

**LAPORAN**  
**PEMROGRAMAN BERBASIS DESKTOP**

“Grid System dan Frame Class”



Oleh :

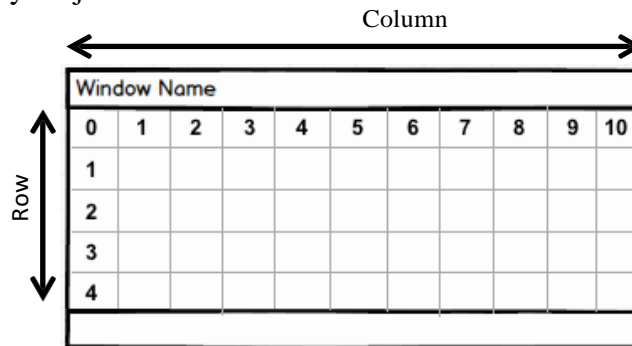
**Ervany Septa Prawara Arisanto**  
**NPM. 193307053**

**JURUSAN TEKNIK**  
**PROGRAM STUDI TEKNOLOGI INFORMASI**  
**POLITEKNIK NEGERI MADIUN**  
**2021**

## I. Dasar Teori

### Grid System

Grid merupakan sebuah konsep dalam melakukan peletakan elemen-elemen di dalam sebuah window. Grid digunakan untuk membantu membuat layout tabel dengan mengatur garis 2 dimensi, yang tersusun atas baris dan kolom. Interaksi antara baris dan kolom disebut sel, dan masing-masing control harus ditempatkan pada satu sel. Grid tidak terlihat secara langsung, oleh karena itu untuk menggunakan grid cukup dengan membayangkannya saja.



Ukuran grid tidak pasti, karena akan menyesuaikan elemen yang ada pada posisi grid tersebut. Untuk menggunakan geometry grid ini menggunakan method *grid()*.

Ada beberapa opsi grid, diantaranya :

#### 1. row, column

Opsi ini digunakan untuk menyimpan posisi widget pada kolom dan baris, nilai di isi mulai dari nol (0).

#### 2. colspan, rowspan

- Opsi colspan ini digunakan untuk melakukan merge pada kolom, misalnya kolom ke-1 pada baris ke-1 di merge dengan kolom ke-2 pada baris ke-1, maka kolom 1 dan 2 akan menyatu menjadi satu kolom.
- Opsi rowspan ini digunakan untuk melakukan merge pada baris, misalnya baris ke-1 di merge dengan baris ke-2, maka baris 1 dan 2 akan menyatu menjadi satu baris.

### Class Frame

Class adalah salah satu cara bagaimana membuat sebuah kode yang mempunyai behaviour tertentu dan lebih mudah dalam mengorganisasi berbagai fungsi dan state-nya. Dalam sebuah class dapat menyimpan sebuah state tanpa harus membuat banyak state bila tidak menggunakan class.

Frame atau bingkai merupakan sebuah container yang berfungsi sebagai wadah meletakkan kontrol-kontrol. Seperti halnya window, frame tidak terlihat dan juga memiliki grid system.

Class Frame sendiri adalah kelas yang bertujuan untuk membuat form, dimana objek dari kelas frame sebagai wadah atau container dari control lain. Untuk membuat class frame, menggunakan keyword **class** diikuti dengan nama class yang diinginkan dengan memberi object **tkinter.Frame** , syntax membuat kelas frame:

```
class namaclass(tkinter.Frame):
```

dan kemudian memberi konstruktor dan parameter pada class frame dengan syntax:

```
def __init__(self, master=None):
```

Kode tersebut menerangkan bahwa SW class yang dibangun merupakan subclass dari tk.Frame. Parameter **master** adalah parent widget dan itu sebuah opsional (defaultnya adalah None) hal tersebut akan melewati instance SW pada class ketika melakukan inisialisasi, **self** adalah pointer ke variabel yang menyimpan instance baru

"\_\_init\_\_" adalah metode yang diperoleh untuk mengembalikan nilai pada kelas python. Ini disebut sebagai konstruktor dalam terminologi berorientasi objek. Metode ini dipanggil ketika sebuah objek dibuat dari sebuah kelas dan memungkinkan kelas untuk menginisialisasi atribut kelas tersebut.

## II. Peralatan

Peralatan praktikum :

1. Laptop
2. Software Visual Studio Code

### III. Hasil Praktikum

Syntax:

a. Syntax Login :

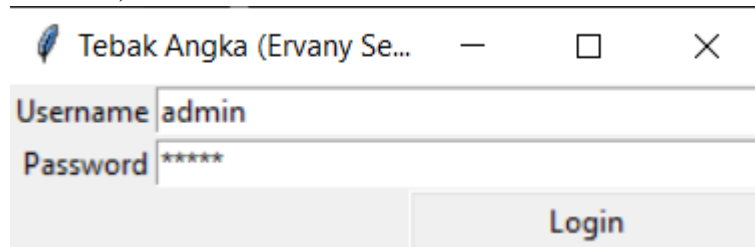
```
Erwany.S.PA-10-4C TugasP3.py X
Erwany Septa Prawara.A-10-4C-Tugas P3 > Code P3 > Erwany.S.PA-10-4C TugasP3.py > ...
1  from tkinter import * #import modul tkinter
2  import tkinter.messagebox #import modul tkinter messagebox
3  from TebakAngka import Tebak, mulai # import lybrary pada modul atau file TebakAngka
4
5  class Login(tkinter.Frame): #membuat class Login dan memanggil Frame pada modul tkinter
6      def __init__(self, master=None):
7          tkinter.Frame.__init__(self, master)
8          self.grid()
9          self.Kontrol()
10
11     def Kontrol(self): #membuat fungsi untuk form login
12         def new_window(self):
13             self.newWindow = Toplevel(self.master)
14             self.app = Tebak(self.newWindow)
15         def akun():
16             # konversi inputan user pada entry username dan password menjadi string
17             user = (str(self.username.get()))
18             pas = (str(self.password.get()))
19
20             #kondisi login jika benar
21             if (user == "admin" and pas == "admin"):
22                 self.newWindow = Toplevel(self.master)
23                 self.app = Tebak(self.newWindow)
24             #kondisi login jika salah
25             else:
26                 tkinter.messagebox.showinfo("Login System", "Username atau Password Salah")
27
28         # membuat label username
29         self.lbl1 = tkinter.Label(self)
30         self.lbl1['text'] = "Username"
31         self.lbl1.grid(row=0, column=0, sticky=tkinter.E) # memberikan fungsi rata kanan dengan atribut sticky=tkinter.E
32
33         # membuat label password
34         self.lbl2 = tkinter.Label(self)
35         self.lbl2['text'] = "Password"
36         self.lbl2.grid(row=1, column=0, sticky=tkinter.E)
37
38         # membuat entry username
39         self.username = tkinter.Entry(self)
40         self.username['width'] = 40
41         self.username.grid(row=0, column=1, columnspan=2)
42
43         # membuat entry password
44         self.password = tkinter.Entry(self)
45         self.password = tkinter.Entry(self, show="*")
46         self.password['width'] = 40
47         self.password.grid(row=1, column=1, columnspan=2)
48
49         # membuat button login
50         self.btn1 = tkinter.Button(self, command= akun)
51         self.btn1['text'] = "Login"
52         self.btn1.grid(row=2, column=2, sticky=tkinter.N + tkinter.E + tkinter.W)
53
54     def main():
55         app = Login()
56         app.master.title("Tebak Angka (Erwany Septa Prawara.A - 193307053)")
57         app.mainloop()
58
59
60 if __name__ == "__main__":
61     main()
```

## b. Syntax TebakAngka:

```
TebakAngka.py X
Ervany Septa Prawara.A-10-4C-Tugas P3 > Code P3 > TebakAngka.py > ...
1  #import modul tkinter
2  import tkinter
3  #import modul tkinter messagebox
4  import tkinter.messagebox
5  #import modul random
6  import random
7
8  class Tebak(tkinter.Frame): #membuat class Tebak
9      def __init__(self, master=None):
10         tkinter.Frame.__init__(self, master)
11         self.grid()
12         self.buttonclick()
13     def buttonclick(self):
14         angka = random.randint(1,100)
15
16         def buttonhasil():
17             #konversi tipe data entry ke int
18             a = int(tbkangka.get())
19
20             # membuat kondisi yang harus dipenuhi dengan if elif else
21             # kondisi tebakan user dengan angka sama, maka program akan menampilkan messagebox "Tebakan user Benar"
22             if a == angka :
23                 tkinter.messagebox.showinfo(
24                     "Hasil", "Tebakan %s, Benar!!!" % (nama.get()))
25             # kondisi tebakan user lebih besar dengan angka, maka program akan menampilkan messagebox "Tebakan user, kurang kecil"
26             elif a > angka :
27                 tkinter.messagebox.showinfo(
28                     "Hasil", "Tebakan %s, Terlalu Besar" % (nama.get()))
29             # kondisi tebakan user lebih besar dengan angka, maka program akan menampilkan messagebox "Tebakan user, kurang besar"
30             else:
31                 tkinter.messagebox.showinfo(
32                     "Hasil", "Tebakan %s, Terlalu Kecil" % (nama.get()))
33
34         # menambah Label "Masukkan Nama Anda"
35         lbl = tkinter.Label(self)
36         lbl['text'] = "Masukkan Nama Anda"
37         lbl.pack()
38
39     # menambah Entry dan ukuran entry untuk memasukkan inputan dari user dan dimasukkan kedalam variable nama
40     nama = tkinter.Entry(self)
41     nama['width'] = 40
42     nama.pack()
43
44     # menambah Label "Tebak angka, masukkan angka 1 sampai 100"
45     lbl2 = tkinter.Label(self)
46     lbl2['text'] = "Tebak angka, masukkan angka 1 sampai 100"
47     lbl2.pack()
48
49     # menambah Entry dan ukuran entry untuk memasukkan inputan dari user untuk angka dan dimasukkan kedalam variabel tbkangka
50     tbkangka = tkinter.Entry(self)
51     tbkangka['width'] = 40
52     tbkangka.pack()
53
54     # menambah button dengan text "Lihat Hasil", jika diclick oleh user akan menjalankan method buttonclick
55     btn = tkinter.Button(self, command=buttonhasil)
56     btn['text'] = "Lihat Hasil"
57     btn.pack()
58
59 def mulai():
60     app2 = Tebak()
61     app2.mainloop()
```


Hasil :

- Hasil setelah program Login dijalankan(saya mengatur username dan password “admin”)



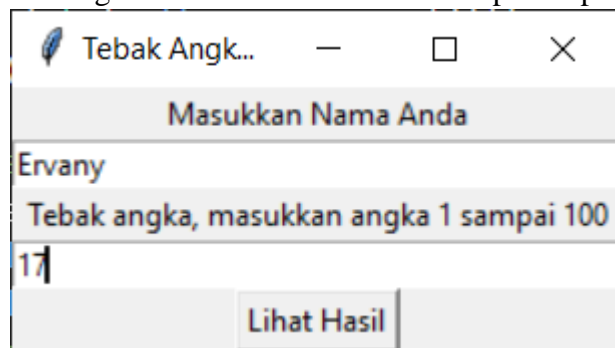
- Jika username dan password salah akan menampilkan pesan berikut

 Login Gagal

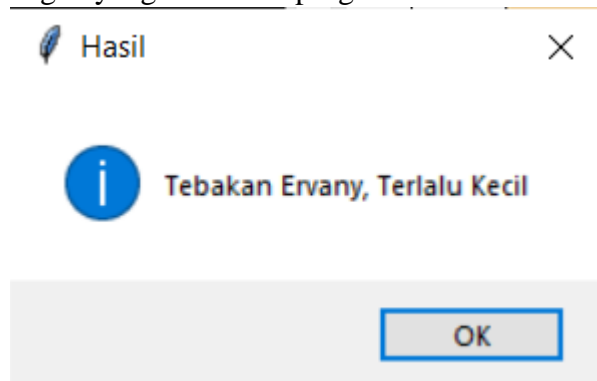
 Username atau Password Salah

OK

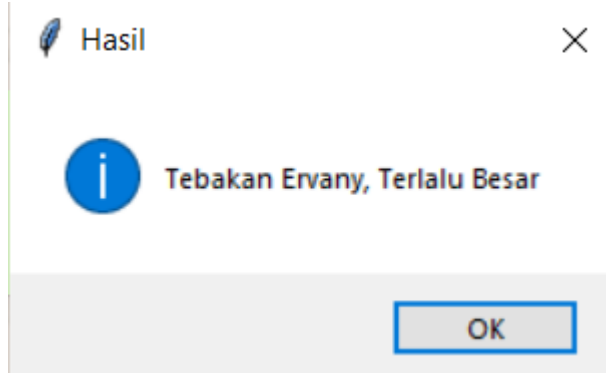
- Jika login berhasil maka akan menampilkan program Tebak Angka



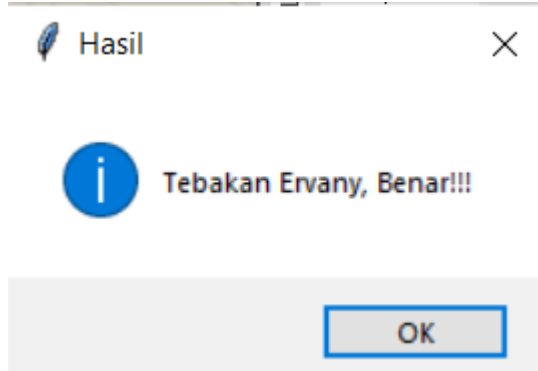
- Hasil setelah button “Lihat Hasil” diclick oleh user dan tebakan user lebih kecil dari angka yang dirandom program



- Hasil setelah button “Lihat Hasil” diclick oleh user dan tebakan user lebih besar dari angka yang dirandom program



- Hasil setelah button “Lihat Hasil” diclick oleh user dan tebakan user sama dari angka yang dirandom program



#### IV. Kesimpulan

Setelah melakukan praktikum Grid System dan Frame Class, kita masih menggunakan modul tkinter untuk membuat form dengan menggunakan Frame dari modul tkinter, dengan membuat class frame terlebih dahulu. Objek dari class Frame sendiri berperan sebagai wadah atau container dari control lain. Kita juga menggunakan geometry manager yaitu grid manager, grid manager disini berperan untuk membantu membuat layout table dengan mengatur garis 2 dimensi, yang tersusun atas baris dan kolom. Interaksi antara baris dan kolom disebut sel, dan masing-masing control harus ditempatkan pada satu sel.

## V. Daftar Pustaka

Fajar, Muhammad Syaeful. 2021. Teori Grid System dan Button Pada TKinter dan Teori Menggunakan Class Frame.

Eksplorasi Antarmuka Grafis Pemakaian Tkinter pada Lingkungan Bahasa Python.  
[http://repository.unpas.ac.id/28589/9/\\_III%20-%20BAB%20III%20EXPLORASI.pdf](http://repository.unpas.ac.id/28589/9/_III%20-%20BAB%20III%20EXPLORASI.pdf)

Membuat Entry Password Menggunakan TKinter.  
<https://stackoverflow.com/questions/2416486/how-to-create-a-password-entry-field-using-tkinter>

Agus, Thami Rusdi. 2020. Memahami Konsep Grid Pada Tkinter(Bagian #1).  
[https://www.academia.edu/20236326/Memahami\\_Konsep\\_Grid\\_Pada\\_Tkinter?auto=download](https://www.academia.edu/20236326/Memahami_Konsep_Grid_Pada_Tkinter?auto=download)

Apa yang \_\_init\_\_ dan self lakukan pada Python?.  
<https://qastack.id/programming/625083/what-init-and-self-do-on-python>

Membuat Class di Python. <https://www.codepolitan.com/membuat-class-di-python-589528b4d558d>