from tkinter import \*

import mysql.connector

from datetime import date, datetime

import tkinter.ttk

from tkinter import messagebox

def registrere\_student\_GUI():

def generer\_studentnr():

# Kode for å generere studentnr og student e-post:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

generer\_snr\_markor = mindatabase.cursor()

generer\_snr\_markor.execute('SELECT Studentnr FROM Student')

for row in generer\_snr\_markor:

maks = max(row)

nytt\_snr = int(maks) + 1

snr.set(nytt\_snr)

ep.set(str(nytt\_snr) + '@usn.no')

generer\_snr\_markor.close()

mindatabase.close()

def registrere\_student():

# Kode for å registrere student:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

registrer\_markor = mindatabase.cursor()

studentnr = int(snr.get())

fornavn = fnavn.get()

etternavn = enavn.get()

epost = ep.get()

telefon = tlf.get()

settinn\_student = ('INSERT INTO Student'

'(Studentnr, Fornavn, Etternavn, Epost, Telefon)'

'VALUES(%s, %s, %s, %s, %s)')

datany\_student = (studentnr, fornavn, etternavn, epost, telefon)

registrer\_markor.execute(settinn\_student, datany\_student)

mindatabase.commit()

messagebox.showinfo('Info', 'Studenten er registrert')

registrer\_markor.close()

mindatabase.close()

# GUI student:

registrere\_student\_vindu = Toplevel()

registrere\_student\_vindu.title('Student (registrering)')

registrere\_student\_vindu.resizable(FALSE, FALSE)

registrere\_student\_vindu.geometry('480x350+600+200')

overskrift = Label(registrere\_student\_vindu, text='Student registrering', font=('Sans-Serif', '22'))

overskrift.grid(row=0, columnspan=4, padx=15, pady=5)

forklaring\_snr = Label(registrere\_student\_vindu, text='Trykk på lag studentnr for å hente nytt (ubrukt) studentnr (løpende nr)', font=('Sans-Serif', '11'))

forklaring\_snr.grid(row=1, columnspan=4, padx=15, pady=(10, 30))

lbl\_studentnr = Label(registrere\_student\_vindu, text='Studentnr:')

lbl\_studentnr.grid(row=2, column=0, padx=5, pady=5, sticky=E)

lbl\_fornavn = Label(registrere\_student\_vindu, text='Fornavn:')

lbl\_fornavn.grid(row=3, column=0, padx=5, pady=5, sticky=E)

lbl\_etternavn = Label(registrere\_student\_vindu, text='Etternavn:')

lbl\_etternavn.grid(row=4, column=0, padx=5, pady=5, sticky=E)

lbl\_epost = Label(registrere\_student\_vindu, text='Epost:')

lbl\_epost.grid(row=5, column=0, padx=5, pady=5, sticky=E)

lbl\_telefon = Label(registrere\_student\_vindu, text='Mobil:')

lbl\_telefon.grid(row=6, column=0, padx=5, pady=5, sticky=E)

snr = StringVar()

ent\_snr = Entry(registrere\_student\_vindu, state='readonly', width=6, textvariable=snr)

ent\_snr.grid(row=2, column=1, padx=5, pady=5, sticky=W)

fnavn = StringVar()

ent\_fnavn = Entry(registrere\_student\_vindu, width=30, textvariable=fnavn)

ent\_fnavn.grid(row=3, column=1, padx=5, pady=5, sticky=W)

enavn = StringVar()

ent\_enavn = Entry(registrere\_student\_vindu, width=20, textvariable=enavn)

ent\_enavn.grid(row=4, column=1, padx=5, pady=5, sticky=W)

ep = StringVar()

ent\_ep = Entry(registrere\_student\_vindu, state='readonly', width=20, textvariable=ep)

ent\_ep.grid(row=5, column=1, padx=5, pady=5, sticky=W)

tlf = StringVar()

ent\_tlf = Entry(registrere\_student\_vindu, width=8, textvariable=tlf)

ent\_tlf.grid(row=6, column=1, padx=5, pady=5, sticky=W)

btn\_sok = Button(registrere\_student\_vindu, text='Lag studentnr', command=generer\_studentnr)

btn\_sok.grid(row=2, column=2, padx=5, pady=5, sticky=E)

btn\_lagre = Button(registrere\_student\_vindu, text='Registrer student', command=registrere\_student)

btn\_lagre.grid(row=7, column=2, padx=5, pady=5, sticky=E)

btn\_tilbake2 = Button(registrere\_student\_vindu, text='Tilbake til meny', command=registrere\_student\_vindu.destroy)

btn\_tilbake2.grid(row=7, column=0, padx=15, pady=25, sticky=W)

def oppdatere\_student\_GUI():

def sok\_student():

# Kode for å søke etter student:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

studentnr = snr.get()

sok\_markor = mindatabase.cursor()

sok\_markor.execute('SELECT \* FROM Student')

for row in sok\_markor:

if studentnr == row[0]:

fnavn.set(row[1])

enavn.set(row[2])

ep.set(row[3])

tlf.set(row[4])

sok\_markor.close()

mindatabase.close()

def oppdatere\_student():

# Kode for å oppdatere student:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

oppdatere\_markor = mindatabase.cursor()

studentnr = int(snr.get())

fornavn = fnavn.get()

etternavn = enavn.get()

epost = ep.get()

telefon = tlf.get()

oppdater\_student = ('Update Student SET Studentnr = %s, Fornavn = %s, Etternavn = %s, Epost = %s, Telefon = %s WHERE Studentnr = %s')

data\_ny\_student = (studentnr, fornavn, etternavn, epost, telefon, studentnr,)

oppdatere\_markor.execute(oppdater\_student, data\_ny\_student)

mindatabase.commit()

messagebox.showinfo('Info', 'Studenten er oppdatert')

oppdatere\_markor.close()

mindatabase.close()

# GUI student:

oppdatere\_student\_vindu = Toplevel()

oppdatere\_student\_vindu.title('Student (oppdatering)')

oppdatere\_student\_vindu.resizable(FALSE, FALSE)

oppdatere\_student\_vindu.geometry('470x350+600+200')

overskrift = Label(oppdatere\_student\_vindu, text='Student oppdatering', font=('Sans-Serif', '22'))

overskrift.grid(row=0, columnspan=2, padx=15, pady=5)

forklaring\_sok = Label(oppdatere\_student\_vindu, text='Du kan søke på student ved å skrive inn studentnr og trykke SØK', font=('Sans-Serif', '11'))

forklaring\_sok.grid(row=1, columnspan=2, padx=15, pady=(10, 30))

lbl\_studentnr = Label(oppdatere\_student\_vindu, text='Studentnr:')

lbl\_studentnr.grid(row=2, column=0, padx=5, pady=5, sticky=E)

lbl\_fornavn = Label(oppdatere\_student\_vindu, text='Fornavn:')

lbl\_fornavn.grid(row=3, column=0, padx=5, pady=5, sticky=E)

lbl\_etternavn = Label(oppdatere\_student\_vindu, text='Etternavn:')

lbl\_etternavn.grid(row=4, column=0, padx=5, pady=5, sticky=E)

lbl\_epost = Label(oppdatere\_student\_vindu, text='Epost:')

lbl\_epost.grid(row=5, column=0, padx=5, pady=5, sticky=E)

lbl\_telefon = Label(oppdatere\_student\_vindu, text='Mobil:')

lbl\_telefon.grid(row=6, column=0, padx=5, pady=5, sticky=E)

snr = StringVar()

ent\_snr = Entry(oppdatere\_student\_vindu, width=6, textvariable=snr)

ent\_snr.grid(row=2, column=1, padx=5, pady=5, sticky=W)

fnavn = StringVar()

ent\_fnavn = Entry(oppdatere\_student\_vindu, width=30, textvariable=fnavn)

ent\_fnavn.grid(row=3, column=1, padx=5, pady=5, sticky=W)

enavn = StringVar()

ent\_enavn = Entry(oppdatere\_student\_vindu, width=20, textvariable=enavn)

ent\_enavn.grid(row=4, column=1, padx=5, pady=5, sticky=W)

ep = StringVar()

ent\_ep = Entry(oppdatere\_student\_vindu, state='readonly', width=40, textvariable=ep)

ent\_ep.grid(row=5, column=1, padx=5, pady=5, sticky=W)

tlf = StringVar()

ent\_tlf = Entry(oppdatere\_student\_vindu, width=8, textvariable=tlf)

ent\_tlf.grid(row=6, column=1, padx=5, pady=5, sticky=W)

btn\_sok = Button(oppdatere\_student\_vindu, text='Søk', command=sok\_student)

btn\_sok.grid(row=2, column=1, padx=5, pady=5, sticky=E)

btn\_lagre = Button(oppdatere\_student\_vindu, text='Oppdater student', command=oppdatere\_student)

btn\_lagre.grid(row=7, column=1, padx=5, pady=5, sticky=E)

btn\_tilbake2 = Button(oppdatere\_student\_vindu, text='Tilbake til meny', command=oppdatere\_student\_vindu.destroy)

btn\_tilbake2.grid(row=7, column=0, padx=15, pady=25, sticky=W)

def slette\_student\_GUI():

def sok\_student():

# Kode for å søke etter student:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

studentnr = snr.get()

sok\_markor = mindatabase.cursor()

sok\_markor.execute('SELECT \* FROM Student')

for row in sok\_markor:

if studentnr == row[0]:

fnavn.set(row[1])

enavn.set(row[2])

ep.set(row[3])

tlf.set(row[4])

sok\_markor.close()

mindatabase.close()

def slette\_student():

# Kode for å slette student:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

slette\_markor = mindatabase.cursor()

studentnr = snr.get()

slett\_vare = ('DELETE FROM Student WHERE Studentnr = %s AND Studentnr NOT IN (SELECT Studentnr FROM Eksamensresultat)')

data\_ny\_student = (studentnr,)

slette\_markor.execute(slett\_vare, data\_ny\_student)

mindatabase.commit()

messagebox.showinfo('Info', 'Studenten er slettet')

slette\_markor.close()

mindatabase.close()

# GUI student:

slette\_student\_vindu = Toplevel()

slette\_student\_vindu.title('Student (sletting)')

slette\_student\_vindu.resizable(FALSE, FALSE)

slette\_student\_vindu.geometry('470x350+600+200')

overskrift = Label(slette\_student\_vindu, text='Student sletting', font=('Sans-Serif', '22'))

overskrift.grid(row=0, columnspan=3, padx=15, pady=5)

forklaring\_sok = Label(slette\_student\_vindu, text='Du kan søke på student ved å skrive inn studentnr og trykke SØK', font=('Sans-Serif', '11'))

forklaring\_sok.grid(row=1, columnspan=2, padx=15, pady=(10, 30))

lbl\_studentnr = Label(slette\_student\_vindu, text='Studentnr:')

lbl\_studentnr.grid(row=2, column=0, padx=5, pady=5, sticky=E)

lbl\_fornavn = Label(slette\_student\_vindu, text='Fornavn:')

lbl\_fornavn.grid(row=3, column=0, padx=5, pady=5, sticky=E)

lbl\_etternavn = Label(slette\_student\_vindu, text='Etternavn:')

lbl\_etternavn.grid(row=4, column=0, padx=5, pady=5, sticky=E)

lbl\_epost = Label(slette\_student\_vindu, text='Epost:')

lbl\_epost.grid(row=5, column=0, padx=5, pady=5, sticky=E)

lbl\_telefon = Label(slette\_student\_vindu, text='Mobil:')

lbl\_telefon.grid(row=6, column=0, padx=5, pady=5, sticky=E)

snr = StringVar()

ent\_snr = Entry(slette\_student\_vindu, width=6, textvariable=snr)

ent\_snr.grid(row=2, column=1, padx=5, pady=5, sticky=W)

fnavn = StringVar()

ent\_fnavn = Entry(slette\_student\_vindu, state='readonly', width=30, textvariable=fnavn)

ent\_fnavn.grid(row=3, column=1, padx=5, pady=5, sticky=W)

enavn = StringVar()

ent\_enavn = Entry(slette\_student\_vindu, state='readonly', width=20, textvariable=enavn)

ent\_enavn.grid(row=4, column=1, padx=5, pady=5, sticky=W)

ep = StringVar()

ent\_ep = Entry(slette\_student\_vindu, state='readonly', width=40, textvariable=ep)

ent\_ep.grid(row=5, column=1, padx=5, pady=5, sticky=W)

tlf = StringVar()

ent\_tlf = Entry(slette\_student\_vindu, state='readonly', width=8, textvariable=tlf)

ent\_tlf.grid(row=6, column=1, padx=5, pady=5, sticky=W)

btn\_sok = Button(slette\_student\_vindu, text='Søk', command=sok\_student)

btn\_sok.grid(row=2, column=1, padx=5, pady=5, sticky=E)

btn\_lagre = Button(slette\_student\_vindu, text='Slette student', command=slette\_student)

btn\_lagre.grid(row=7, column=1, padx=5, pady=5, sticky=E)

btn\_tilbake2 = Button(slette\_student\_vindu, text='Tilbake til meny', command=slette\_student\_vindu.destroy)

btn\_tilbake2.grid(row=7, column=0, padx=15, pady=25, sticky=W)

def utskrift\_eksamensresultater\_student\_GUI():

def utskrift\_alle\_resultater():

# Kode for utskrift alle resultater:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

studentnr = snr.get()

utskrift\_markor = mindatabase.cursor()

utskrift\_markor.execute('SELECT Eksamensresultat.\*, Emnenavn, Studiepoeng FROM Eksamensresultat, Emne WHERE Eksamensresultat.Emnekode = Emne.Emnekode ORDER BY Dato ASC')

data = utskrift\_markor.fetchall()

tekst\_vindu.delete('1.0', END)

totalt\_snr = 0

for row in data:

if studentnr == row[0] and row[3] != None:

tekst\_vindu.insert(END, "Snr:" + str(row[0]) +" | " + "Emnek: " + str(row[1]) + " | " + "Dato: " + str(row[2]) + " | " + "Karakter: " + str(row[3]) + "\n" + "Emnenavn: " + str(row[4]) + " | " + "Studiepoeng: " + str(row[5]) + "\n" +"\n")

totalt\_snr += row[5]

totalt.set(totalt\_snr)

utskrift\_markor.close()

mindatabase.close()

def utskrift\_vitnemal():

# Kode for utskrift av vitnemål:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

studentnr = snr.get()

utskrift\_markor = mindatabase.cursor()

utskrift\_markor.execute('SELECT Eksamensresultat.\*, Emnenavn, Studiepoeng FROM Eksamensresultat, Emne WHERE Eksamensresultat.Emnekode = Emne.Emnekode AND Eksamensresultat.Karakter IS NOT NULL ORDER BY RIGHT (Eksamensresultat.Emnekode, 4) ASC')

data = utskrift\_markor.fetchall()

tekst\_vindu.delete('1.0', END)

totalt\_snr = 0

for row in data:

if studentnr == row[0]:

tekst\_vindu.insert(END, "Snr:" + str(row[0]) +" | " + "Emnek: " + str(row[1]) + " | " + "Dato: " + str(row[2]) + " | " + "Karakter: " + str(row[3]) + "\n" + "Emnenavn: " + str(row[4]) + " | " + "Studiepoeng: " + str(row[5]) + "\n" +"\n")

totalt\_snr += row[5]

totalt.set(totalt\_snr)

utskrift\_markor.close()

mindatabase.close()

# GUI utskrift student:

utskrift\_alle\_resultater\_vindu = Toplevel()

utskrift\_alle\_resultater\_vindu.title('Student (utskrift)')

utskrift\_alle\_resultater\_vindu.resizable(FALSE, FALSE)

utskrift\_alle\_resultater\_vindu.geometry('535x470+600+100')

overskrift = Label(utskrift\_alle\_resultater\_vindu, text='Eksamensresultater / Vitnemål', font=('Sans-Serif', '22'))

overskrift.grid(row=0, columnspan=7, padx=15, pady=5)

lbl\_studentnr = Label(utskrift\_alle\_resultater\_vindu, text='Studentnr:')

lbl\_studentnr.grid(row=1, column=0, padx=5, pady=5, sticky=E)

lbl\_total\_student\_poeng = Label(utskrift\_alle\_resultater\_vindu, text='Studentpoeng (totalt):')

lbl\_total\_student\_poeng.grid(row=1, column=3, padx=5, pady=5, sticky=E)

snr = StringVar()

ent\_snr = Entry(utskrift\_alle\_resultater\_vindu, width=6, textvariable=snr)

ent\_snr.grid(row=1, column=1, padx=5, pady=5, sticky=W)

totalt = StringVar()

ent\_totalt = Entry(utskrift\_alle\_resultater\_vindu, state='readonly', width=5, textvariable=totalt)

ent\_totalt.grid(row=1, column=4, padx=5, pady=5, sticky=W)

tekst\_vindu = Text(utskrift\_alle\_resultater\_vindu, width=65, height=20)

tekst\_vindu.grid(row=2, columnspan=5, padx=5, pady=5)

btn\_hent\_data1 = Button(utskrift\_alle\_resultater\_vindu, text='Hent alle resultater', command=utskrift\_alle\_resultater)

btn\_hent\_data1.grid(row=4, column=3, padx=15, pady=5, sticky=SE)

btn\_hent\_data2 = Button(utskrift\_alle\_resultater\_vindu, text='Hent vitnemål', command=utskrift\_vitnemal)

btn\_hent\_data2.grid(row=4, column=4, padx=15, pady=5, sticky=SE)

btn\_tilbake2 = Button(utskrift\_alle\_resultater\_vindu, text='Tilbake til meny', command=utskrift\_alle\_resultater\_vindu.destroy)

btn\_tilbake2.grid(row=4, column=0, padx=15, pady=5, sticky=SW)

def registrere\_eksamen\_GUI():

def sok\_eksamen():

# Kode for å søke etter eksamen:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

dato = date.get()

sok\_markor = mindatabase.cursor()

sok\_markor.execute('SELECT \* FROM Eksamen')

data = sok\_markor.fetchall()

lst\_eksamen.delete('1.0', END)

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

for row in data:

if dato\_konvertert == row[1]:

lst\_eksamen.insert(END, "Romnr: " + str(row[2]) + " | OPPTATT" + '\n')

sok\_markor.close()

mindatabase.close()

def registrere\_eksamen():

# Kode for å registrere eksamen:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

emnekode = ek.get()

dato = date.get()

romnr = romn.get()

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

registrer\_eksamen\_markor = mindatabase.cursor()

sql = 'SELECT Eksamen.Dato,Eksamen.Romnr FROM Eksamen WHERE Eksamen.Dato = %s AND Eksamen.Romnr = %s'

data\_ny\_eksamen = (dato\_konvertert, romnr,)

registrer\_eksamen\_markor.execute(sql, data\_ny\_eksamen)

data = registrer\_eksamen\_markor.fetchall()

if not data:

settinn\_eksamen = ('INSERT INTO Eksamen'

'(Emnekode, Dato, Romnr)'

'VALUES(%s, %s, %s)')

datany\_eksamen = (emnekode, dato\_konvertert, romnr)

registrer\_eksamen\_markor.execute(settinn\_eksamen, datany\_eksamen)

mindatabase.commit()

messagebox.showinfo('Info', 'Eksamen er registrert')

if data:

messagebox.showinfo('Info', 'Rommet er opptatt')

registrer\_eksamen\_markor.close()

mindatabase.close()

# GUI registrere eksamen:

registrere\_eksamen\_vindu = Toplevel()

registrere\_eksamen\_vindu.title('Eksamen (registrering)')

registrere\_eksamen\_vindu.resizable(FALSE, FALSE)

registrere\_eksamen\_vindu.geometry('580x300+600+200')

overskrift = Label(registrere\_eksamen\_vindu, text='Eksamen registrering', font=('Sans-Serif', '22'))

overskrift.grid(row=0, columnspan=5, padx=15, pady=5)

forklaring\_sok = Label(registrere\_eksamen\_vindu, text='Du kan søke på dato for å se hvilke rom som er opptatt på ønsket dato', font=('Sans-Serif', '11'))

forklaring\_sok.grid(row=1, columnspan=5, padx=15, pady=(10, 30))

lst\_eksamen = Text(registrere\_eksamen\_vindu, width=30, height=5)

lst\_eksamen.grid(row=2, column=2, rowspan=3, padx=5, pady=5, sticky=W)

lbl\_emndekode = Label(registrere\_eksamen\_vindu, text='Emnekode:')

lbl\_emndekode.grid(row=2, column=0, padx=5, pady=5, sticky=E)

lbl\_dato = Label(registrere\_eksamen\_vindu, text='Dato (yyyy-mm-dd):')

lbl\_dato.grid(row=3, column=0, padx=5, pady=5, sticky=E)

lbl\_romnr = Label(registrere\_eksamen\_vindu, text='Romnr:')

lbl\_romnr.grid(row=4, column=0, padx=5, pady=5, sticky=E)

ek = StringVar()

ent\_ek = Entry(registrere\_eksamen\_vindu, width=8, textvariable=ek)

ent\_ek.grid(row=2, column=1, padx=5, pady=5, sticky=W)

date = StringVar()

ent\_date = Entry(registrere\_eksamen\_vindu, width=10, textvariable=date)

ent\_date.grid(row=3, column=1, padx=5, pady=5, sticky=W)

romn = StringVar()

ent\_romn = Entry(registrere\_eksamen\_vindu, width=4, textvariable=romn)

ent\_romn.grid(row=4, column=1, padx=5, pady=5, sticky=W)

btn\_sok = Button(registrere\_eksamen\_vindu, text='Søk', command=sok\_eksamen)

btn\_sok.grid(row=2, column=4, padx=5, pady=5, sticky=E)

btn\_lagre = Button(registrere\_eksamen\_vindu, text='Registrer eksamen', command=registrere\_eksamen)

btn\_lagre.grid(row=7, column=4, padx=5, pady=25, sticky=E)

btn\_tilbake2 = Button(registrere\_eksamen\_vindu, text='Tilbake til meny', command=registrere\_eksamen\_vindu.destroy)

btn\_tilbake2.grid(row=7, column=0, padx=15, pady=25, sticky=W)

def oppdatere\_eksamen\_GUI():

def sok\_eksamen():

# Kode for å søke etter eksamen:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

emnekode = ek.get()

dato = date.get()

sok\_markor = mindatabase.cursor()

sok\_markor.execute('SELECT \* FROM Eksamen')

data = sok\_markor.fetchall()

if len(emnekode) != 0:

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

for row in data:

if emnekode == row[0] and dato\_konvertert == row[1]:

ek.set(row[0])

date.set(row[1])

romn.set(row[2])

sok\_markor.close()

mindatabase.close()

def oppdatere\_eksamen():

# Kode for å oppdatere eksamen:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

oppdatere\_markor = mindatabase.cursor()

emnekode = ek.get()

dato = date.get()

romnr = romn.get()

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

registrer\_eksamen\_markor = mindatabase.cursor()

sql = 'SELECT Eksamen.Dato,Eksamen.Romnr FROM Eksamen WHERE Eksamen.Dato = %s AND Eksamen.Romnr = %s'

data\_ny\_eksamen = (dato\_konvertert, romnr,)

registrer\_eksamen\_markor.execute(sql, data\_ny\_eksamen)

data = registrer\_eksamen\_markor.fetchall()

if not data:

oppdater\_eksamen = ('UPDATE Eksamen SET Emnekode = %s, Dato = %s, Romnr = %s WHERE Emnekode = %s AND Dato = %s')

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

data\_ny\_eksamen = (emnekode, dato\_konvertert, romnr, emnekode, dato\_konvertert,)

oppdatere\_markor.execute(oppdater\_eksamen, data\_ny\_eksamen)

mindatabase.commit()

messagebox.showinfo('Info', 'Eksamen er oppdatert')

if data:

messagebox.showinfo('Info', 'Rommet er opptatt')

oppdatere\_markor.close()

mindatabase.close()

# GUI oppdatere eksamen:

oppdatere\_eksamen\_vindu = Toplevel()

oppdatere\_eksamen\_vindu.title('Student (oppdatering)')

oppdatere\_eksamen\_vindu.resizable(FALSE, FALSE)

oppdatere\_eksamen\_vindu.geometry('540x280+600+200')

overskrift = Label(oppdatere\_eksamen\_vindu, text='Eksamens oppdatering', font=('Sans-Serif', '22'))

overskrift.grid(row=0, columnspan=3, padx=15, pady=5)

forklaring\_sok = Label(oppdatere\_eksamen\_vindu, text='Du kan søke på eksamen ved å skrive inn emnekode og dato og trykke SØK', font=('Sans-Serif', '11'))

forklaring\_sok.grid(row=1, columnspan=3, padx=15, pady=(10, 30))

lbl\_emnekode = Label(oppdatere\_eksamen\_vindu, text='Emnekode:')

lbl\_emnekode.grid(row=2, column=0, padx=5, pady=5, sticky=E)

lbl\_dato = Label(oppdatere\_eksamen\_vindu, text='Dato (yyyy-mm-dd):')

lbl\_dato.grid(row=3, column=0, padx=5, pady=5, sticky=E)

lbl\_romnr = Label(oppdatere\_eksamen\_vindu, text='Romnr:')

lbl\_romnr.grid(row=4, column=0, padx=5, pady=5, sticky=E)

ek = StringVar()

ent\_ek = Entry(oppdatere\_eksamen\_vindu, width=8, textvariable=ek)

ent\_ek.grid(row=2, column=1, padx=5, pady=5, sticky=W)

date = StringVar()

ent\_date = Entry(oppdatere\_eksamen\_vindu, width=10, textvariable=date)

ent\_date.grid(row=3, column=1, padx=5, pady=5, sticky=W)

romn = StringVar()

ent\_romn = Entry(oppdatere\_eksamen\_vindu, width=4, textvariable=romn)

ent\_romn.grid(row=4, column=1, padx=5, pady=5, sticky=W)

btn\_sok = Button(oppdatere\_eksamen\_vindu, text='Søk', command=sok\_eksamen)

btn\_sok.grid(row=2, column=2, padx=5, pady=5, sticky=E)

btn\_lagre = Button(oppdatere\_eksamen\_vindu, text='Oppdater eksamen', command=oppdatere\_eksamen)

btn\_lagre.grid(row=5, column=2, padx=5, pady=5, sticky=E)

btn\_tilbake2 = Button(oppdatere\_eksamen\_vindu, text='Tilbake til meny', command=oppdatere\_eksamen\_vindu.destroy)

btn\_tilbake2.grid(row=5, column=0, padx=15, pady=25, sticky=W)

def slette\_eksamen\_GUI():

def sok\_eksamen():

# Kode for å søke etter eksamen:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

emnekode = ek.get()

dato = date.get()

sok\_markor = mindatabase.cursor()

sok\_markor.execute('SELECT \* FROM Eksamen')

data = sok\_markor.fetchall()

if len(emnekode) != 0:

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

for row in data:

if emnekode == row[0] and dato\_konvertert == row[1]:

ek.set(row[0])

date.set(row[1])

romn.set(row[2])

sok\_markor.close()

mindatabase.close()

def slett\_eksamen():

# Kode for å slette eksamen:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

slette\_markor = mindatabase.cursor()

emnekode = ek.get()

dato = date.get()

slett\_vare = ('DELETE FROM Eksamen WHERE Emnekode = %s AND Dato = %s AND Emnekode NOT IN (SELECT Emnekode FROM Eksamensresultat)')

data\_ny\_student = (emnekode, dato,)

slette\_markor.execute(slett\_vare, data\_ny\_student)

mindatabase.commit()

messagebox.showinfo('Info', 'Eksamen er slettet')

slette\_markor.close()

mindatabase.close()

# GUI slette eksamen:

oppdatere\_eksamen\_vindu = Toplevel()

oppdatere\_eksamen\_vindu.title('Student (oppdatering)')

oppdatere\_eksamen\_vindu.resizable(FALSE, FALSE)

oppdatere\_eksamen\_vindu.geometry('540x310+600+200')

overskrift = Label(oppdatere\_eksamen\_vindu, text='Eksamens sletting', font=('Sans-Serif', '22'))

overskrift.grid(row=0, columnspan=3, padx=15, pady=5)

forklaring\_sok = Label(oppdatere\_eksamen\_vindu, text='Du kan søke på eksamen ved å skrive inn emnekode og dato og trykke SØK', font=('Sans-Serif', '11'))

forklaring\_sok.grid(row=1, columnspan=3, padx=15, pady=5)

forklaring\_sok2 = Label(oppdatere\_eksamen\_vindu, text='Eksamen blir kun slettet hvis det ikke er noen resultater registrert', font=('Sans-Serif', '11'))

forklaring\_sok2.grid(row=2, columnspan=3, padx=15, pady=(5, 30))

lbl\_emnekode = Label(oppdatere\_eksamen\_vindu, text='Emnekode:')

lbl\_emnekode.grid(row=3, column=0, padx=5, pady=5, sticky=E)

lbl\_dato = Label(oppdatere\_eksamen\_vindu, text='Dato (yyyy-mm-dd):')

lbl\_dato.grid(row=4, column=0, padx=5, pady=5, sticky=E)

lbl\_romnr = Label(oppdatere\_eksamen\_vindu, text='Romnr:')

lbl\_romnr.grid(row=5, column=0, padx=5, pady=5, sticky=E)

ek = StringVar()

ent\_ek = Entry(oppdatere\_eksamen\_vindu, width=8, textvariable=ek)

ent\_ek.grid(row=3, column=1, padx=5, pady=5, sticky=W)

date = StringVar()

ent\_date = Entry(oppdatere\_eksamen\_vindu, width=10, textvariable=date)

ent\_date.grid(row=4, column=1, padx=5, pady=5, sticky=W)

romn = StringVar()

ent\_romn = Entry(oppdatere\_eksamen\_vindu, width=4, textvariable=romn)

ent\_romn.grid(row=5, column=1, padx=5, pady=5, sticky=W)

btn\_sok = Button(oppdatere\_eksamen\_vindu, text='Søk', command=sok\_eksamen)

btn\_sok.grid(row=3, column=2, padx=5, pady=5, sticky=E)

btn\_lagre = Button(oppdatere\_eksamen\_vindu, text='Slett eksamen', command=slett\_eksamen)

btn\_lagre.grid(row=6, column=2, padx=5, pady=5, sticky=E)

btn\_tilbake2 = Button(oppdatere\_eksamen\_vindu, text='Tilbake til meny', command=oppdatere\_eksamen\_vindu.destroy)

btn\_tilbake2.grid(row=6, column=0, padx=15, pady=25, sticky=W)

def utskrift\_eksamen\_GUI():

def alle\_eksamen\_en\_dag():

# Kode for utskrift av alle eksamner på en dag:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

dato\_fra = date\_from.get()

utskrift\_markor = mindatabase.cursor()

utskrift\_markor.execute('SELECT \* FROM Eksamen')

dato\_konvertert = datetime.strptime(dato\_fra, '%Y-%m-%d').date()

data = utskrift\_markor.fetchall()

tekst\_vindu.delete('1.0', END)

for row in data:

if dato\_konvertert == row[1]:

tekst\_vindu.insert(END, "Emnekode: " + str(row[0]) + " | " + "Rom: " + str(row[2]) + "\n")

utskrift\_markor.close()

mindatabase.close()

def alle\_eksamen\_periode():

# Kode for utskrift av alle eksamener innen for en periode, sortert etter dato:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

dato\_fra = date\_from.get()

dato\_til = date\_to.get()

dato\_konvertert = datetime.strptime(dato\_fra, '%Y-%m-%d').date()

dato\_konvertert2 = datetime.strptime(dato\_til, '%Y-%m-%d').date()

utskrift\_markor = mindatabase.cursor()

sql = 'SELECT \* FROM Eksamen WHERE Dato >= %s AND Dato <= %s ORDER BY Dato'

data\_dato = (dato\_konvertert, dato\_konvertert2,)

utskrift\_markor.execute(sql, data\_dato)

data = utskrift\_markor.fetchall()

tekst\_vindu.delete('1.0', END)

for row in data:

tekst\_vindu.insert(END, "Emnekode: " + str(row[0]) + " | " + "Dato: " + str(row[1]) + " | " + "Rom: " + str(row[2]) + "\n")

# GUI utskrift eksamen:

utskrift\_eksamen\_vindu = Toplevel()

utskrift\_eksamen\_vindu.title('Eksamen (utskrift)')

utskrift\_eksamen\_vindu.resizable(FALSE, FALSE)

utskrift\_eksamen\_vindu.geometry('535x530+600+100')

overskrift = Label(utskrift\_eksamen\_vindu, text='Eksamens utskrift', font=('Sans-Serif', '22'))

overskrift.grid(row=0, columnspan=7, padx=15, pady=5)

forklaring\_utskrift = Label(utskrift\_eksamen\_vindu, text='Hvis du ønsker å søke på kun en dag så fyll inn ønsket dato i dato (fra)', font=('Sans-Serif', '11'))

forklaring\_utskrift.grid(row=1, columnspan=7, padx=15, pady=(10, 30))

lbl\_studentnr = Label(utskrift\_eksamen\_vindu, text='Dato (yyyy-mm-dd):')

lbl\_studentnr.grid(row=2, column=0, padx=5, pady=5, sticky=E)

lbl\_total\_student\_poeng = Label(utskrift\_eksamen\_vindu, text='Dato (yyyy-mm-dd):')

lbl\_total\_student\_poeng.grid(row=2, column=3, padx=5, pady=5, sticky=E)

date\_from = StringVar()

ent\_date = Entry(utskrift\_eksamen\_vindu, width=10, textvariable=date\_from)

ent\_date.grid(row=2, column=1, padx=5, pady=5, sticky=W)

date\_to = StringVar()

ent\_totalt = Entry(utskrift\_eksamen\_vindu, width=10, textvariable=date\_to)

ent\_totalt.grid(row=2, column=4, padx=5, pady=5, sticky=W)

tekst\_vindu = Text(utskrift\_eksamen\_vindu, width=65, height=20)

tekst\_vindu.grid(row=3, columnspan=5, padx=5, pady=5)

btn\_hent\_data1 = Button(utskrift\_eksamen\_vindu, text='Hent data (1 dag)', command=alle\_eksamen\_en\_dag)

btn\_hent\_data1.grid(row=4, column=3, padx=15, pady=5, sticky=SE)

btn\_hent\_data2 = Button(utskrift\_eksamen\_vindu, text='Hent data (periode)', command=alle\_eksamen\_periode)

btn\_hent\_data2.grid(row=4, column=4, padx=15, pady=5, sticky=SE)

btn\_tilbake2 = Button(utskrift\_eksamen\_vindu, text='Tilbake til meny', command=utskrift\_eksamen\_vindu.destroy)

btn\_tilbake2.grid(row=4, column=0, padx=15, pady=5, sticky=SW)

def registrere\_eksamensresultat\_GUI():

def sok\_en\_eksamen():

# Søke etter oppsatt eksamen:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

emnekode = sok\_ek.get()

dato = sok\_date.get()

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

markor = mindatabase.cursor()

markor.execute('SELECT \* FROM Eksamen')

data = markor.fetchall()

for row in data:

if emnekode == row[0] and dato\_konvertert == row[1]:

ek.set(row[0])

date.set(row[1])

markor.close()

mindatabase.close()

def registrer\_en\_karakter():

# Registrere en enkel karakter:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

studentnr = snr.get()

emnekode = ek.get()

dato = date.get()

karakter = grade.get()

if karakter == '':

karakter = None

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

markor = mindatabase.cursor()

sett\_inn\_karakter = ('INSERT INTO Eksamensresultat'

'(Studentnr, Emnekode, Dato, Karakter)'

'VALUES(%s, %s, %s, %s)')

data\_ny\_eksamen = (studentnr, emnekode, dato\_konvertert, karakter)

markor.execute(sett\_inn\_karakter, data\_ny\_eksamen)

mindatabase.commit()

messagebox.showinfo('Info', 'Eksamenskarakteren er registrert')

markor.close()

mindatabase.close()

def sok\_eksamensresultat():

# Hente frem alle studenter for å kunne masse registrere eksamensresultat:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

markor = mindatabase.cursor()

markor.execute('SELECT \* FROM Student')

data = markor.fetchall()

eksamen\_pa\_dato = []

for row in data:

eksamen\_pa\_dato += [row[0]]

innhold\_eksamensresultat.set(tuple(eksamen\_pa\_dato))

markor.close()

mindatabase.close()

def sok\_en\_eksamen2():

# Hente frem ønsket eksamen for å kunne masse registrere eksamensresultat:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

emnekode = sok\_ek2.get()

dato = sok\_date2.get()

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

markor = mindatabase.cursor()

markor.execute('SELECT \* FROM Eksamen')

data = markor.fetchall()

for row in data:

if emnekode == row[0] and dato\_konvertert == row[1]:

ek2.set(row[0])

date2.set(row[1])

markor.close()

mindatabase.close()

def hente\_student\_eksamensresultat(handling):

# Koble resultatet fra student listen med set'er til masseregistrering for å gjøre det enklere å masseregistrere:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

valgt = liste\_eksamen.get(liste\_eksamen.curselection())

markor = mindatabase.cursor()

markor.execute('SELECT \* FROM Student')

data = markor.fetchall()

for row in data:

if valgt == row[0]:

snr2.set(row[0])

markor.close()

mindatabase.close()

def registrer\_eksamensresultat\_mengde():

# Kode for å registrere hver enkelt student under masseregistrering:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

studentnr = snr2.get()

emnekode = ek2.get()

dato = date2.get()

karakter = grade3.get()

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

markor = mindatabase.cursor()

sett\_inn\_karakter = ('INSERT INTO Eksamensresultat'

'(Studentnr, Emnekode, Dato, Karakter)'

'VALUES(%s, %s, %s, %s)')

data\_ny\_eksamen = (studentnr, emnekode, dato\_konvertert, karakter)

markor.execute(sett\_inn\_karakter, data\_ny\_eksamen)

mindatabase.commit()

messagebox.showinfo('Info', 'Eksamenskarakteren er registrert')

markor.close()

mindatabase.close()

# GUI registrere eksamensresultat:

registrer\_eksamensresultat\_vindu = Toplevel()

registrer\_eksamensresultat\_vindu.title('Eksamensresultat (registrering)')

registrer\_eksamensresultat\_vindu.resizable(FALSE, FALSE)

registrer\_eksamensresultat\_vindu.geometry('478x755+600+10')

overskrift = Label(registrer\_eksamensresultat\_vindu, text='Eksamensresultat registrering', font=('Sans-Serif', '22'))

overskrift.grid(row=0, columnspan=5, padx=15, pady=5)

forklaring\_sok = Label(registrer\_eksamensresultat\_vindu, text='Skriv inn emnekode og dato for å søke på eksamen', font=('Sans-Serif', '11'))

forklaring\_sok.grid(row=1, columnspan=5, padx=15, pady=(10, 30))

btn\_sok = Button(registrer\_eksamensresultat\_vindu, text='Søk eksamen', command=sok\_en\_eksamen)

btn\_sok.grid(row=2, column=4, padx=5, pady=5, sticky=E)

lbl\_sok\_emnekode = Label(registrer\_eksamensresultat\_vindu, text='Emnekode:')

lbl\_sok\_emnekode.grid(row=2, column=0, padx=5, pady=5, sticky=E)

sok\_ek = StringVar()

ent\_ek = Entry(registrer\_eksamensresultat\_vindu, width=8, textvariable=sok\_ek)

ent\_ek.grid(row=2, column=1, padx=5, pady=5, sticky=W)

lbl\_sok\_dato = Label(registrer\_eksamensresultat\_vindu, text='Dato (yyyy-mm-dd):')

lbl\_sok\_dato.grid(row=3, column=0, padx=5, pady=5, sticky=E)

sok\_date = StringVar()

ent\_sok\_date = Entry(registrer\_eksamensresultat\_vindu, width=10, textvariable=sok\_date)

ent\_sok\_date.grid(row=3, column=1, padx=5, pady=5, sticky=W)

tkinter.ttk.Separator(registrer\_eksamensresultat\_vindu, orient=HORIZONTAL).grid(row=4, columnspan=5, pady=10, sticky='WE')

lbl\_studentnr = Label(registrer\_eksamensresultat\_vindu, text='Studentnr:')

lbl\_studentnr.grid(row=5, column=0, padx=5, pady=5, sticky=E)

snr = StringVar()

ent\_snr = Entry(registrer\_eksamensresultat\_vindu, width=6, textvariable=snr)

ent\_snr.grid(row=5, column=1, padx=5, pady=5, sticky=W)

lbl\_emnekode = Label(registrer\_eksamensresultat\_vindu, text='Emnekode:')

lbl\_emnekode.grid(row=6, column=0, padx=5, pady=5, sticky=E)

ek = StringVar()

ent\_ek = Entry(registrer\_eksamensresultat\_vindu, state='readonly', width=8, textvariable=ek)

ent\_ek.grid(row=6, column=1, padx=5, pady=5, sticky=W)

lbl\_dato = Label(registrer\_eksamensresultat\_vindu, text='Dato (yyyy-mm-dd):')

lbl\_dato.grid(row=7, column=0, padx=5, pady=5, sticky=E)

date = StringVar()

ent\_date = Entry(registrer\_eksamensresultat\_vindu, state='readonly', width=10, textvariable=date)

ent\_date.grid(row=7, column=1, padx=5, pady=5, sticky=W)

lbl\_karakter = Label(registrer\_eksamensresultat\_vindu, text='Karakter:')

lbl\_karakter.grid(row=8, column=0, padx=5, pady=5, sticky=E)

grade = StringVar()

ent\_grade = Entry(registrer\_eksamensresultat\_vindu, width=6, textvariable=grade)

ent\_grade.grid(row=8, column=1, padx=5, pady=5, sticky=W)

btn\_registrer\_eksamen = Button(registrer\_eksamensresultat\_vindu, text='Registrer eksamensresultat', command=registrer\_en\_karakter)

btn\_registrer\_eksamen.grid(row=9, column=4, padx=5, pady=5, sticky=E)

tkinter.ttk.Separator(registrer\_eksamensresultat\_vindu, orient=HORIZONTAL).grid(row=10, columnspan=5, pady=10, sticky='WE')

forklaring\_sok2 = Label(registrer\_eksamensresultat\_vindu, text='Skriv inn emnekode og dato for å hente eksamens info', font=('Sans-Serif', '11'))

forklaring\_sok2.grid(row=11, columnspan=5, padx=15, pady=5)

forklaring\_sok3 = Label(registrer\_eksamensresultat\_vindu, text='Trykk på hent studenter for å hente studentnr', font=('Sans-Serif', '11'))

forklaring\_sok3.grid(row=12, columnspan=5, padx=15, pady=(0, 30))

btn\_sok2 = Button(registrer\_eksamensresultat\_vindu, text='Søk eksamen', command=sok\_en\_eksamen2)

btn\_sok2.grid(row=13, column=4, padx=5, pady=5, sticky=E)

lbl\_sok\_emnekode2 = Label(registrer\_eksamensresultat\_vindu, text='Emnekode:')

lbl\_sok\_emnekode2.grid(row=13, column=0, padx=5, pady=5, sticky=E)

sok\_ek2 = StringVar()

ent\_ek2 = Entry(registrer\_eksamensresultat\_vindu, width=8, textvariable=sok\_ek2)

ent\_ek2.grid(row=13, column=1, padx=5, pady=5, sticky=W)

lbl\_sok\_dato2 = Label(registrer\_eksamensresultat\_vindu, text='Dato (yyyy-mm-dd):')

lbl\_sok\_dato2.grid(row=14, column=0, padx=5, pady=5, sticky=E)

sok\_date2 = StringVar()

ent\_sok\_date2 = Entry(registrer\_eksamensresultat\_vindu, width=10, textvariable=sok\_date2)

ent\_sok\_date2.grid(row=14, column=1, padx=5, pady=5, sticky=W)

tkinter.ttk.Separator(registrer\_eksamensresultat\_vindu, orient=HORIZONTAL).grid(row=15, columnspan=5, pady=10, sticky='WE')

btn\_sok3 = Button(registrer\_eksamensresultat\_vindu, text='Hent studenter', command=sok\_eksamensresultat)

btn\_sok3.grid(row=16, column=4, padx=5, pady=5, sticky=E)

lbl\_studentnr2 = Label(registrer\_eksamensresultat\_vindu, text='Studentnr:')

lbl\_studentnr2.grid(row=16, column=0, padx=5, pady=5, sticky=E)

snr2 = StringVar()

ent\_snr2 = Entry(registrer\_eksamensresultat\_vindu, width=6, textvariable=snr2)

ent\_snr2.grid(row=16, column=1, padx=5, pady=5, sticky=W)

y\_scroll = Scrollbar(registrer\_eksamensresultat\_vindu, orient=VERTICAL)

y\_scroll.grid(row=16, rowspan=4, column=2, padx=(100, 0), pady=5, sticky=NS)

innhold\_eksamensresultat = StringVar()

liste\_eksamen = Listbox(registrer\_eksamensresultat\_vindu, width=15, height=7, listvariable=innhold\_eksamensresultat, yscrollcommand=y\_scroll.set)

liste\_eksamen.grid(row=16, rowspan=4, column=2, padx=5, pady=5)

y\_scroll['command'] = liste\_eksamen.yview

lbl\_emnekode2 = Label(registrer\_eksamensresultat\_vindu, text='Emnekode:')

lbl\_emnekode2.grid(row=17, column=0, padx=5, pady=5, sticky=E)

ek2 = StringVar()

ent\_ek2 = Entry(registrer\_eksamensresultat\_vindu, width=8, textvariable=ek2)

ent\_ek2.grid(row=17, column=1, padx=5, pady=5, sticky=W)

lbl\_dato2 = Label(registrer\_eksamensresultat\_vindu, text='Dato (yyyy-mm-dd):')

lbl\_dato2.grid(row=18, column=0, padx=5, pady=5, sticky=E)

date2 = StringVar()

ent\_date2 = Entry(registrer\_eksamensresultat\_vindu, width=10, textvariable=date2)

ent\_date2.grid(row=18, column=1, padx=5, pady=5, sticky=W)

lbl\_karakter2 = Label(registrer\_eksamensresultat\_vindu, text='Karakter:')

lbl\_karakter2.grid(row=19, column=0, padx=5, pady=5, sticky=E)

grade3 = StringVar()

ent\_grade = Entry(registrer\_eksamensresultat\_vindu, width=6, textvariable=grade3)

ent\_grade.grid(row=19, column=1, padx=5, pady=5, sticky=W)

btn\_slett2 = Button(registrer\_eksamensresultat\_vindu, text='Registrer eksamensresultat', command=registrer\_eksamensresultat\_mengde)

btn\_slett2.grid(row=20, column=4, padx=5, pady=5, sticky=E)

btn\_tilbake = Button(registrer\_eksamensresultat\_vindu, text='Tilbake til meny', command=registrer\_eksamensresultat\_vindu.destroy)

btn\_tilbake.grid(row=20, column=0, padx=15, pady=25, sticky=W)

liste\_eksamen.bind('<<ListboxSelect>>', hente\_student\_eksamensresultat)

def oppdatere\_eksamensresultat\_GUI():

def sok\_en\_karakter():

# Kode for å søke etter karakter:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

studentnr = snr.get()

emnekode = ek.get()

dato = date.get()

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

markor = mindatabase.cursor()

markor.execute('SELECT \* FROM Eksamensresultat')

data = markor.fetchall()

for row in data:

if studentnr == row[0] and emnekode == row[1] and dato\_konvertert == row[2]:

grade.set(row[3])

markor.close()

mindatabase.close()

def oppdatere\_en\_karakter():

# Kode for å oppdatere karakter:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

studentnr = snr.get()

emnekode = ek.get()

dato = date.get()

karakter = grade2.get()

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

markor = mindatabase.cursor()

endre\_karakter = ('Update Eksamensresultat SET Karakter = %s WHERE Studentnr = %s AND Emnekode = %s AND Dato = %s')

data\_ny\_karakter = (karakter, studentnr, emnekode, dato\_konvertert,)

markor.execute(endre\_karakter, data\_ny\_karakter)

mindatabase.commit()

messagebox.showinfo('Info', 'Eksamenskarakteren er oppdatert')

markor.close()

mindatabase.close()

def sok\_eksamensresultat():

# Kode for å hente frem studenter med eksamensresultat innenfor en eksamen:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

emnekode = ek2.get()

dato = date2.get()

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

markor = mindatabase.cursor()

sql = 'SELECT Studentnr, Emnekode, Dato, Karakter FROM Eksamensresultat WHERE Emnekode = %s AND Dato = %s'

data\_eksamen = (emnekode, dato\_konvertert,)

markor.execute(sql, data\_eksamen)

data = markor.fetchall()

studenter = []

for row in data:

studenter += [row[0]]

innhold\_eksamensresultat.set(tuple(studenter))

markor.close()

mindatabase.close()

def hente\_student\_eksamensresultat(handling):

# Koble valget fra student listen sammen med set'er boksene:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

valgt = liste\_eksamen.get(liste\_eksamen.curselection())

emnekode = ek2.get()

dato = date2.get()

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

markor = mindatabase.cursor()

markor.execute('SELECT \* FROM Eksamensresultat')

data = markor.fetchall()

for row in data:

if valgt == row[0] and emnekode == row[1] and dato\_konvertert == row[2]:

snr2.set(row[0])

ek2.set(row[1])

date2.set(row[2])

grade3.set(row[3])

markor.close()

mindatabase.close()

def oppdater\_eksamensresultat\_mengde():

# Utføre oppdatering av karakter:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

studentnr = snr2.get()

emnekode = ek2.get()

dato = date2.get()

karakter = grade4.get()

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

markor = mindatabase.cursor()

endre\_karakter = ('Update Eksamensresultat SET Karakter = %s WHERE Studentnr = %s AND Emnekode = %s AND Dato = %s')

data\_ny\_karakter = (karakter, studentnr, emnekode, dato\_konvertert,)

markor.execute(endre\_karakter, data\_ny\_karakter)

mindatabase.commit()

messagebox.showinfo('Info', 'Eksamenskarakteren er oppdatert')

markor.close()

mindatabase.close()

# GUI oppdatere eksamensresultat:

oppdater\_eksamensresultat\_vindu = Toplevel()

oppdater\_eksamensresultat\_vindu.title('Eksamensresultat (oppdatering)')

oppdater\_eksamensresultat\_vindu.resizable(FALSE, FALSE)

oppdater\_eksamensresultat\_vindu.geometry('478x620+600+100')

overskrift = Label(oppdater\_eksamensresultat\_vindu, text='Eksamensresultat oppdatering', font=('Sans-Serif', '22'))

overskrift.grid(row=0, columnspan=4, padx=15, pady=5)

forklaring\_sok = Label(oppdater\_eksamensresultat\_vindu, text='Skriv inn studentnr, emnekode og dato for å søke på student', font=('Sans-Serif', '11'))

forklaring\_sok.grid(row=1, columnspan=4, padx=15, pady=(10, 30))

btn\_sok = Button(oppdater\_eksamensresultat\_vindu, text='Søk student', command=sok\_en\_karakter)

btn\_sok.grid(row=2, column=3, padx=5, pady=5, sticky=E)

lbl\_studentnr = Label(oppdater\_eksamensresultat\_vindu, text='Studentnr:')

lbl\_studentnr.grid(row=2, column=0, padx=5, pady=5, sticky=E)

snr = StringVar()

ent\_snr = Entry(oppdater\_eksamensresultat\_vindu, width=6, textvariable=snr)

ent\_snr.grid(row=2, column=1, padx=5, pady=5, sticky=W)

lbl\_emnekode = Label(oppdater\_eksamensresultat\_vindu, text='Emnekode:')

lbl\_emnekode.grid(row=3, column=0, padx=5, pady=5, sticky=E)

ek = StringVar()

ent\_ek = Entry(oppdater\_eksamensresultat\_vindu, width=8, textvariable=ek)

ent\_ek.grid(row=3, column=1, padx=5, pady=5, sticky=W)

lbl\_dato = Label(oppdater\_eksamensresultat\_vindu, text='Dato (yyyy-mm-dd):')

lbl\_dato.grid(row=4, column=0, padx=5, pady=5, sticky=E)

date = StringVar()

ent\_date = Entry(oppdater\_eksamensresultat\_vindu, width=10, textvariable=date)

ent\_date.grid(row=4, column=1, padx=5, pady=5, sticky=W)

lbl\_karakter = Label(oppdater\_eksamensresultat\_vindu, text='Karakter:')

lbl\_karakter.grid(row=5, column=0, padx=5, pady=5, sticky=E)

grade = StringVar()

ent\_grade = Entry(oppdater\_eksamensresultat\_vindu, state='readonly', width=6, textvariable=grade)

ent\_grade.grid(row=5, column=1, padx=5, pady=5, sticky=W)

lbl\_ny\_karakter = Label(oppdater\_eksamensresultat\_vindu, text='Ny karakter:')

lbl\_ny\_karakter.grid(row=6, column=0, padx=5, pady=5, sticky=E)

grade2 = StringVar()

ent\_grade2 = Entry(oppdater\_eksamensresultat\_vindu, width=6, textvariable=grade2)

ent\_grade2.grid(row=6, column=1, padx=5, pady=5, sticky=W)

btn\_lagre = Button(oppdater\_eksamensresultat\_vindu, text='Oppdater eksamensresultat', command=oppdatere\_en\_karakter)

btn\_lagre.grid(row=7, column=3, padx=5, pady=5, sticky=E)

tkinter.ttk.Separator(oppdater\_eksamensresultat\_vindu, orient=HORIZONTAL).grid(row=8, columnspan=4, pady=10, sticky='WE')

forklaring\_sok2 = Label(oppdater\_eksamensresultat\_vindu, text='Skriv inn emnekode og dato for å søke på eksamen', font=('Sans-Serif', '11'))

forklaring\_sok2.grid(row=9, columnspan=4, padx=15, pady=(10, 30))

btn\_sok2 = Button(oppdater\_eksamensresultat\_vindu, text='Søk eksamen', command=sok\_eksamensresultat)

btn\_sok2.grid(row=10, column=3, padx=5, pady=5, sticky=E)

lbl\_studentnr2 = Label(oppdater\_eksamensresultat\_vindu, text='Studentnr:')

lbl\_studentnr2.grid(row=10, column=0, padx=5, pady=5, sticky=E)

snr2 = StringVar()

ent\_snr2 = Entry(oppdater\_eksamensresultat\_vindu, state='readonly', width=6, textvariable=snr2)

ent\_snr2.grid(row=10, column=1, padx=5, pady=5, sticky=W)

y\_scroll = Scrollbar(oppdater\_eksamensresultat\_vindu, orient=VERTICAL)

y\_scroll.grid(row=10, rowspan=4, column=2, padx=(100, 0), pady=5, sticky=NS)

innhold\_eksamensresultat = StringVar()

liste\_eksamen = Listbox(oppdater\_eksamensresultat\_vindu, width=15, height=7, listvariable=innhold\_eksamensresultat, yscrollcommand=y\_scroll.set)

liste\_eksamen.grid(row=10, rowspan=4, column=2, padx=5, pady=5)

y\_scroll['command'] = liste\_eksamen.yview

lbl\_emnekode2 = Label(oppdater\_eksamensresultat\_vindu, text='Emnekode:')

lbl\_emnekode2.grid(row=11, column=0, padx=5, pady=5, sticky=E)

ek2 = StringVar()

ent\_ek2 = Entry(oppdater\_eksamensresultat\_vindu, width=8, textvariable=ek2)

ent\_ek2.grid(row=11, column=1, padx=5, pady=5, sticky=W)

lbl\_dato2 = Label(oppdater\_eksamensresultat\_vindu, text='Dato (yyyy-mm-dd):')

lbl\_dato2.grid(row=12, column=0, padx=5, pady=5, sticky=E)

date2 = StringVar()

ent\_date2 = Entry(oppdater\_eksamensresultat\_vindu, width=10, textvariable=date2)

ent\_date2.grid(row=12, column=1, padx=5, pady=5, sticky=W)

lbl\_karakter2 = Label(oppdater\_eksamensresultat\_vindu, text='Karakter:')

lbl\_karakter2.grid(row=13, column=0, padx=5, pady=5, sticky=E)

grade3 = StringVar()

ent\_grade2 = Entry(oppdater\_eksamensresultat\_vindu, state='readonly', width=6, textvariable=grade3)

ent\_grade2.grid(row=13, column=1, padx=5, pady=5, sticky=W)

lbl\_ny\_karakter2 = Label(oppdater\_eksamensresultat\_vindu, text='Ny karakter:')

lbl\_ny\_karakter2.grid(row=14, column=0, padx=5, pady=5, sticky=E)

grade4 = StringVar()

ent\_grade3 = Entry(oppdater\_eksamensresultat\_vindu, width=6, textvariable=grade4)

ent\_grade3.grid(row=14, column=1, padx=5, pady=5, sticky=W)

btn\_lagre2 = Button(oppdater\_eksamensresultat\_vindu, text='Oppdater eksamensresultat', command=oppdater\_eksamensresultat\_mengde)

btn\_lagre2.grid(row=15, column=3, padx=5, pady=5, sticky=E)

btn\_tilbake = Button(oppdater\_eksamensresultat\_vindu, text='Tilbake til meny', command=oppdater\_eksamensresultat\_vindu.destroy)

btn\_tilbake.grid(row=15, column=0, padx=15, pady=25, sticky=W)

liste\_eksamen.bind('<<ListboxSelect>>', hente\_student\_eksamensresultat)

def slette\_eksamensresultat\_GUI():

def sok\_en\_karakter():

# Kode for å søke på karakter:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

studentnr = snr.get()

emnekode = ek.get()

dato = date.get()

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

markor = mindatabase.cursor()

markor.execute('SELECT \* FROM Eksamensresultat')

data = markor.fetchall()

for row in data:

if studentnr == row[0] and emnekode == row[1] and dato\_konvertert == row[2]:

grade.set(row[3])

markor.close()

mindatabase.close()

def slette\_en\_karakter():

# Kode for å slette karakter:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

studentnr = snr.get()

emnekode = ek.get()

dato = date.get()

karakter = grade.get()

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

markor = mindatabase.cursor()

endre\_karakter = ('DELETE FROM Eksamensresultat WHERE Studentnr = %s AND Emnekode = %s AND Dato = %s AND Karakter =%s')

data\_ny\_karakter = (studentnr, emnekode, dato\_konvertert, karakter,)

markor.execute(endre\_karakter, data\_ny\_karakter)

mindatabase.commit()

messagebox.showinfo('Info', 'Eksamenskarakteren er slettet')

markor.close()

mindatabase.close()

def sok\_eksamensresultat():

# Kode for å hente frem studenter med eksamensresultat innenfor en eksamen:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

emnekode = ek2.get()

dato = date2.get()

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

markor = mindatabase.cursor()

sql = 'SELECT Studentnr, Emnekode, Dato, Karakter FROM Eksamensresultat WHERE Emnekode = %s AND Dato = %s'

data\_eksamen = (emnekode, dato\_konvertert,)

markor.execute(sql, data\_eksamen)

data = markor.fetchall()

studenter = []

for row in data:

studenter += [row[0]]

innhold\_eksamensresultat.set(tuple(studenter))

markor.close()

mindatabase.close()

def hente\_student\_eksamensresultat(handling):

# Koble valget fra student listen sammen med set'er boksene:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

valgt = liste\_eksamen.get(liste\_eksamen.curselection())

emnekode = ek2.get()

dato = date2.get()

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

markor = mindatabase.cursor()

markor.execute('SELECT \* FROM Eksamensresultat')

data = markor.fetchall()

for row in data:

if valgt == row[0] and emnekode == row[1] and dato\_konvertert == row[2]:

snr2.set(row[0])

ek2.set(row[1])

date2.set(row[2])

grade3.set(row[3])

markor.close()

mindatabase.close()

def slett\_eksamensresultat\_mengde():

# Utføre sletting av karakter:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

studentnr = snr2.get()

emnekode = ek2.get()

dato = date2.get()

karakter = grade3.get()

dato\_konvertert = datetime.strptime(dato, '%Y-%m-%d').date()

markor = mindatabase.cursor()

endre\_karakter = ('DELETE FROM Eksamensresultat WHERE Studentnr = %s AND Emnekode = %s AND Dato = %s AND Karakter = %s')

data\_ny\_karakter = (studentnr, emnekode, dato\_konvertert, karakter,)

markor.execute(endre\_karakter, data\_ny\_karakter)

mindatabase.commit()

messagebox.showinfo('Info', 'Eksamenskarakteren er slettet')

markor.close()

mindatabase.close()

# GUI slette eksamensresultat:

slett\_eksamensresultat\_vindu = Toplevel()

slett\_eksamensresultat\_vindu.title('Eksamensresultat (sletting)')

slett\_eksamensresultat\_vindu.resizable(FALSE, FALSE)

slett\_eksamensresultat\_vindu.geometry('478x560+600+100')

overskrift = Label(slett\_eksamensresultat\_vindu, text='Eksamensresultat sletting', font=('Sans-Serif', '22'))

overskrift.grid(row=0, columnspan=4, padx=15, pady=5)

forklaring\_sok = Label(slett\_eksamensresultat\_vindu, text='Skriv inn studentnr, emnekode og dato for å søke på student', font=('Sans-Serif', '11'))

forklaring\_sok.grid(row=1, columnspan=4, padx=15, pady=(10, 30))

btn\_sok = Button(slett\_eksamensresultat\_vindu, text='Søk student', command=sok\_en\_karakter)

btn\_sok.grid(row=2, column=3, padx=5, pady=5, sticky=E)

lbl\_studentnr = Label(slett\_eksamensresultat\_vindu, text='Studentnr:')

lbl\_studentnr.grid(row=2, column=0, padx=5, pady=5, sticky=E)

snr = StringVar()

ent\_snr = Entry(slett\_eksamensresultat\_vindu, width=6, textvariable=snr)

ent\_snr.grid(row=2, column=1, padx=5, pady=5, sticky=W)

lbl\_emnekode = Label(slett\_eksamensresultat\_vindu, text='Emnekode:')

lbl\_emnekode.grid(row=3, column=0, padx=5, pady=5, sticky=E)

ek = StringVar()

ent\_ek = Entry(slett\_eksamensresultat\_vindu, width=8, textvariable=ek)

ent\_ek.grid(row=3, column=1, padx=5, pady=5, sticky=W)

lbl\_dato = Label(slett\_eksamensresultat\_vindu, text='Dato (yyyy-mm-dd):')

lbl\_dato.grid(row=4, column=0, padx=5, pady=5, sticky=E)

date = StringVar()

ent\_date = Entry(slett\_eksamensresultat\_vindu, width=10, textvariable=date)

ent\_date.grid(row=4, column=1, padx=5, pady=5, sticky=W)

lbl\_karakter = Label(slett\_eksamensresultat\_vindu, text='Karakter:')

lbl\_karakter.grid(row=5, column=0, padx=5, pady=5, sticky=E)

grade = StringVar()

ent\_grade = Entry(slett\_eksamensresultat\_vindu, width=6, textvariable=grade)

ent\_grade.grid(row=5, column=1, padx=5, pady=5, sticky=W)

btn\_slett = Button(slett\_eksamensresultat\_vindu, text='Slett eksamensresultat', command=slette\_en\_karakter)

btn\_slett.grid(row=6, column=3, padx=5, pady=5, sticky=E)

tkinter.ttk.Separator(slett\_eksamensresultat\_vindu, orient=HORIZONTAL).grid(row=7, columnspan=4, pady=10, sticky='WE')

forklaring\_sok2 = Label(slett\_eksamensresultat\_vindu, text='Skriv inn emnekode og dato for å søke på eksamen', font=('Sans-Serif', '11'))

forklaring\_sok2.grid(row=8, columnspan=4, padx=15, pady=(10, 30))

btn\_sok2 = Button(slett\_eksamensresultat\_vindu, text='Søk eksamen', command=sok\_eksamensresultat)

btn\_sok2.grid(row=9, column=3, padx=5, pady=5, sticky=E)

lbl\_studentnr2 = Label(slett\_eksamensresultat\_vindu, text='Studentnr:')

lbl\_studentnr2.grid(row=9, column=0, padx=5, pady=5, sticky=E)

snr2 = StringVar()

ent\_snr2 = Entry(slett\_eksamensresultat\_vindu, state='readonly', width=6, textvariable=snr2)

ent\_snr2.grid(row=9, column=1, padx=5, pady=5, sticky=W)

y\_scroll = Scrollbar(slett\_eksamensresultat\_vindu, orient=VERTICAL)

y\_scroll.grid(row=9, rowspan=4, column=2, padx=(100, 0), pady=5, sticky=NS)

innhold\_eksamensresultat = StringVar()

liste\_eksamen = Listbox(slett\_eksamensresultat\_vindu, width=15, height=7, listvariable=innhold\_eksamensresultat, yscrollcommand=y\_scroll.set)

liste\_eksamen.grid(row=9, rowspan=4, column=2, padx=5, pady=5)

y\_scroll['command'] = liste\_eksamen.yview

lbl\_emnekode2 = Label(slett\_eksamensresultat\_vindu, text='Emnekode:')

lbl\_emnekode2.grid(row=10, column=0, padx=5, pady=5, sticky=E)

ek2 = StringVar()

ent\_ek2 = Entry(slett\_eksamensresultat\_vindu, width=8, textvariable=ek2)

ent\_ek2.grid(row=10, column=1, padx=5, pady=5, sticky=W)

lbl\_dato2 = Label(slett\_eksamensresultat\_vindu, text='Dato (yyyy-mm-dd):')

lbl\_dato2.grid(row=11, column=0, padx=5, pady=5, sticky=E)

date2 = StringVar()

ent\_date2 = Entry(slett\_eksamensresultat\_vindu, width=10, textvariable=date2)

ent\_date2.grid(row=11, column=1, padx=5, pady=5, sticky=W)

lbl\_karakter2 = Label(slett\_eksamensresultat\_vindu, text='Karakter:')

lbl\_karakter2.grid(row=12, column=0, padx=5, pady=5, sticky=E)

grade3 = StringVar()

ent\_grade = Entry(slett\_eksamensresultat\_vindu, state='readonly', width=6, textvariable=grade3)

ent\_grade.grid(row=12, column=1, padx=5, pady=5, sticky=W)

btn\_slett2 = Button(slett\_eksamensresultat\_vindu, text='Slett eksamensresultat', command=slett\_eksamensresultat\_mengde)

btn\_slett2.grid(row=14, column=3, padx=5, pady=5, sticky=E)

btn\_tilbake = Button(slett\_eksamensresultat\_vindu, text='Tilbake til meny', command=slett\_eksamensresultat\_vindu.destroy)

btn\_tilbake.grid(row=14, column=0, padx=15, pady=25, sticky=W)

liste\_eksamen.bind('<<ListboxSelect>>', hente\_student\_eksamensresultat)

def utskrift\_eksamensresultat\_GUI():

def utskrift\_karakterliste():

# Kode for utskrift av karakterliste:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

emnekode = ek.get()

utskrift\_markor = mindatabase.cursor()

sql = 'SELECT \* FROM Eksamensresultat WHERE Emnekode = %s ORDER BY Studentnr'

data\_utskrift = (emnekode,)

utskrift\_markor.execute(sql, data\_utskrift)

data = utskrift\_markor.fetchall()

tekst\_vindu.delete('1.0', END)

for row in data:

if row[3] != None:

tekst\_vindu.insert(END, "Snr:" + str(row[0]) + " | " + "Emnek: " + str(row[1]) + " | " + "Dato: " + str(row[2]) + " | " + "Karakter: " + str(row[3]) + "\n")

utskrift\_markor.close()

mindatabase.close()

def utskrift\_karakterstatistikk():

# Kode for utskrift av karakterstatistikk utifra valgt dato:

mindatabase = mysql.connector.connect(host='localhost', port=3306, user='Eksamenssjef', password='oblig2021', db='oblig2021')

emnekode = ek.get()

dato\_tastet = dato.get()

dato\_konvertert = datetime.strptime(dato\_tastet, '%Y-%m-%d').date()

utskrift\_markor = mindatabase.cursor()

sql = 'SELECT \* FROM Eksamensresultat WHERE Emnekode = %s AND Dato = %s ORDER BY Studentnr'

data\_utskrift = (emnekode, dato\_konvertert,)

utskrift\_markor.execute(sql, data\_utskrift)

data = utskrift\_markor.fetchall()

karakter\_A = 0

karakter\_B = 0

karakter\_C = 0

karakter\_D = 0

karakter\_E = 0

karakter\_F = 0

tekst\_vindu.delete('1.0', END)

for row in data:

if row[3] != None:

tekst\_vindu.insert(END, "Snr:" + str(row[0]) + " | " + "Emnek: " + str(row[1]) + " | " + "Dato: " + str(row[2]) + " | " + "Karakter: " + str(row[3]) + "\n")

if row[3] == 'A':

karakter\_A += 1

elif row[3] == 'B':

karakter\_B += 1

elif row[3] == 'C':

karakter\_C += 1

elif row[3] == 'D':

karakter\_D += 1

elif row[3] == 'E':

karakter\_E += 1

elif row[3] == 'F':

karakter\_F += 1

karaktera.set(karakter\_A)

karakterb.set(karakter\_B)

karakterc.set(karakter\_C)

karakterd.set(karakter\_D)

karaktere.set(karakter\_E)

karakterf.set(karakter\_F)

utskrift\_markor.close()

mindatabase.close()

# GUI utskrift eksamensresultat:

utskrift\_eksamensresultat\_vindu = Toplevel()

utskrift\_eksamensresultat\_vindu.title('Eksamensresultater (utskrift)')

utskrift\_eksamensresultat\_vindu.resizable(FALSE, FALSE)

utskrift\_eksamensresultat\_vindu.geometry('535x570+600+100')

overskrift = Label(utskrift\_eksamensresultat\_vindu, text='Eksamensresultat utskrift', font=('Sans-Serif', '22'))

overskrift.grid(row=0, columnspan=7, padx=15, pady=5)

forklaring\_utskrift = Label(utskrift\_eksamensresultat\_vindu, text='Velg utskrift av "karakterliste" eller karakterstatistikk nederst', font=('Sans-Serif', '11'))

forklaring\_utskrift.grid(row=1, columnspan=7, padx=15, pady=(10, 30))

lbl\_dato = Label(utskrift\_eksamensresultat\_vindu, text='Dato (yyyy-mm-dd):')

lbl\_dato.grid(row=2, column=1, padx=5, pady=5, stick=E)

dato = StringVar()

ent\_dato = Entry(utskrift\_eksamensresultat\_vindu, width=10, textvariable=dato)

ent\_dato.grid(row=2, column=2, padx=5, pady=5, sticky=W)

lbl\_ek = Label(utskrift\_eksamensresultat\_vindu, text='Emnekode:')

lbl\_ek.grid(row=3, column=1, padx=5, pady=5, sticky=E)

ek = StringVar()

ent\_ek = Entry(utskrift\_eksamensresultat\_vindu, width=10, textvariable=ek)

ent\_ek.grid(row=3, column=2, padx=5, pady=5, sticky=W)

lbl\_karaktera = Label(utskrift\_eksamensresultat\_vindu, text='A:')

lbl\_karaktera.grid(row=3, column=3, padx=(0, 0), pady=5, sticky=W)

karaktera = StringVar()

ent\_karaktera = Entry(utskrift\_eksamensresultat\_vindu, state='readonly', width=2, textvariable=karaktera)

ent\_karaktera.grid(row=3, column=3, padx=(20, 0), pady=5, sticky=W)

lbl\_karakterb = Label(utskrift\_eksamensresultat\_vindu, text='B:')

lbl\_karakterb.grid(row=3, column=3, padx=(40, 0), pady=5, sticky=W)

karakterb = StringVar()

ent\_karakterb = Entry(utskrift\_eksamensresultat\_vindu, state='readonly', width=2, textvariable=karakterb)

ent\_karakterb.grid(row=3, column=3, padx=(60, 0), pady=5, sticky=W)

lbl\_karakterc = Label(utskrift\_eksamensresultat\_vindu, text='C:')

lbl\_karakterc.grid(row=3, column=3, padx=(80, 0), pady=5, sticky=W)

karakterc = StringVar()

ent\_karakterc = Entry(utskrift\_eksamensresultat\_vindu, state='readonly', width=2, textvariable=karakterc)

ent\_karakterc.grid(row=3, column=3, padx=(100, 0), pady=5, sticky=W)

lbl\_karakterd = Label(utskrift\_eksamensresultat\_vindu, text='D:')

lbl\_karakterd.grid(row=3, column=3, padx=(120, 0), pady=5, sticky=W)

karakterd = StringVar()

ent\_karakterd = Entry(utskrift\_eksamensresultat\_vindu, state='readonly', width=2, textvariable=karakterd)

ent\_karakterd.grid(row=3, column=3, padx=(140, 0), pady=5, sticky=W)

lbl\_karaktere = Label(utskrift\_eksamensresultat\_vindu, text='E:')

lbl\_karaktere.grid(row=3, column=3, padx=(160, 0), pady=5, sticky=W)

karaktere = StringVar()

ent\_karaktere = Entry(utskrift\_eksamensresultat\_vindu, state='readonly', width=2, textvariable=karaktere)

ent\_karaktere.grid(row=3, column=3, padx=(180, 0), pady=5, sticky=W)

lbl\_karakterf = Label(utskrift\_eksamensresultat\_vindu, text='F:')

lbl\_karakterf.grid(row=3, column=3, padx=(200, 0), pady=5, sticky=W)

karakterf = StringVar()

ent\_karakterf = Entry(utskrift\_eksamensresultat\_vindu, state='readonly', width=2, textvariable=karakterf)

ent\_karakterf.grid(row=3, column=3, padx=(220, 0), pady=5, sticky=W)

tekst\_vindu = Text(utskrift\_eksamensresultat\_vindu, width=65, height=20)

tekst\_vindu.grid(row=4, columnspan=5, padx=5, pady=5)

btn\_hent\_data1 = Button(utskrift\_eksamensresultat\_vindu, text='Utskrift av "karakterliste"', command=utskrift\_karakterliste)

btn\_hent\_data1.grid(row=5, column=2, padx=5, pady=5, sticky=SE)

btn\_hent\_data2 = Button(utskrift\_eksamensresultat\_vindu, text='Utskrift av karakterstatistikk', command=utskrift\_karakterstatistikk)

btn\_hent\_data2.grid(row=5, column=3, padx=15, pady=5, sticky=SE)

btn\_tilbake2 = Button(utskrift\_eksamensresultat\_vindu, text='Tilbake til meny', command=utskrift\_eksamensresultat\_vindu.destroy)

btn\_tilbake2.grid(row=5, column=1, padx=15, pady=5, sticky=SW)

def main():

# GUI meny:

hovedvindu = Tk()

hovedvindu.title('Meny')

hovedvindu.resizable(FALSE, FALSE)

hovedvindu.geometry('500x300+600+200')

overskrift = Label(hovedvindu, text='USN Student og eksamens program', font=('Sans-Serif', '22'))

overskrift.grid(row=0, columnspan=5, padx=15, pady=15)

text\_student = Label(hovedvindu, text='Student:', font=('Sans-Serif', '10'))

text\_student.grid(row=1, column=0, padx=15, pady=5)

btn\_registrere\_student = Button(hovedvindu, text="Registrere", command=registrere\_student\_GUI)

btn\_registrere\_student.grid(row=1, column=1, padx=5, pady=20, sticky=W)

btn\_oppdatere\_student = Button(hovedvindu, text='Oppdatere', command=oppdatere\_student\_GUI)

btn\_oppdatere\_student.grid(row=1, column=2, padx=5, pady=20, sticky=W)

btn\_slette\_student = Button(hovedvindu, text='Slette', command=slette\_student\_GUI)

btn\_slette\_student.grid(row=1, column=3, padx=5, pady=20, sticky=W)

btn\_utskrift\_student = Button(hovedvindu, text='Utskrift', command=utskrift\_eksamensresultater\_student\_GUI)

btn\_utskrift\_student.grid(row=1, column=4, padx=5, pady=20, sticky=W)

text\_eksamen = Label(hovedvindu, text='Eksamen:', font=('Sans-Serif', '10'))

text\_eksamen.grid(row=2, column=0, padx=15, pady=5)

btn\_registrere\_eksamen = Button(hovedvindu, text="Registrere", command=registrere\_eksamen\_GUI)

btn\_registrere\_eksamen.grid(row=2, column=1, padx=5, pady=20, sticky=W)

btn\_oppdatere\_eksamen = Button(hovedvindu, text='Oppdatere', command=oppdatere\_eksamen\_GUI)

btn\_oppdatere\_eksamen.grid(row=2, column=2, padx=5, pady=20, sticky=W)

btn\_slette\_eksamen = Button(hovedvindu, text='Slette', command=slette\_eksamen\_GUI)

btn\_slette\_eksamen.grid(row=2, column=3, padx=5, pady=20, sticky=W)

btn\_utskrift\_eksamen = Button(hovedvindu, text='Utskrift', command=utskrift\_eksamen\_GUI)

btn\_utskrift\_eksamen.grid(row=2, column=4, padx=5, pady=20, sticky=W)

text\_eksamensresultat = Label(hovedvindu, text='Eksamensresultat:', font=('Sans-Serif', '10'))

text\_eksamensresultat.grid(row=3, column=0, padx=15, pady=5)

btn\_registrere\_eksamensresultat = Button(hovedvindu, text="Registrere", command=registrere\_eksamensresultat\_GUI)

btn\_registrere\_eksamensresultat.grid(row=3, column=1, padx=5, pady=20, sticky=W)

btn\_oppdatere\_eksamensresultat = Button(hovedvindu, text='Oppdatere', command=oppdatere\_eksamensresultat\_GUI)

btn\_oppdatere\_eksamensresultat.grid(row=3, column=2, padx=5, pady=20, sticky=W)

btn\_slette\_eksamenresultat = Button(hovedvindu, text='Slette', command=slette\_eksamensresultat\_GUI)

btn\_slette\_eksamenresultat.grid(row=3, column=3, padx=5, pady=20, sticky=W)

btn\_utskrift\_eksamensresultat = Button(hovedvindu, text='Utskrift', command=utskrift\_eksamensresultat\_GUI)

btn\_utskrift\_eksamensresultat.grid(row=3, column=4, padx=5, pady=20, sticky=W)

hovedvindu.mainloop()

main()