

MATERI WORKSHOP

PERSIAPAN TOOL

1. Install Python 3.4 berikut PIP
2. Install XAMPP jika menggunakan OS win , jika OS linux silakan install apache, mysql dan phpmyadmin
3. Install library yang dibutuhkan :

Install library PyMySQL

```
python setup.py install
```

```
pip install pyMySQL
```

install library requests

```
python setup.py install
```

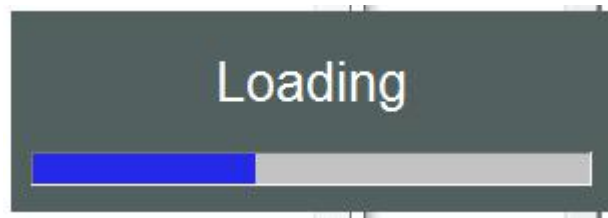
```
pip install requests
```

install library cx_freeze

```
pip install cx_Freeze-4.3.4-cp34-none-win_amd64.whl
```

4. Memiliki image **icon.ico** untuk icon aplikasi yang akan di build

1. Create Progress Bar



Berikut adalah Sorce Codenya **bar.py**

```
import tkinter as tk
from tkinter import *
from tkinter import ttk

class SplashScreen:
    def __init__(self, parent):
        self.parent = parent
        self.aturWindow()

    def aturWindow(self):
        # ambil ukuran dari file image
        lebar = 300
        tinggi = 100

        setengahLebar = (self.parent.winfo_screenwidth()-lebar)//2
        setengahTinggi = (self.parent.winfo_screenheight()-tinggi)//2

        # atur posisi window di tengah-tengah layar
        self.parent.geometry("%ix%i+%i+%i" %(lebar,
tinggi,setengahLebar,setengahTinggi))

        # atur Image via Komponen Label
        Label(self.parent, text="Loading",bg="#526060",fg="white",
font=("Helvetica", 20), width=9, height=2).pack(side=TOP,anchor=N,fill=X)
        s = ttk.Style()
        s.theme_use('alt')
        s.configure("red.Horizontal.TProgressbar", background='#2529E8',
foreground="white", relief=FLAT,bd=0,height=2)
```

```

        self.progress = ttk.Progressbar(self.parent,
style="red.Horizontal.TProgressbar",orient="horizontal",length=600,
mode="determinate")
        self.progress.pack(side=TOP,anchor=N,padx=10)
        self.bytes = 0
        self.maxbytes = 0
        self.start()

def start(self):
    self.progress["value"] = 0
    self.maxbytes = 50000
    self.progress["maximum"] = 50000
    self.read_bytes()
    if self.progress["value"] == 60000:
        self.parent.destroy()
        if self.parent.destroy:
            #import login
            #login.main()

def read_bytes(self):
    "simulate reading 500 bytes; update progress bar"
    #self.bytes += 40+self.bytes
    self.bytes += 1000
    self.progress["value"] = self.bytes
    self.progress.after(100, self.start)

def main():
    root = Tk()
    root.configure(bg="#526060")
    # menghilangkan judul dan batas frame Window
    root.overridereirect(True)
    SplashScreen(root)
    root.mainloop()

main()

```

2. Create Form Login



Berikut adalah source code dari **login.py** :

```
from tkinter import *
import tkinter
import tkinter.messagebox as mb

datUser = "sas"
datPassword = "....."

class Login:
    def __init__(self, induk, title):
        self.induk = induk
        self.aturWindow(title)
        self.aturKomponen()
        self.entUser.focus_set()

    def aturWindow(self, title):
        lebar = 400
        tinggi = 150

        setTengahX = (self.induk.winfo_screenwidth()-lebar)/2
        setTengahY = (self.induk.winfo_screenheight()-tinggi)/2

        self.induk.geometry("%ix%i+%i+%i" %(lebar, tinggi, setTengahX,
setTengahY))
        self.induk.title(title)
```

```
self.induk.protocol("WM_DELETE_WINDOW", self.Tutup)
```

```
def aturKomponen(self):
```

```
    # atur frame utama
```

```
    frameUtama = Frame(self.induk, bd=10)
```

```
    frameUtama.pack(fill=BOTH, expand=YES)
```

```
    # atur frame data
```

```
    frData = Frame(frameUtama, bd=5)
```

```
    frData.pack(fill=BOTH, expand=YES)
```

```
    # atur input username
```

```
    Label(frData, text='User').grid(row=0, column=0, sticky=W)
```

```
    self.entUser = Entry(frData)
```

```
    self.entUser.grid(row=0, column=1)
```

```
    # atur input password
```

```
    Label(frData, text='Password').grid(row=1, column=0, sticky=W)
```

```
    self.entPass = Entry(frData, show='*')
```

```
    self.entPass.grid(row=1, column=1)
```

```
    # atur cek --> perlihatkan kata kunci
```

```
    self.cek = IntVar()
```

```
    self.cbShowPass = Checkbutton(frData, text='see keywords',  
                                  variable=self.cek, command=self.lihatPassword)
```

```
    self.cbShowPass.grid(row=2, column=1, sticky=E)
```

```
    # atur frame tombol
```

```
    frTombol = Frame(frameUtama, bd=5)
```

```
    frTombol.pack(fill=BOTH, expand=YES)
```

```
    # atur tombol login
```

```
    self.btnLogin = Button(frTombol, text='Login', command=self.prosesLogin)
```

```
    self.btnLogin.pack(side=LEFT, fill=BOTH, expand=YES)
```

```
    self.btnBatal = Button(frTombol, text='Cancel', command=self.Tutup)
```

```
    self.btnBatal.pack(side=LEFT, fill=BOTH, expand=YES)
```

```

def prosesLogin(self, event=None):
    # ambil data input dari pengguna
    nmUser = self.entUser.get()
    passUser = self.entPass.get()

    # logika pemrograman
    if nmUser=="":
        mb.showwarning('Warning !!', 'Username can not be empty!',
parent=self.induk)
        self.entUser.focus_set()
    elif passUser=="":
        mb.showwarning('Warning !!', 'password can not be empty!',
parent=self.induk)
        self.entPass.focus_set()
    elif (nmUser==datUser) and (passUser==datPassword):
        self.induk.destroy()
        # if self.induk.destroy:
        # import dashboard

    else:
        mb.showwarning('Warning !!', 'Username or Keyword WRONG!!',
parent=self.induk)
        self.Hapus()

def lihatPassword(self, event=None):
    nilaiCek = self.cek.get()

    if nilaiCek== 1:
        self.entPass['show'] = "
    else:
        self.entPass['show'] = '*'

def Tutup(self, event=None):
    self.induk.destroy()

def Hapus(self, event=None):
    self.entUser.delete(0, END)
    self.entPass.delete(0, END)

```

```
self.entUser.focus_set()
```

```
def main():  
    root = Tk()  
    app = Login(root, "|| Login ~ www.sasmitohrr.web.id ~ ||")  
    root.mainloop()
```

3. Create Menu Dashboard :



Berikut adalah source code **dashboard.py** :

```
from tkinter import *  
import tkinter  
  
class dashboard:  
    def __init__(self, induk, title):  
        self.induk = induk  
        self.aturWindow(title)  
        self.aturKomponen()  
  
    def aturWindow(self, title):  
        lebar = 400
```

```

tinggi = 150
setTengahX = (self.induk.winfo_screenwidth()-lebar)/2
setTengahY = (self.induk.winfo_screenheight()-tinggi)/2
self.induk.geometry("%ix%i+%i+%i" %(lebar, tinggi, setTengahX,
setTengahY))
self.induk.title(title)
self.induk.protocol("WM_DELETE_WINDOW", self.Tutup)

def aturKomponen(self):
    # atur frame utama
    frameUtama = Frame(self.induk, bd=10)
    frameUtama.pack(fill=BOTH, expand=YES)

    # atur frame data
    frData = Frame(frameUtama, bd=5)
    frData.pack(fill=BOTH, expand=YES)

    # atur LABEL
    Label(frData, text='MENU PILIHAN',fg="blue", font=("Arial Bold",
30)).grid(row=0, column=1, sticky=W)

    # atur frame tombol
    frTombol = Frame(frameUtama, bd=5)
    frTombol.pack(fill=BOTH, expand=YES)

    # atur tombol login
    self.btnCal = Button(frTombol, text='Calculator', command=self.calculator)
    self.btnCal.pack(side=LEFT, fill=BOTH, expand=YES)

    self.btnStopwatch = Button(frTombol, text='Stopwatch',
command=self.stwatch)
    self.btnStopwatch.pack(side=LEFT, fill=BOTH, expand=YES)

    self.btnCrud = Button(frTombol, text='CRUD', command=self.mhs)
    self.btnCrud.pack(side=LEFT, fill=BOTH, expand=YES)

    self.btnClose = Button(frTombol, text='Close', command=self.Tutup)
    self.btnClose.pack(side=RIGHT, fill=BOTH, expand=YES)

```



```

def Tutup(self, event=None):
    self.induk.destroy()

def calculator(self, event=None):
    self.induk.destroy()
    if self.induk.destroy:
        import kalkulator

def stwatch(self, event=None):
    self.induk.destroy()
    if self.induk.destroy:
        import stopwatch

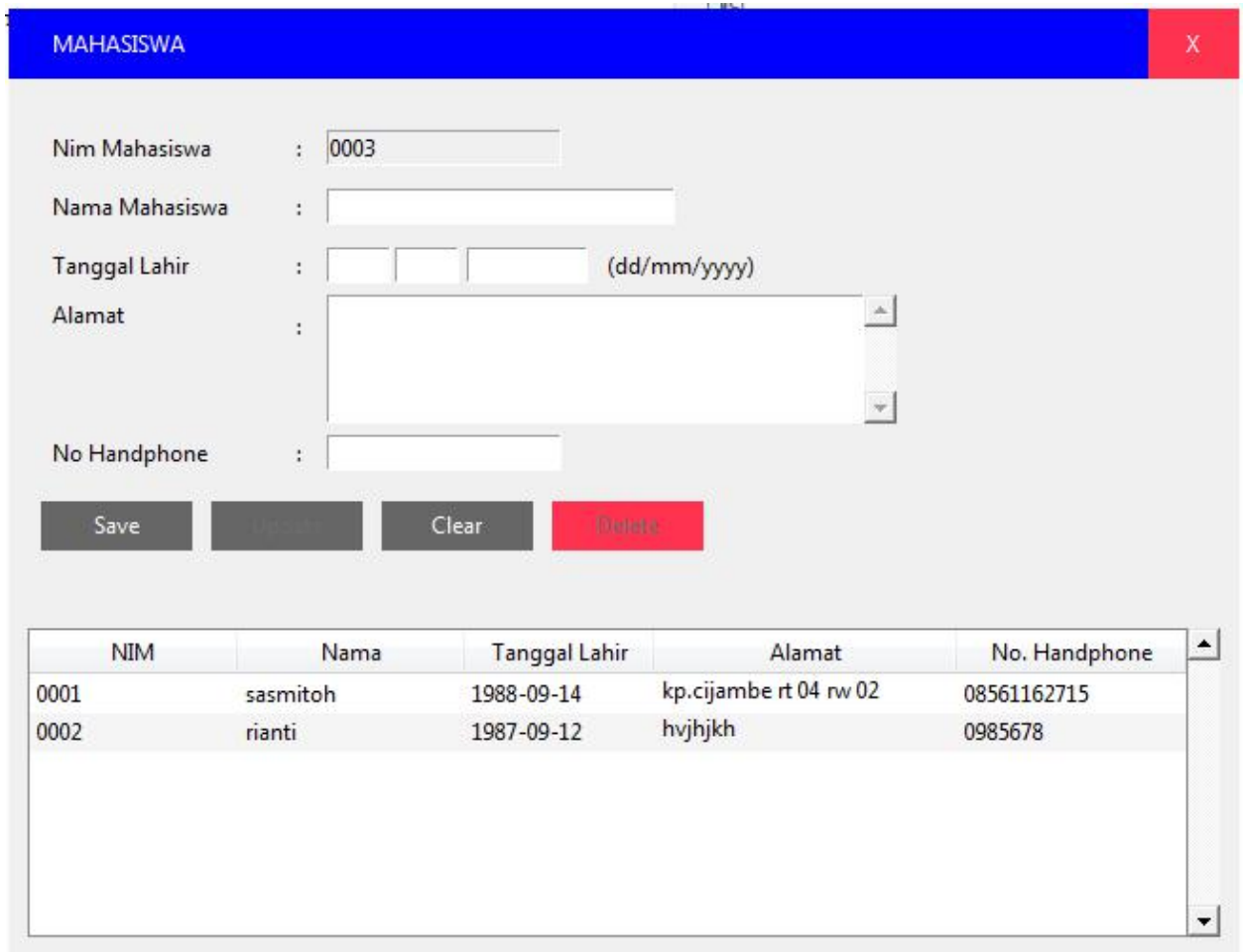
def mhs(self, event=None):
    self.induk.destroy()
    if self.induk.destroy:
        import mahasiswa

def main ():
    root = Tk()
    root.configure(bg="#526060")
    root.overrideredirect(True)
    app = dashboard(root, "Dashboard")
    root.mainloop()

main()

```

4. Create CRUD Data Mahasiswa



| NIM | Nama | Tanggal Lahir | Alamat | No. Handphone |
|------|----------|---------------|------------------------|---------------|
| 0001 | sasmitoh | 1988-09-14 | kp.cijambe rt 04 rw 02 | 08561162715 |
| 0002 | rianti | 1987-09-12 | hvjhjkh | 0985678 |

Beriku adalah Source codenya **mahasiswa.py** :

```
import tkinter
from tkinter import *
from tkinter import ttk
from tkinter.scrolledtext import ScrolledText
import pymysql
import datetime
import time

root = Tk()
class WindowDraggable():
```

```

def __init__(self, label):
    self.label = label
    label.bind('<ButtonPress-1>', self.StartMove)
    label.bind('<ButtonRelease-1>', self.StopMove)
    label.bind('<B1-Motion>', self.OnMotion)

def StartMove(self, event):
    self.x = event.x
    self.y = event.y

def StopMove(self, event):
    self.x = None
    self.y = None

def OnMotion(self, event):
    x = (event.x_root - self.x - self.label.winfo_rootx() +
self.label.winfo_rootx())
    y = (event.y_root - self.y - self.label.winfo_rooty() +
self.label.winfo_rooty())
    root.geometry("+%s+%s" % (x, y))

judul_kolom = ("NIM", "Nama", "Tanggal Lahir", "Alamat", "No. Handphone")
class Petugas:
    def __init__(self, parent):
        self.parent = parent
        self.parent.protocol("WM_DELETE_WINDOWS", self.keluar)
        lebar=650
        tinggi=500
        setTengahX = (self.parent.winfo_screenwidth()-lebar)//2
        setTengahY = (self.parent.winfo_screenheight()-tinggi)//2
        self.parent.geometry("%ix%i+%i+%i" %(lebar, tinggi, setTengahX,
setTengahY))
        self.parent.overrideredirect(1)
        self.aturKomponen()
        self.auto()

    def keluar(self, event=None):
        self.parent.destroy()

```

```

def OnDoubleClick(self, event):

    self.entKode.config(state="normal")
    self.entKode.delete(0, END)
    self.entNama.delete(0, END)
    self.entHari.delete(0, END)
    self.entAlamat.delete('1.0', 'end')
    self.entHp.delete(0, END)

    it = self.trvTabel.selection()[0]
    ck = str(self.trvTabel.item(it, "values"))[2:6]

    self.entKode.insert(END, ck)

    cKode = self.entKode.get()
    con = pymysql.connect(db="db_mahasiswa", user="root", passwd="",
host="localhost", port=3306,autocommit=True)
    cur = con.cursor()
    sql = "SELECT mahasiswa_nama, mahasiswa_tgl_lahir,
mahasiswa_alamat,mahasiswa_no_hp FROM mahasiswa WHERE
mahasiswa_kode = %s"
    cur.execute(sql,cKode)
    data = cur.fetchone()

    self.entNama.insert(END, data[0])

    #TGL Lahir
    self.entHari.insert(END, data[1])
    cTglLahir = self.entHari.get()

    pecahTahun =
str(cTglLahir[0]+cTglLahir[1]+cTglLahir[2]+cTglLahir[3])
    pecahBulan = str(cTglLahir[5]+cTglLahir[6])
    pecahHari = str(cTglLahir[8]+cTglLahir[9])

    self.entHari.delete(0, END)
    self.entBulan.delete(0, END)
    self.entTahun.delete(0, END)

```

```

self.entHari.insert(END, pecahHari)
self.entBulan.insert(END, pecahBulan)
self.entTahun.insert(END, pecahTahun)

self.entAlamat.insert(END, data[2])
self.entHp.insert(END, data[3])
self.entKode.config(state="disable")
self.btnSave.config(state="disable")
self.btnUpdate.config(state="normal")
self.btnDelete.config(state="normal")

def aturKomponen(self):
    frameWin = Frame(self.parent, bg="blue")
    frameWin.pack(fill=X,side=TOP)
    WindowDraggable(frameWin)
    Label(frameWin,
text='MAHASISWA',bg="blue",fg="white").pack(side=LEFT,padx=20)
    buttonx = Button(frameWin, text="X",fg="white", bg="#FF334F",
width=6, height=2,bd=0,\
        activebackground="#FB8072",activeforeground="white",
command=self.onClose, relief=FLAT)
    buttonx.pack(side=RIGHT)
    mainFrame = Frame(self.parent)
    mainFrame.pack(side=TOP,fill=X)
    btnFrame = Frame(self.parent)
    btnFrame.pack(side=TOP, fill=X)
    tabelFrame = Frame(self.parent)
    tabelFrame.pack( expand=YES, side=TOP,fill=Y)

    Label(mainFrame, text=' ').grid(row=0, column=0)
    Label(btnFrame, text=' ').grid(row=1, column=0)

    Label(mainFrame, text='Nim Mahasiswa').grid(row=1, column=0,
sticky=W,padx=20)
    Label(mainFrame, text=':').grid(row=1, column=1,
sticky=W,pady=5,padx=10)
    self.entKode = Entry(mainFrame, width=20)
    self.entKode.grid(row=1, column=2,sticky=W)

```

```

        Label(mainFrame, text="Nama Mahasiswa").grid(row=2, column=0,
sticky=W,padx=20)
        Label(mainFrame, text=':').grid(row=2, column=1,
sticky=W,pady=5,padx=10)
        self.entNama = Entry(mainFrame, width=30)
        self.entNama.grid(row=2, column=2,sticky=W)

        Label(mainFrame, text="Tanggal Lahir").grid(row=3, column=0,
sticky=W,padx=20)
        Label(mainFrame, text=':').grid(row=3, column=1,
sticky=W,pady=5,padx=10)

        #tgl
        tgl = Frame(mainFrame)
        tgl.grid(row=3,column=2,sticky=W)
        self.entHari = Entry(tgl, width=5)
        self.entHari.grid(row=1, column=0,sticky=W)
        self.entBulan = Entry(tgl, width=5)
        self.entBulan.grid(row=1, column=1,sticky=W,padx=2)
        self.entTahun = Entry(tgl, width=10)
        self.entTahun.grid(row=1, column=2,sticky=W,padx=2)
        Label(tgl, text='(dd/mm/yyyy)').grid(row=1, column=3,
sticky=E,padx=5)

        Label(mainFrame, text="Alamat").grid(row=4, column=0,
sticky=NW,padx=20)
        Label(mainFrame, text=':').grid(row=4, column=1,
sticky=NW,padx=10,pady=6)
        self.entAlamat = ScrolledText(mainFrame,height=4,width=35)
        self.entAlamat.grid(row=4, column=2,sticky=W)

        Label(mainFrame, text="No Handphone").grid(row=5, column=0,
sticky=W,padx=20)
        Label(mainFrame, text=':').grid(row=5, column=1,
sticky=W,pady=5,padx=10)
        self.entHp = Entry(mainFrame, width=20)
        self.entHp.grid(row=5, column=2,sticky=W)

```

```

self.btnSave = Button(btnFrame, text='Save',\
                      command=self.onSave, width=10,\
                      relief=FLAT, bd=2, bg="#666",\
                      fg="white",activebackground="#444",activeforeground="white" )
self.btnSave.grid(row=0, column=1,padx=5)

self.btnUpdate = Button(btnFrame, text='Update',\
                      command=self.onUpdate,state="disable", width=10,\
                      relief=FLAT, bd=2, bg="#666",\
                      fg="white",activebackground="#444",activeforeground="white")
self.btnUpdate.grid(row=0,column=2,pady=10, padx=5)

self.btnClear = Button(btnFrame, text='Clear',\
                      command=self.onClear, width=10,\
                      relief=FLAT, bd=2, bg="#666",\
                      fg="white",activebackground="#444",activeforeground="white")
self.btnClear.grid(row=0,column=3,pady=10, padx=5)

self.btnDelete = Button(btnFrame, text='Delete',\
                      command=self.onDelete,state="disable", width=10,\
                      relief=FLAT, bd=2, bg="#FF334F",\
                      fg="white",activebackground="#444",activeforeground="white")
self.btnDelete.grid(row=0,column=4,pady=10, padx=5)

self.fr_data = Frame(tabelFrame, bd=10)
self.fr_data.pack(fill=BOTH, expand=YES)
self.trvTabel = ttk.Treeview(self.fr_data,
columns=judul_kolom,show='headings')
self.trvTabel.bind("<Double-1>", self.OnDoubleClick)
sbVer = Scrollbar(self.fr_data,
orient='vertical',command=self.trvTabel.yview)
sbVer.pack(side=RIGHT, fill=Y)

self.trvTabel.pack(side=TOP, fill=BOTH)
self.trvTabel.configure(yscrollcommand=sbVer.set)
self.table()

```

```

def table(self):

    con = pymysql.connect(db="db_mahasiswa", user="root", passwd="",
host="localhost", port=3306,autocommit=True)
    cur = con.cursor()
    cur.execute("SELECT * FROM mahasiswa")
    data_table = cur.fetchall()

    for kolom in judul_kolom:
        self.trvTabel.heading(kolom,text=kolom)

    self.trvTabel.column("NIM", width=110,anchor="w")
    self.trvTabel.column("Nama", width=120,anchor="w")
    self.trvTabel.column("Tanggal Lahir", width=100,anchor="w")
    self.trvTabel.column("Alamat", width=160,anchor="w")
    self.trvTabel.column("No. Handphone", width=120,anchor="w")

    i=0
    for dat in data_table:
        if(i%2):
            baris="genap"
        else:
            baris="ganjil"
        self.trvTabel.insert("", 'end', values=dat, tags=baris)
        i+=1

    self.trvTabel.tag_configure("ganjil", background="#FFFFFF")
    self.trvTabel.tag_configure("genap", background="whitesmoke")
    cur.close()
    con.close()

def auto(self):
    con = pymysql.connect(db='db_mahasiswa', user='root', passwd="",
host='localhost', port=3306,autocommit=True)
    cur = con.cursor()

```



```

cuv = con.cursor()
sqlkode = "SELECT max(mahasiswa_kode) FROM mahasiswa"
sql = "SELECT mahasiswa_kode FROM mahasiswa"
cur.execute(sqlkode)
cuv.execute(sql)
maxkode = cur.fetchone()

if cuv.rowcount > 0:
    autohit = int(maxkode[0])+1
    hits = "000"+str(autohit)
    if len(hits) == 4:
        self.entKode.insert(0, hits)
        self.entNama.focus_set()
    elif len(hits) == 5:
        hit = "00"+str(autohit)
        self.entKode.insert(0, hit)
        self.entNama.focus_set()
    elif len(hits) == 6:
        hit = "0"+str(autohit)
        self.entKode.insert(0, hit)
        self.entNama.focus_set()
    elif len(hits) == 7:
        hit = ""+str(autohit)
        self.entKode.insert(0, hit)
        self.entNama.focus_set()

    else:
        messagebox.showwarning(title="Peringatan", \
                                message="maaf lebar data hanya sampai 4 digit")

    else:
        hit = "0001"
        self.entKode.insert(0, hit)
        self.entNama.focus_set()

    self.entKode.config(state="readonly")
def onClose(self, event=None):
    self.parent.destroy()

```

```

def onDelete(self):
    con = pymysql.connect(db='db_mahasiswa', user='root', passwd="",
host='localhost', port=3306,autocommit=True)
    cur = con.cursor()
    self.entKode.config(state="normal")
    cKode = self.entKode.get()
    sql = "DELETE FROM mahasiswa WHERE mahasiswa_kode =%s"
    cur.execute(sql,cKode)
    self.onClear()
    messagebox.showinfo(title="Informasi", \
        message="Data sudah di hapus.")

    cur.close()
    con.close()

def onClear(self):
    self.btnSave.config(state="normal")
    self.btnUpdate.config(state="disable")
    self.btnDelete.config(state="disable")
    self.entKode.config(state="normal")
    self.entKode.delete(0, END)
    self.entNama.delete(0, END)
    self.entHari.delete(0, END)
    self.entBulan.delete(0, END)
    self.entTahun.delete(0, END)
    self.entAlamat.delete('1.0', 'end')
    self.entHp.delete(0, END)
    self.trvTabel.delete(*self.trvTabel.get_children())
    self.fr_data.after(0, self.table())

    self.auto()
    self.entNama.focus_set()

def onSave(self):

    con = pymysql.connect(db='db_mahasiswa', user='root', passwd="",

```

```

host='localhost', port=3306,autocommit=True)

cKode = self.entKode.get()
cNama = self.entNama.get()

####
cHari = self.entHari.get()
cBulan = self.entBulan.get()
cTahun = self.entTahun.get()
dLahir = datetime.date(int(cTahun),int(cBulan),int(cHari))
cAlamat = self.entAlamat.get('1.0', 'end')
cHp = self.entHp.get()
if len(cHari) == 0 and len(cBulan) == 0 and len(cTahun):
    messagebox.showwarning(title="Peringatan",message="Tanggal
Tidak boleh kosong")
else:

    cur = con.cursor()
    sql = "INSERT INTO mahasiswa
(mahasiswa_kode,mahasiswa_nama, mahasiswa_tgl_lahir,
mahasiswa_alamat,mahasiswa_no_hp)"+\
        "VALUES(%s,%s,%s,%s,%s)"
    cur.execute(sql,(cKode,cNama,dLahir,cAlamat,cHp))
    self.onClear()
    messagebox.showinfo(title="Informasi", \
        message="Data sudah di tersimpan.")

    cur.close()
    con.close()

def onUpdate(self):
    cKode = self.entKode.get()

    if len(cKode) == 0:
        messagebox.showwarning(title="Peringatan",message="Kode
kosong.")
        self.entKode.focus_set()

    else:

```

```

        con = pymysql.connect(db='db_mahasiswa', user='root', passwd="",
host="localhost",
                                port=3306, autocommit=True)
        cur = con.cursor()
        cKode = self.entKode.get()
        cNama = self.entNama.get()

        #####
        cHari = self.entHari.get()
        cBulan = self.entBulan.get()
        cTahun = self.entTahun.get()
        dLahir = datetime.date(int(cTahun),int(cBulan),int(cHari))
        cAlamat = self.entAlamat.get('1.0', 'end')
        cHp = self.entHp.get()

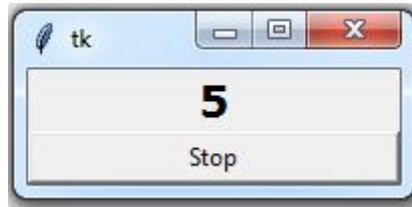
        sql = "UPDATE mahasiswa SET mahasiswa_nama=%s,
mahasiswa_tgl_lahir=%s, mahasiswa_alamat=%s,mahasiswa_no_hp=%s WHERE
mahasiswa_kode =%s"
        cur.execute(sql,(cNama,dLahir,cAlamat,cHp,cKode))
        self.onClear()
        messagebox.showinfo(title="Informasi", \
                                message="Data sudah di terupdate.")

        cur.close()
        con.close()

def main():
    Petugas(root)
    root.mainloop()
main()

```

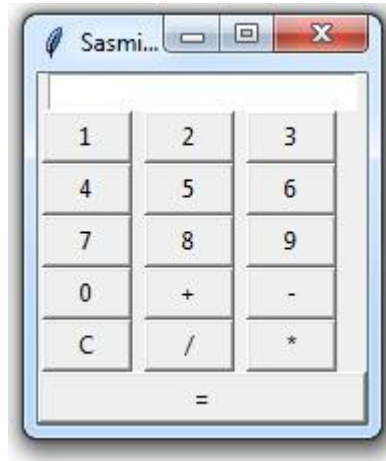
5. Create Stop Watch



Berikut adalah source code **stopwatch.py** :

```
import tkinter as tk
penghitung = 0
def hitung(teks):
    def count():
        global penghitung
        penghitung += 1
        teks.config(text=str(penghitung))
        teks.after(1000, count)
    count()
root = tk.Tk()
teks = tk.Label(root, font = "Verdana 16 bold")
teks.pack()
hitung(teks)
tombol = tk.Button(root, text='Stop', width=25, command=root.destroy)
tombol.pack()
root.mainloop()
```

6. Create Kalkulator



Berikut adalah source code **calculator.py** :

```
from functools import partial
import tkinter as tk

class aplikasiKalkulator(tk.Tk):
    def __init__(self):
        tk.Tk.__init__(self)
        self.title("Kalkulator tkinter")
        self.membuatTombol()
        self.penentu = False

    def membuatTombol(self):
        self.layar = tk.Entry(self, width=25)
        self.layar.grid(row=0, column=0, columnspan=5)

        btn_list = [
            '1', '2', '3',
            '4', '5', '6',
            '7', '8', '9',
            '0', '+', '-',
            'C', '/', '*',
            '='
        ]
```

```

baris = 1
kolom = 0
for penampung in btn_list:
    perintah = partial(self.hitung, penampung)
    if penampung == '=':
        tk.Button(self, text='=', width=22, command=perintah).grid(row=baris,
column=kolom, columnspan=5)
    else :
        tk.Button(self, text=penampung, width=5,
command=perintah).grid(row=baris, column=kolom)
        kolom += 1
        if kolom > 2:
            kolom = 0
            baris += 1

def hitung(self, key):
    if key == '=':
        self.penentu = True
        try:
            result = eval(self.layar.get())
            self.layar.delete(0, tk.END)
            self.layar.insert(tk.END, str(result))
        except:
            self.layar.insert(tk.END, "-> Error!")
    elif key == 'C':
        self.layar.delete(0, tk.END)
    else:
        if self.penentu :
            self.layar.delete(0, tk.END)
            self.penentu = False
        self.layar.insert(tk.END, key)

panggil = aplikasiKalkulator()
panggil.mainloop()

```

7. Build .py to .exe

Perintah : **'python setup.py build'**

Berikut adalah source code **setup.py** :

```
from cx_Freeze import setup, Executable

target = Executable(
    script="bar.py",
    base="Win32GUI",
    compress=False,
    copyDependentFiles=True,
    appendScriptToExe=True,
    appendScriptToLibrary=False,
    icon="jo.ico"
)

setup(
    name="workshop",
    version="1.0",
    description="stt pelita bangsa",
    author="sas",
    options={"build_exe": {"packages": ["tkinter"]}},
    executables=[target]
)
```