

<b>Nama</b>	MOHAMAD MALIK FAJAR BAIHAQI
<b>NPM</b>	254311011
<b>Kelas</b>	TRPL/1A

## KEMAMPUAN AKHIR YANG DIRENCANAKAN

*Mahasiswa mampu memahami seleksi dalam mengambil suatu keputusan dari beberapa kondisi serta penerapannya dalam pemrograman*

## INDIKATOR

1. Mampu memahami penggunaan perintah kondisional dengan banyak kondisi
2. Mampu menerapkan perintah conditional statement

## PRAKTIKUM

Dengan menggunakan statemen if, buatlah program untuk kasus:

1. Menentukan bonus/hadiah pembelian, jika total pembelian sudah mencapai 100.000, maka pembeli akan mendapatkan bonus/hadiah
2. Menentukan operasi aritmatika dari dua nilai yang diinputkan user kemudian menampilkan hasilnya. Operasi aritmatika tersebut meliputi, penjumlahan, pengurangan, perkalian, dan pembagian.
3. Membuat aplikasi *vending machine* penjualan minuman. Aplikasi harus dapat memberikan uang kembalian jika nilai koin yang dimasukkan user melebihi dari harga minuman yang dipilih. Pemilihan koin dan minuman ditentukan dari aplikasi.

**Nomor 1 : Menentukan bonus/hadiah pembelian, jika total pembelian sudah mencapai 100.000, maka pembeli akan mendapatkan bonus/hadiah**

🔗 Nomor 1\_Hadiah Pembelian.py > ...

```
1  import tkinter as gui
2
3  # Variabel, list, dan dictionary
4  daftarPembelian = []
5  Button = [
6      'RTX 5090',
7      'Mouse',
8      'SanDisk 64 GB',
9      'Keyboard',
10     'Ethernet-to-USB',
11     'Charger 12V'
12 ]
13
14  daftarHarga = {
15     'RTX 5090': 60000000,
16     'Mouse': 30000,
17     'SanDisk 64 GB': 70000,
18     'Keyboard': 50000,
19     'Ethernet-to-USB': 60000,
20     'Charger 12V': 65000,
21 }
22
23  # Fungsi untuk menambah daftarPembelian dan memperbarui checkout
24  def beli(item):
25     daftarPembelian.append((item, daftarHarga[item]))
26     update_checkout()
```

```
27
28  # Fungsi untuk memperbarui tampilan checkout
29  def update_checkout():
30     if not daftarPembelian:
31         checkout_text.set("Belum ada produk yang dibeli.")
32     else:
33         teks = "Produk yang dibeli:\n"
34         for i, (nama, harga) in enumerate(daftarPembelian, start=1):
35             teks += f"{i}. {nama} - Rp {harga:,.0f}\n"
36         checkout_text.set(teks)
37
38  # Fungsi untuk menghitung total dan hadiah diskon
39  def hitungTotal():
40     total = sum(harga for _, harga in daftarPembelian)
41     if total >= 100000:
42         reward = "🎉 Selamat! Anda mendapat diskon 10%!"
43         totalAkhir = total * 0.9
44     else:
45         reward = "Belanja lagi hingga total 100.000 untuk dapat diskon!"
46         totalAkhir = total
47
48     label_total.config(text=f"Harga Asli: Rp {total:,.0f}\nTotal Pembayaran: Rp {totalAkhir:,.0f}")
49     label_reward.config(text=reward)
```

```

50
51 # Root Window
52 root = gui.Tk()
53 root.title("Bonus Pembelian")
54 root.configure(background=█ '#292b2c')
55
56 # Judul
57 judul = gui.Label(root, text="Pilih Produk yang Ingin Dibeli", font=('Arial', 16, 'bold'),
58 | | | | | bg=█ '#292b2c', fg='white', pady=10)
59 judul.grid(row=0, column=0, columnspan=3)
60
61 # Membuat tombol-tombol barang
62 row, col = 1, 0
63 for nama in Button:
64     harga = daftarHarga[nama]
65     teks_tombol = f"{nama}\nRp {harga:,.0f}"
66     gui.Button(root, text=teks_tombol, width=18, height=4, font=('Arial', 13),
67 | | | | | bg=█ '#343a40', fg='white', command=lambda n=nama: beli(n)).grid(row=row, column=col, padx=10, pady=10)
68     col += 1
69     if col > 2:
70         col = 0
71         row += 1
72
73 # Area checkout
74 checkout_text = gui.StringVar()
75 checkout_text.set("Belum ada produk yang dibeli.")
76 label_checkout = gui.Label(root, textvariable=checkout_text, font=('Arial', 13),
77 | | | | | bg=█ '#292b2c', fg=█ '#f7f7f7', justify='left', anchor='w')
78 label_checkout.grid(row=row+1, column=0, columnspan=3, sticky='w', padx=10)
79

```

```

80 # Tombol hitung total
81 gui.Button(root, text="Hitung Total", font=('Arial', 14), bg=█ '#0275d8', fg='white',
82 | | | | | command=hitungTotal).grid(row=row+2, column=0, columnspan=3, pady=10)
83
84 # Label hasil
85 label_total = gui.Label(root, text="Total Pembayaran: Rp 0", font=('Arial', 16), bg=█ '#292b2c', fg='white')
86 label_total.grid(row=row+3, column=0, columnspan=3)
87
88 label_reward = gui.Label(root, text="", font=('Arial', 14), bg=█ '#292b2c', fg=█ '#5bc0de')
89 label_reward.grid(row=row+4, column=0, columnspan=3)
90
91 root.mainloop()
92

```

Hasil :

Bonus Pembelian

### Pilih Produk yang Ingin Dibeli

RTX 5090 Rp 60,000,000	Mouse Rp 30,000	SanDisk 64 GB Rp 70,000
Keyboard Rp 50,000	Ethernet-to-USB Rp 60,000	Charger 12V Rp 65,000

Produk yang dibeli:

1. SanDisk 64 GB - Rp 70,000
2. Ethernet-to-USB - Rp 60,000

Hitung Total

Harga Asli: Rp 130,000  
Total Pembayaran: Rp 117,000

 Selamat! Anda mendapat diskon 10%!

**Nomor 2 : Menentukan operasi aritmatika dari dua nilai yang diinputkan user kemudian menampilkan hasilnya. Operasi aritmatika tersebut meliputi, penjumlahan, pengurangan, perkalian, dan pembagian.**

🔗 Nomor 2\_Calculator.py > ...

```
1  import tkinter as gui
2
3
4
5  #fungsi untuk kalkulasi aritmatika
6  def calculate():
7      print('calculate debug',hasil.get())
8      try :
9          expression = Entry.get()
10         value=eval(expression)
11         hasil.set(str(value))
12         if isinstance(value, float) and value.is_integer():
13             value = int(value)
14             hasil.set(str(value))
15     except ZeroDivisionError:
16         hasil.set('0')
17     except Exception :
18         hasil.set("Input tidak valid")
19
20 def tombolClear():
21     Entry.delete(0,gui.END)
22     hasil.set("hasil")
23
24 def tombolBackspace():
25     current = Entry.get()
26     if current :
27         Entry.delete(len(current)-1,gui.END)
28
```

```

29
30 #deklaras root
31 root = gui.Tk()
32 root.title("Kalkulator aritmatika basic")
33 root.configure(background=□ '#292b2c')
34
35 #kolom input
36 Entry = gui.Entry(root,width=20,font=("Arial",20),justify='right',relief='ridge',background='white')
37 Entry.grid(row=0,column=0,columnspan=4,padx=10,pady=10)
38
39 #kolom hasil
40 hasil = gui.StringVar(value='hasil')
41 label_hasil = gui.Label(root, textvariable=hasil,font=('Arial',18),anchor='e',background='white',
42 | | | | | | | | relief='sunken',width=20)
43 label_hasil.grid(row=1,column=0,columnspan=4,padx=10,pady=10)
44
45
46 #list/array Button dalam bentuk matrix
47 Button = [
48 | '7','8','9','/',
49 | '4','5','6','*',
50 | '1','2','3','- ',
51 | '0','.','=','+'
52 ]
53

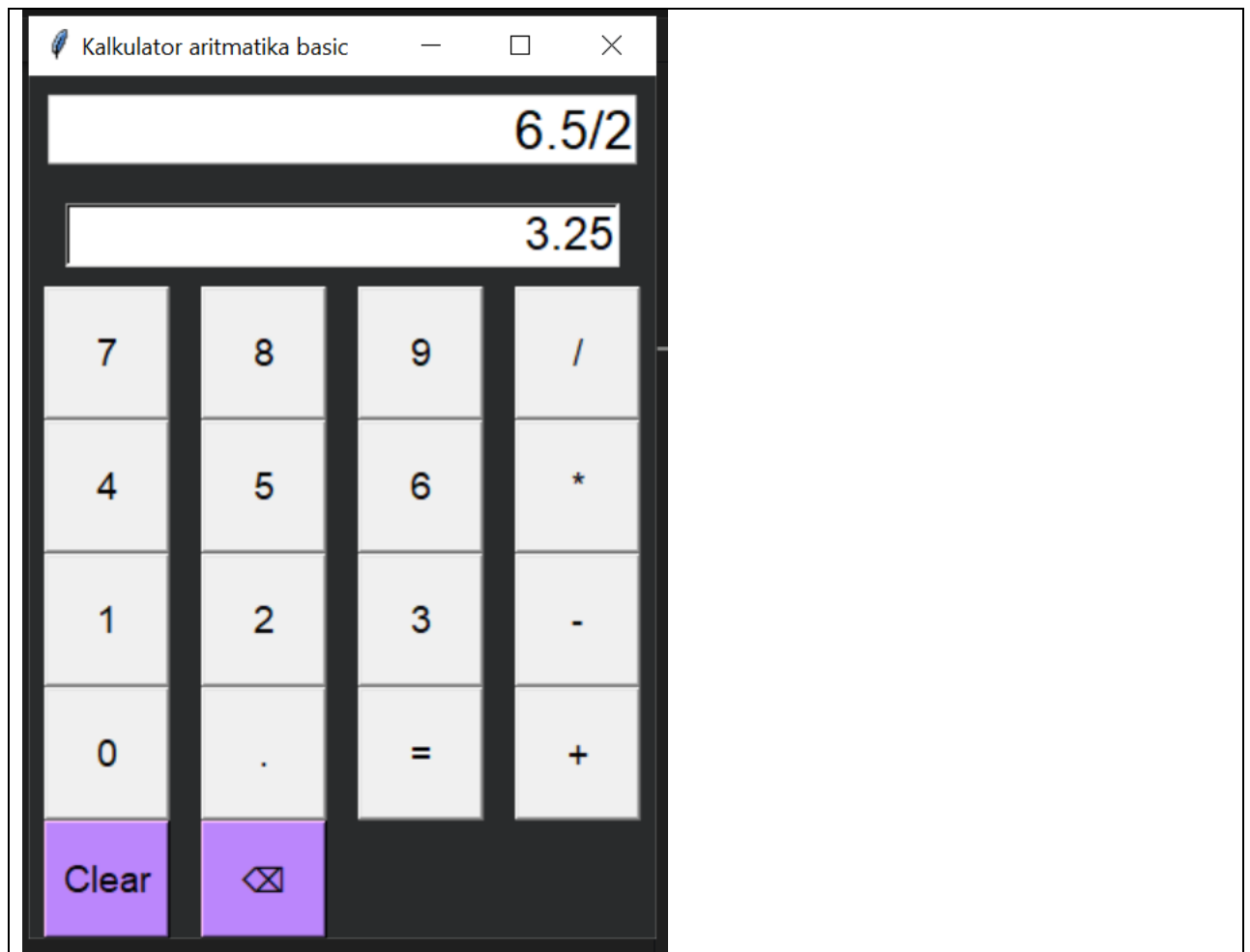
```

```

54
55 #loop untuk generate tombol
56 row,col = 2,0
57 for i in Button:
58 | if i == '=':
59 | | cmd = calculate #pass fungsi calculate ke tombol '='
60 | else :
61 | | cmd = lambda x=i: Entry.insert(gui.END, x)
62 | gui.Button(root,text=i,width=5,height=2,font=('Arial',14),pady=5,command=cmd).grid(row=row,column=col)
63
64 | col += 1
65 | if col>3:
66 | | col = 0
67 | | row += 1
68
69 #tombol clear
70 gui.Button(root,text='Clear',width=5,height=2,font=('Arial',14),background=□ '#BB86FC',
71 | | | | | command=tombolClear).grid(row=row,column=0)
72
73 #tombol backspace
74 gui.Button(root,text='⌫',width=5,height=2,font=('Arial',14),background=□ '#BB86FC',
75 | | | | | command=tombolBackspace).grid(row=row,column=1)
76
77
78 root.mainloop()
79

```

Hasil :



**Nomor 3 : Membuat aplikasi *vending machine* penjualan minuman. Aplikasi harus dapat memberikan uang kembalian jika nilai koin yang dimasukkan user melebihi dari harga minuman yang dipilih. Pemilihan koin dan minuman ditentukan dari aplikasi.**

```
Nomor 3_Vending Machine.py > ...
1  import tkinter as tk
2  from tkinter import messagebox
3
4  window = tk.Tk()
5  window.title("Vending Machine Penjualan Minuman")
6  window.configure(bg="#292b2c")
7
8  # Data produk dan harga
9  produk = {
10     "Teh Botol": 5000,
11     "Air Mineral": 3000,
12     "Kopi Hitam": 7000,
13     "Susu Coklat": 8000,
14     "Jus Jeruk": 9000
15 }
16
17 # Pecahan uang yang diterima
18 pecahan_uang = [1000, 2000, 5000, 10000, 20000, 50000, 100000]
19
20 # Variabel global
21 total_harga = 0
22 total_uang = 0
23 keranjang = []
24
25 # FRAME utama
26 frame_produk = tk.Frame(window, bg="#292b2c")
27 frame_produk.pack(pady=10)
28
29 frame_uang = tk.Frame(window, bg="#292b2c")
30 frame_uang.pack(pady=10)
```



```

30 frame_uang.pack(pady=10)
31
32 # FRAME untuk log pembelian (dengan scrollbar)
33 frame_log = tk.Frame(window, bg="#292b2c")
34 frame_log.pack(pady=10)
35
36 log_text = tk.Text(frame_log, width=50, height=10, bg="#1c1e1f", fg="white", state="disabled", wrap="word")
37 log_text.pack(side=tk.LEFT, fill=tk.BOTH)
38
39 scrollbar = tk.Scrollbar(frame_log, command=log_text.yview)
40 scrollbar.pack(side=tk.RIGHT, fill=tk.Y)
41 log_text.config(yscrollcommand=scrollbar.set)
42
43 # Fungsi bantu
44 def log(msg):
45     """Menambahkan teks ke area log tanpa memperbesar window"""
46     log_text.config(state="normal")
47     log_text.insert(tk.END, msg + "\n")
48     log_text.config(state="disabled")
49     log_text.see(tk.END)
50
51 def pilih_produk(nama):
52     global total_harga
53     harga = produk[nama]
54     keranjang.append((nama, harga))
55     total_harga += harga
56     log(f"Produk dipilih: {nama} - Rp{harga:,}")

```

```

57
58 def masukkan_uang(nilai):
59     global total_uang
60     total_uang += nilai
61     log(f"Uang dimasukkan: Rp{nilai:,}")
62
63 def Checkout():
64     if not keranjang:
65         messagebox.showinfo("Info", "Belum ada produk yang dipilih.")
66         return
67     log("\n=== Checkout ===")
68     for nama, harga in keranjang:
69         log(f"{nama} - Rp{harga:,}")
70     log(f"Total harga: Rp{total_harga:,}")
71     log(f"Total uang: Rp{total_uang:,}")

```

```

72
73 def Submit():
74     global total_harga, total_uang, keranjang
75     if total_uang == 0:
76         messagebox.showwarning("Peringatan", "Masukkan uang terlebih dahulu!")
77         return
78     if not keranjang:
79         messagebox.showwarning("Peringatan", "Pilih minuman terlebih dahulu!")
80         return
81
82     if total_uang >= total_harga:
83         kembalian = total_uang - total_harga
84         messagebox.showinfo("Transaksi Sukses", f"Pembelian berhasil!\nKembalian Anda: Rp{kembalian:,}")
85         log(f"Transaksi selesai. Kembalian: Rp{kembalian:,}\n")
86     else:
87         kekurangan = total_harga - total_uang
88         messagebox.showwarning("Uang Kurang", f"Uang Anda kurang Rp{kekurangan:,}!")
89
90 def reset():
91     global total_harga, total_uang, keranjang
92     total_harga = 0
93     total_uang = 0
94     keranjang.clear()
95     log_text.config(state="normal")
96     log_text.delete(1.0, tk.END)
97     log_text.config(state="disabled")
98     log("Aplikasi telah direset.\n")

```

```

99
100 # Tombol produk
101 tk.Label(frame_produk, text="Pilih Produk:", bg="black", fg="white").pack()
102 for nama, harga in produk.items():
103     tk.Button(frame_produk, text=f"{nama} (Rp{harga:,})", command=lambda n=nama: pilih_produk(n),
104              bg="black", fg="white", width=25).pack(pady=2)
105
106 # Tombol pecahan uang
107 tk.Label(frame_uang, text="Masukkan Uang:", bg="black", fg="white").pack()
108 for uang in pecahan_uang:
109     tk.Button(frame_uang, text=f"Rp{uang:,}", command=lambda u=uang: masukkan_uang(u),
110              bg="black", fg="white", width=15).pack(pady=1)
111
112 # Tombol kontrol
113 tk.Button(window, text="Checkout", command=Checkout, bg="red", fg="black", width=15).pack(pady=0)
114 tk.Button(window, text="Submit", command=Submit, bg="blue", fg="white", width=15).pack(pady=0)
115 tk.Button(window, text="Reset", command=reset, bg="green", fg="white", width=15).pack(pady=0)
116
117 window.mainloop()
118

```

Hasil :

