stem Splitting AI!Date: Mon, 16 Oct 2023 18:07:41 +0000  
D: 14Sender: "Kahoot!" <kahoot@team.kahoot.com> Subject: =?utf-8?B?3VzIzEhbnB3aXRoIHRoG9vdCg8J+RuyA=?=? Date: Wed, 18 Oct 2023 15:10:33 +0000  
D: 15Sender: "Kahoot!" <kahoot@team.kahoot.com> Subject: How likely are you to recommend Kahoot! to your friends and colleagues?Date: Thu, 26 Oct 2023 06:45:22 +0000  
D: 16Sender: "Free Music Archive" <contact@freemusicarchive.org> Subject: Verify Email AddressDate: Thu, 02 Nov 2023 10:33:20 +0000  
D: 17Sender: LALALAI <noreply@lala.ai> Subject: Try Orion on Drums & Grab 50% OFF + Extra Minutes as Free!Date: Fri, 03 Nov 2023 18:17:36 +0000  
D: 18Sender: LALALAI <noreply@lala.ai> Subject: LAST CHANCE to Save 50% + Get Extra Minutes as Gift!Date: Fri, 03 Nov 2023 18:17:36 +0000  
D: 19Sender: pngtree-recommendations@email.pngtree.com Subject: =?UTF-8?Q?F0=9F=A2?=? Date: Fri, 03 Nov 2023 18:17:36 +0000  
D: 20Sender: pngtree-promotion@email.pngtree.com Subject: =?UTF-8?Q?WAIT, Don't forget your free d=? Date: Fri, 03 Nov 2023 18:17:36 +0000  
D: 21Sender: "Kahoot!" <kahoot@team.kahoot.com> Subject: Hey, Black Friday is almost here! Get the deal be=? Date: Fri, 03 Nov 2023 18:17:36 +0000  
D: 22Sender: pngtree@email.pngtree.com Subject: =?utf-8?Q?F0=9F=A2?=? Date: Fri, 03 Nov 2023 18:17:36 +0000  
D: 23Sender: pngtree-ideas@email.pngtree.com Subject: =?utf-8?Q?F0=9F=A2?=? Date: Fri, 03 Nov 2023 18:17:36 +0000  
D: 24Sender: Epic Games <help@accts.epicgames.com> Subject: Epic Games - Email VerificationDate: Thu, 23 Nov 2023 18:17:36 +0000  
D: 25Sender: LALALAI <noreply@lala.ai> Subject: Black Friday Sale! Grab the 650-Min Pack!Date: Thu, 23 Nov 2023 18:17:36 +0000  
D: 26Sender: "Kahoot!" <kahoot@team.kahoot.com> Subject: Hey, Black Friday is here! Get over 45% off Kahoot!Date: Thu, 23 Nov 2023 18:17:36 +0000  
D: 27Sender: "Kahoot!" <kahoot@team.kahoot.com> Subject: Hey, Cyber Week is here! Get over 45% off Kahoot!Date: Thu, 23 Nov 2023 18:17:36 +0000  
D: 28Sender: "Kahoot!" <kahoot@team.kahoot.com> Subject: Hey, last chance to get over 45% off Kahoot!+ Date: Wed, 29 Nov 2023 10:55:07 +0000  
D: 29Sender: Team Anima <no-reply@animaapp.com> Subject: Welcome to Anima! Start delivering frontend 2x fasterDate: Fri, 01 Dec 2023 10:29:44 +0000  
D: 30Sender: "Free Music Archive (FMA)" <community@tribeofnoise.com> Subject: Urgent: Help Change an Artist's Life Today - Will You Join Us?Date: Fri, 01 Dec 2023 10:31:28 +0000  
D: 31Sender: "Kahoot!" <kahoot@team.kahoot.com> Subject: =?utf-8?B?3VzIzEhbnB3aXRoIHRoG9vdCg8J+agA=?=? Date: Fri, 01 Dec 2023 15:12:18 +0000  
D: 32Sender: LALALAI <noreply@lala.ai> Subject: LAST CHANCE to Save 50% & Get 650 Minutes!Date: Sat, 02 Dec 2023 11:27:25 +0000  
D: 33Sender: Team Anima <no-reply@animaapp.com> Subject: What can you expect from Anima's generated code!Date: Mon, 04 Dec 2023 09:59:45 +0000  
D: 34Sender: "Kahoot!" <kahoot@team.kahoot.com> Subject: Hey, save over 20% on Kahoot!- this festive season!Date: Mon, 04 Dec 2023 20:03:59 +0000  
D: 35Sender: "Kahoot!" <kahoot@team.kahoot.com> Subject: =?utf-8?B?3VzIzEhbnB3aXRoIHRoG9vdCg8J+agA=?=? Date: Thu, 07 Dec 2023 15:12:56 +0000  
D: 36Sender: pngtree@email.pngtree.com Subject: =?UTF-8?Q?20231208The most fashionable png this week=F0=9F=9C=?=? Date: Fri, 08 Dec 2023 04:20:33 +0000  
D: 37Sender: Team Anima <no-reply@animaapp.com> Subject: How is Anima going for you so far!Date: Fri, 08 Dec 2023 10:00:04 +0000  
D: 38Sender: LALALAI <noreply@lala.ai> Subject: New! Record HQ Audio & Video Enhanced with LALALAI Noise Removal!Date: Fri, 08 Dec 2023 15:19:32 +0000  
D: 39Sender: "Kahoot!" <kahoot@team.kahoot.com> Subject: =?utf-8?B?3VzIzEhbnB3aXRoIHRoG9vdCg8J+agA=?=? Date: Fri, 08 Dec 2023 15:19:32 +0000  
D: 40Sender: NVIDIA Accounts <account@nvidia.com> Subject: NVIDIA Account Created!Date: Sun, 10 Dec 2023 19:05:58 +0000  
D: 41Sender: "Kahoot!" <kahoot@team.kahoot.com> Subject: =?utf-8?B?3VzIzEhbnB3aXRoIHRoG9vdCg8J+agA=?=? Date: Thu, 14 Dec 2023 22:09:30 +0000  
D: 42Sender: "Kahoot!" <kahoot@team.kahoot.com> Subject: =?utf-8?B?3VzIzEhbnB3aXRoIHRoG9vdCg8J+agA=?=? Date: Thu, 14 Dec 2023 22:09:30 +0000  
D: 43Sender: "Kahoot!" <kahoot@team.kahoot.com> Subject: =?utf-8?B?3VzIzEhbnB3aXRoIHRoG9vdCg8J+agA=?=? Date: Wed, 20 Dec 2023 19:03:45 +0000  
D: 44Sender: "Kahoot!" <kahoot@team.kahoot.com> Subject: =?utf-8?B?3VzIzEhbnB3aXRoIHRoG9vdCg8J+agA=?=? Date: Thu, 21 Dec 2023 19:02:24 +0000

Dette er sånn det til slutt så ut når ferdig kodet.

FYLLES UT VED PROSJEKTETS SLUTT

Nådde jeg målet?



Det målet jeg satt for meg selv oppnådde jeg og litt til. Jeg tenkte kanskje på å gå litt lenger når jeg startet, men så trakk jeg meg selv tilbake igjen til noe jeg kan oppnå på tiden jeg har.

### Hva har jeg lært

Vel jeg har lært hvordan jeg bruker Tkinter og hva en «Label», «Button», «Listbox» og sånt er og gjør i programmet. Jeg har en grei forståelse av modulen, men jeg trenger fortsatt litt hjelp hvis jeg skal gjøre noen ting i det. Du kan også si at Tkinter er et verktøy på en måte som jeg har lært.

### Hva ville vært neste steg for deg om du skal lære mer om akkurat dette?

Egentlig så er det en del jeg kan gjøre videre hvis jeg hadde valgt dette på nytt igjen neste YFF periode. Den første er å kategorisere epostene. Den andre er å lage applikasjonen mer visuell og kanskje responsiv. Den siste er å lage noe mekanisme for å sende post til andre adresser, dette hadde vært veldig vanskelig tror jeg, men det er noe jeg kan få til.

### Hvilke ønsker har du for neste YFF-prosjekt?

Som noen andre YFF prosjektet så vet jeg ikke hva jeg vil jobbe med for det neste prosjektet. Jeg nevnte å videreutvikle applikasjonen siden nå er den teknisk sett bare en prototype.

