



Laporan Praktikum Algoritma & Pemrograman

Semester Genap 2024/2025

SAYA MENYATAKAN BAHWA LAPORAN PRAKTIKUM INI SAYA BUAT DENGAN USAHA SENDIRI TANPA MENGGUNAKAN BANTUAN ORANG LAIN. SEMUA MATERI YANG SAYA AMBIL DARI SUMBER LAIN SUDAH SAYA CANTUMKAN SUMBERNYA DAN TELAH SAYA TULIS ULANG DENGAN BAHASA SAYA SENDIRI.

SAYA SANGGUP MENERIMA SANKSI JIKA MELAKUKAN KEGIATAN PLAGIASI, TERMASUK SANKSI TIDAK LULUS MATA KULIAH INI.

NIM	<71231046>
Nama Lengkap	<FREIRE HANAN PUTRA>
Minggu ke / Materi	04 / STRUKTUR KONTROL PERCABANGAN

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
YOGYAKARTA
2025

BAGIAN 1: MATERI MINGGU INI (40%)

Pada minggu ini saya mempelajari materi tentang struktur control dan percabangan. Materi-materi tersebut saya tuliskan sebagai berikut;

MATERI 1 Boolean Expression dan Logical Operator

Boolean merupakan tipe data yang menerapkan prinsip true atau false. Misalkan hanan = laki laki yang diminta hanan adalah Perempuan maka output yang muncul adalah false. Karena Boolean memiliki hasil hanya dua kemungkinan yakni true or false tergantung pada isi dari sebuah variabel. Misalkan seperti kode dibawah ini:

```
Microsoft Windows [Version 10.0.22621.4317]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Freire Hanan>python
Python 3.12.9 (tags/v3.12.9:fdb8142, Feb  4 2025, 15:27:58) [MSC v.1942 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> pembelian = 12000
>>> pembelian <= 1000
False
>>> pembelian = 12000
>>> pembelian >= 1000
True
>>> |
```

Operator	Keterangan
<code>X == Y</code>	Apakah X sama dengan Y
<code>X != Y</code>	Apakah X tidak sama dengan Y
<code>X > Y</code>	Apakah X lebih dari Y
<code>X >= Y</code>	Apakah X lebih dari sama dengan Y
<code>X < Y</code>	Apakah X kurang dari Y
<code>X <= Y</code>	Apakah X kurang dari sama dengan Y
<code>X is Y</code>	Apakah X sama dengan Y
<code>X is not Y</code>	Apakah X tidak sama dengan Y

Untuk menerapkan tipe data boolean expression dan harus memilih operator yang tepat tergantung masalah yang harus diselesaikan.

- Bentuk boolean hanya ada dua yaitu true dan false.
- Harus memperhatikan kata-kata khusus seperti minimum, maksimum, tidak lebih dari, tidak kurang dari, tidak sama, tidak berbeda.
- Memperhatikan dengan seksama setelah itu tentukan variable yang sesuai dan mudah dipahami sesuai permasalahan yang akan diselesaikan.

Adapun contoh masalah yang dapat diselesaikan dengan tipe data boolean sebagai berikut;

Tabel 3.2: Operator-operator perbandingan (comparison).

Contoh masalah	Boolean expression
Untuk lulus dibutuhkan IPK minimum 2.25	<code>ipk >= 2.25</code>
Golden Button hanya diberikan untuk Youtuber dengan subscriber lebih dari 1 juta	<code>subscriber > 1000000</code>
Pengendara dengan kecepatan lebih dari 90 km/jam akan mendapatkan tilang	<code>kecepatan > 90</code>
Wahana Rollercoaster hanya bisa dinaiki oleh mereka yang tinggi badannya lebih dari 110 cm	<code>tinggi > 110</code>
Nilai ujian Hanna adalah 75 sedangkan Robby mendapatkan nilai 75. Apakah nilai keduanya sama?	<code>hanna is robbby</code>
Junaedi memiliki 10 sepatu, Ricky punya 15 sepatu dan Arnold punya 20 sepatu. Apakah gabungan sepatu Junaedi dan Ricky lebih banyak dari sepatu milik Arnold?	<code>junaedi + ricky > arnold</code>

Sumber: dari Modul

MATERI 2 Bentuk Bentuk Percabangan

Dalam Bahasa pemrograman python ada tiga bentuk percabangan yaitu conditional, alternative dan chained conditional. Adapun contoh percabangan adalah sebagai berikut;

Berikut ini merupakan contoh percabangan conditional.

```
if <kondisi>:  
    <lakukan ini>  
    <lakukan ini>  
    ...  
else:  
    <lakukan itu>  
    <lakukan itu>
```

Sedangkan dibawah ini merupakan contoh percabangan chained conditional

```
Contoh 3.py > ...
1  bilangan = int(input("Masukkan suatu bilangan: "))
2  if bilangan > 0:
3      print("Positif")
4  elif bilangan < 0:
5      print("Negatif")
6  elif bilangan == 0:
7      print("Nol")
8
```

Contoh lain dari bentuk percabangan adalah seperti pada kode dibawah ini:

```
contoh.py > ...
1  nilai = int(input('Masukan nilai anda: '))
2
3  if nilai > 70:
4      print("Anda Lulus")
5  else:
6      print("Anda Tidak Lulus")
```

PROBLEMS DEBUG CONSOLE OUTPUT PORTS TERMINAL

Microsoft Windows [Version 10.0.22621.4317]
(c) Microsoft Corporation. All rights reserved.

D:\Semester 4\PrakAlpro 2425\week3>"C:/Users/Freire Hanan/AppData/Local/Programs/Python/Python312/python.exe" "d:/Semester 4/PrakAlpro 2425/week3/contoh.py"

Masukan nilai anda: 80
Anda Lulus

D:\Semester 4\PrakAlpro 2425\week3>

MATERI 3 Penanganan Kesalahan Input Menggunakan Exception Handling

Dalam menangani input dari pengguna, kita juga harus memperhatikan potensi kesalahan yang mungkin terjadi agar program dapat berjalan dengan benar. Sebagai contoh, pertimbangkan sebuah program yang meminta pengguna untuk memasukkan usianya, kemudian program akan menampilkan kategori usia pengguna, seperti lansia, dewasa, remaja, kanak-kanak, atau balita, berdasarkan aturan tertentu. Cara untuk menangani kesalahan input adalah dengan menuliskan try dan except. Penerapannya seperti pada code dibawah ini:

```
contoh.py > ...
7
8  usia = int(input("Masukan Usia Anda: "))
9  if usia <= 5:
10     print("Balita")
11  elif usia >= 6 and usia <= 11:
12     print("Kanak-Kanak")
13  elif usia >= 12 and usia <= 25:
14     print("Remaja")
15  elif usia <= 26 and usia <= 45:
16     print("Dewasa")
17  elif usia >= 45:
18     print("Lansia")
19

PROBLEMS  DEBUG CONSOLE  OUTPUT  PORTS  TERMINAL

D:\Semester 4\PrakAlpro 2425\week3> "C:\Users\Freire Hanan\AppData\Local\Programs\Python\Python312\python.exe" "d:\Semester 4\PrakAlpro 2425\week3\contoh.py"
Masukan Usia Anda: dua belas
Traceback (most recent call last):
  File "d:\Semester 4\PrakAlpro 2425\week3\contoh.py", line 8, in <module>
    usia = int(input("Masukan Usia Anda: "))
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
ValueError: invalid literal for int() with base 10: 'dua belas'

D:\Semester 4\PrakAlpro 2425\week3> |
```

Seperti yang telah dibahas pada bab sebelumnya, fungsi `input()` digunakan untuk menerima masukan atau input dari pengguna. Fungsi `input()` akan menghasilkan output berupa string (teks). Namun, karena usia seharusnya berupa angka atau bilangan, kita perlu mengonversi string tersebut menjadi bilangan bulat menggunakan fungsi `int()`.

Program ini akan berjalan dengan baik selama pengguna memasukkan input yang sesuai, yaitu angka. Namun, jika pengguna memasukkan input yang tidak menunjukkan situasi di mana pengguna memasukkan input yang tidak sesuai dengan yang diharapkan.

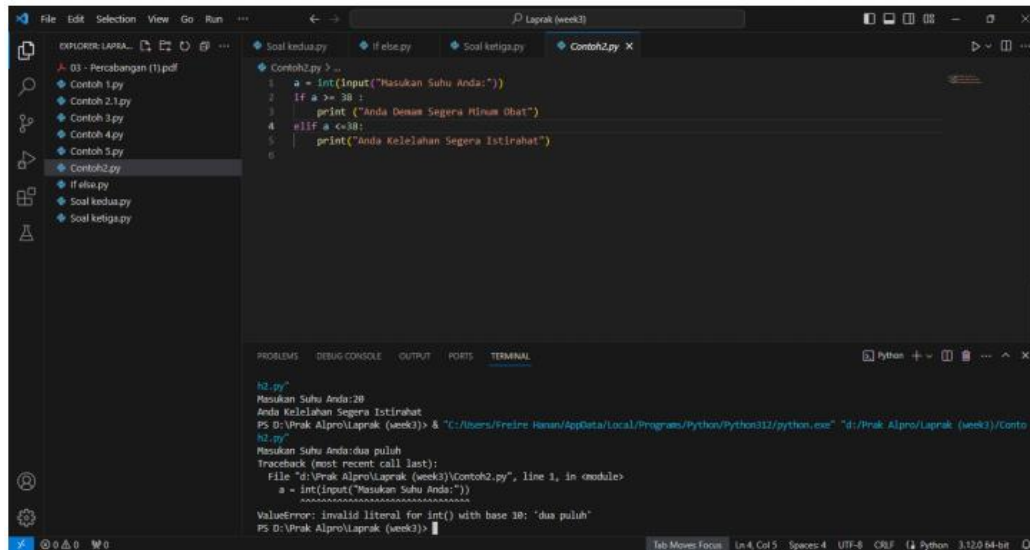
```
File  Edit  Selection  View  Go  Run  ...  Laprak (week3)
EXPLORER LAPRAK...  03 - Percabangan (11).pdf
  Contoh 1.py
  Contoh 2.1.py
  Contoh 3.py
  Contoh 4.py
  Contoh 5.py
  Contoh2.py
  if else.py
  Soal kedua.py
  Soal ketiga.py

  Contoh 1.py > ...
1  yang_benar = input("Masukan Suhu Anda")
2  try:
3      a = int(yang_benar)
4      if a >= 38.0:
5          print("Anda Demam Segera Minum Obat")
6      elif a <= 38:
7          print("Anda Kelelahan Segera Istirahat")
8  except:
9      print("Input Anda Salah ")

PROBLEMS  DEBUG CONSOLE  OUTPUT  PORTS  TERMINAL

PS D:\Vrak Alpro\Laprak (week3)> & "C:\Users\Freire Hanan\AppData\Local\Programs\Python\Python312\python.exe" "d:\Prak Alpro\Laprak (week3)\Conto
h.1.py"
Masukan Suhu Anda21
Anda Kelelahan Segera Istirahat
PS D:\Vrak Alpro\Laprak (week3)> & "C:\Users\Freire Hanan\AppData\Local\Programs\Python\Python312\python.exe" "d:\Prak Alpro\Laprak (week3)\Conto
h.1.py"
Masukan Suhu Anda200
Input Anda Salah
PS D:\Vrak Alpro\Laprak (week3)> |
```

Gambar diatas menggunakan kode try except untuk menangani kesalahan input. Sedangkan jika tidak menggunakan kesalahan input akan seperti ini;



The screenshot shows a Python IDE with a file explorer on the left containing files like 'Contoh1.py', 'Contoh2.py', 'Contoh3.py', 'Contoh4.py', 'Contoh5.py', 'Soal kedua.py', 'Soal ketiga.py', and 'Soal keempat.py'. The main editor window displays the code for 'Contoh2.py':

```
1 a = int(input("Masukan Suhu Anda:"))
2 if a >= 38:
3     print("Anda Demam Segera Minum Obat")
4 elif a <= 38:
5     print("Anda Kelelahan Segera Istirahat")
6
```

The bottom panel shows the 'TERMINAL' output:

```
h2.py
Masukan Suhu Anda:38
Anda Kelelahan Segera Istirahat
PS D:\Prak Alpro\Laparak (week3) > "C:/Users/Freire/Ansan/AppData/Local/Programs/Python/Python311/python.exe" "d:/Prak Alpro/Laparak (week3)/Conto
h2.py"
Masukan Suhu Anda:dua puluh
Traceback (most recent call last):
  File "d:\Prak Alpro\Laparak (week3)\Contoh2.py", line 1, in <module>
    a = int(input("Masukan Suhu Anda:"))
          ~~~~~^~~~~
ValueError: invalid literal for int() with base 10: 'dua puluh'
PS D:\Prak Alpro\Laparak (week3) >
```

Gambar diatas jika tidak menggunakan kode try except.

BAGIAN 2: LATIHAN MANDIRI (60%)

Link Github: https://github.com/Freirehnn23/prak_alpro_week4

SOAL 1

Code:

```
jupyter FREIRE HANAN PUTRA (71231046) Last Checkpoint: 2 minutes ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help

[Icons] [Run] [Code]

In [2]: #bagian 1

try:
    suhu = int(input("Masukkan suhu tubuh: "))
    if suhu >= 38:
        print("Anda demam")
    else:
        print("Anda tidak demam")
except:
    print("Format inputan anda salah, mohon input bertipe integer(angka)")

Masukkan suhu tubuh: 23
Anda tidak demam
```

1.

```
jupyter FREIRE HANAN PUTRA (71231046) Last Checkpoint: 5 minutes ago (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help

[Icons] [Run] [Code]

In [4]: #bagian 2

try:
    bilangan = int(input("Masukkan suatu bilangan: "))
    if bilangan > 0:
        print("Positif")
    elif bilangan < 0:
        print("Negatif")
    elif bilangan == 0:
        print("Nol")
except:
    print("Format inputan anda salah, mohon input bertipe integer(angka)")
```

2.

```
elif bilangan == 0:
    print("No!")
except:
    print("Format inputan anda salah, mohon input bertipe integer(angka)")
```

Masukkan suatu bilangan: sdf
Format inputan anda salah, mohon input bertipe integer(angka)

In [6]: # bagian 3

```
try:
    a = int(input("Masukkan bilangan pertama: "))
    b = int(input("Masukkan bilangan kedua: "))
    c = int(input("Masukkan bilangan ketiga: "))
    if a > b and a > c:
        print("Terbesar: ", a)
    elif b > a and b > c:
        print("Terbesar: ", b)
    elif c > a and c > b:
        print("Terbesar: ", c)
except:
    print("Format inputan anda salah, mohon input bertipe integer(angka)")
```

3.

Penerapan:

Bagian 1

Masukkan suhu tubuh: 23
Anda tidak demam

1.

Masukkan suhu tubuh: asd
Format inputan anda salah, mohon input bertipe integer(angka)

2.

Bagian 2

Masukkan suatu bilangan: 32
Positif

1.

Masukkan suatu bilangan: sdf
Format inputan anda salah, mohon input bertipe integer(angka)

2.

Bagian 3

Masukkan bilangan pertama: wq
Format inputan anda salah, mohon input bertipe integer(angka)

1.

Masukkan bilangan pertama: 12
Masukkan bilangan kedua: 32
Masukkan bilangan ketiga: 43
Terbesar: 43

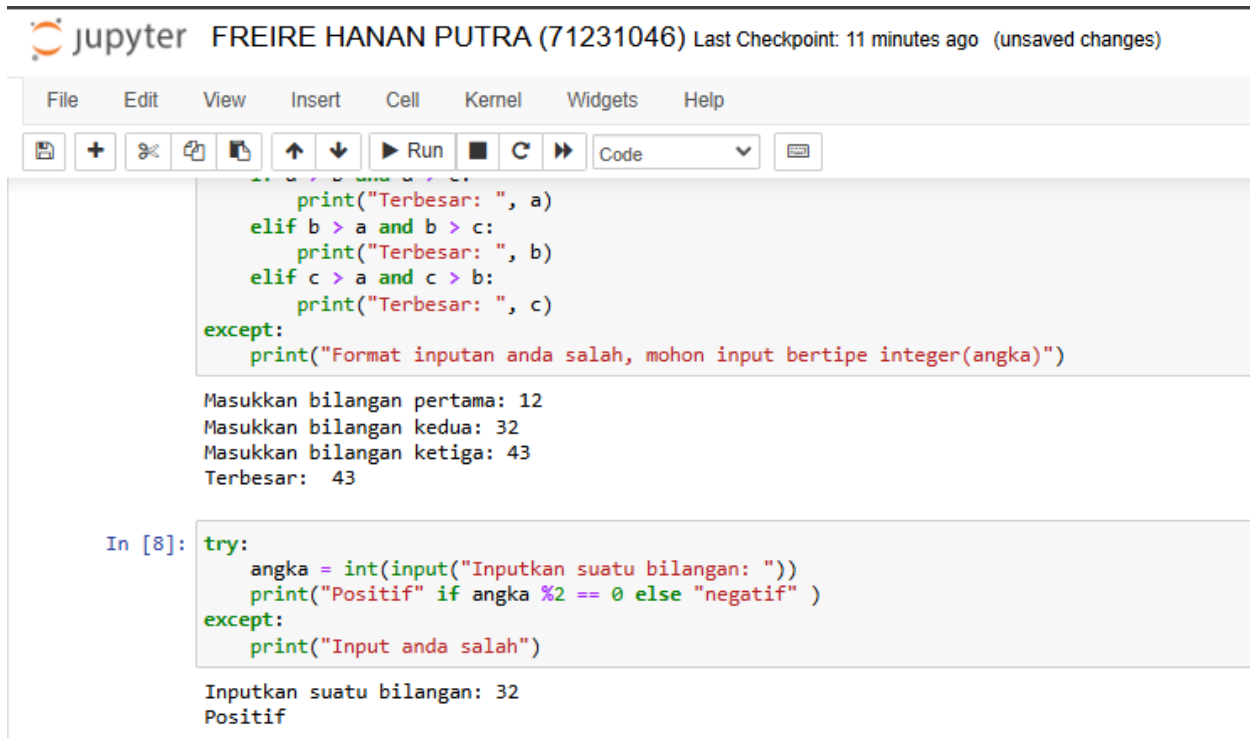
2.

Penjelasan:

Secara garis besar penggunaan try dan except itu dapat digunakan untuk menspesifikan tipe data dari inputan oleh user. Penggunaan nya adalah dengan cara menambahkan main program pada try setelah itu jika tipe data yang tidak sesuai dengan main akan diprogram pada bagian except.

SOAL 2

Code:



```
jupyter FREIRE HANAN PUTRA (71231046) Last Checkpoint: 11 minutes ago (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help

Masukkan bilangan pertama: 12
Masukkan bilangan kedua: 32
Masukkan bilangan ketiga: 43
Terbesar: 43

In [8]: try:
        angka = int(input("Inputkan suatu bilangan: "))
        print("Positif" if angka % 2 == 0 else "negatif" )
      except:
        print("Input anda salah")

Inputkan suatu bilangan: 32
Positif
```

Penerapan:

```
Inputkan suatu bilangan: wq
Input anda salah
```

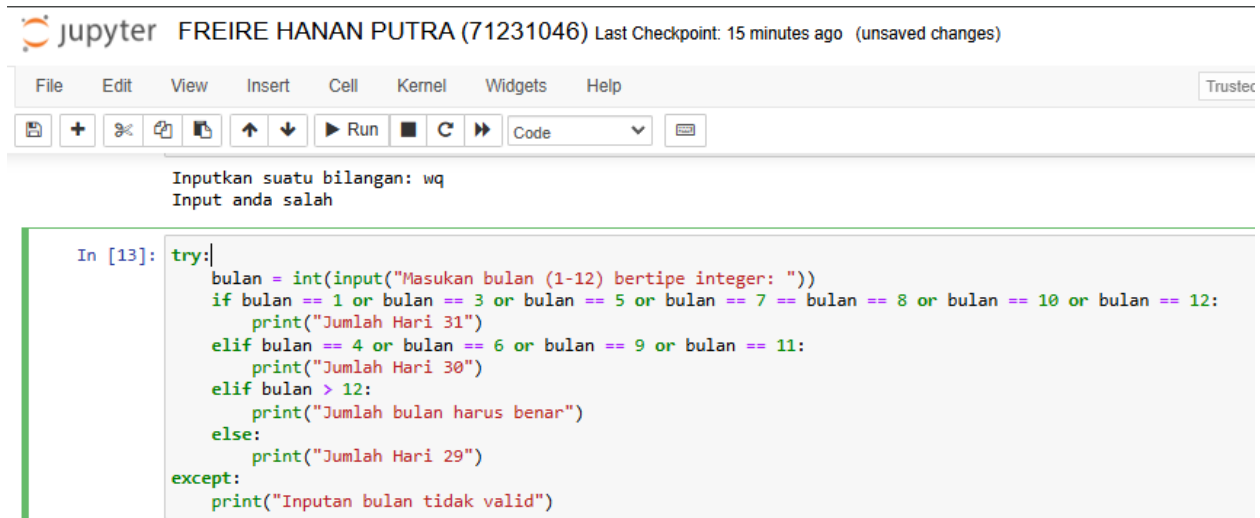
```
Inputkan suatu bilangan: 32
Positif
```

Penjelasan:

- Pertama saya membuat try untuk mengecek apakah tipe data yang diinputkan itu sama dengan tipe data yang digunakan pada program.
- Selanjutnya saya buat dengan metode ternary operator dengan cara mengecek apakah angka itu modulus 2 sama dengan 0 maka print positif jika salah maka print negative.
- Setelah itu pada bagian except jika yang diinputkan user bukan integer maka print input anda salah.

SOAL 3

Code:



The screenshot shows a Jupyter Notebook interface. At the top, the username 'FREIRE HANAN PUTRA (71231046)' and 'Last Checkpoint: 15 minutes ago (unsaved changes)' are visible. Below the menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) is a toolbar with icons for saving, adding cells, undo, redo, and running code. The code cell contains the following Python code:

```
In [13]: try:
        bulan = int(input("Masukan bulan (1-12) bertipe integer: "))
        if bulan == 1 or bulan == 3 or bulan == 5 or bulan == 7 == bulan == 8 or bulan == 10 or bulan == 12:
            print("Jumlah Hari 31")
        elif bulan == 4 or bulan == 6 or bulan == 9 or bulan == 11:
            print("Jumlah Hari 30")
        elif bulan > 12:
            print("Jumlah bulan harus benar")
        else:
            print("Jumlah Hari 29")
    except:
        print("Inputan bulan tidak valid")
```

The output of the code cell shows the following text:

```
Inputkan suatu bilangan: wq
Input anda salah
```

Penerapan:

```
Masukan bulan (1-12) bertipe integer: 12
Jumlah Hari 31
```

```
Masukan bulan (1-12) bertipe integer: tiga
Inputan bulan tidak valid
```

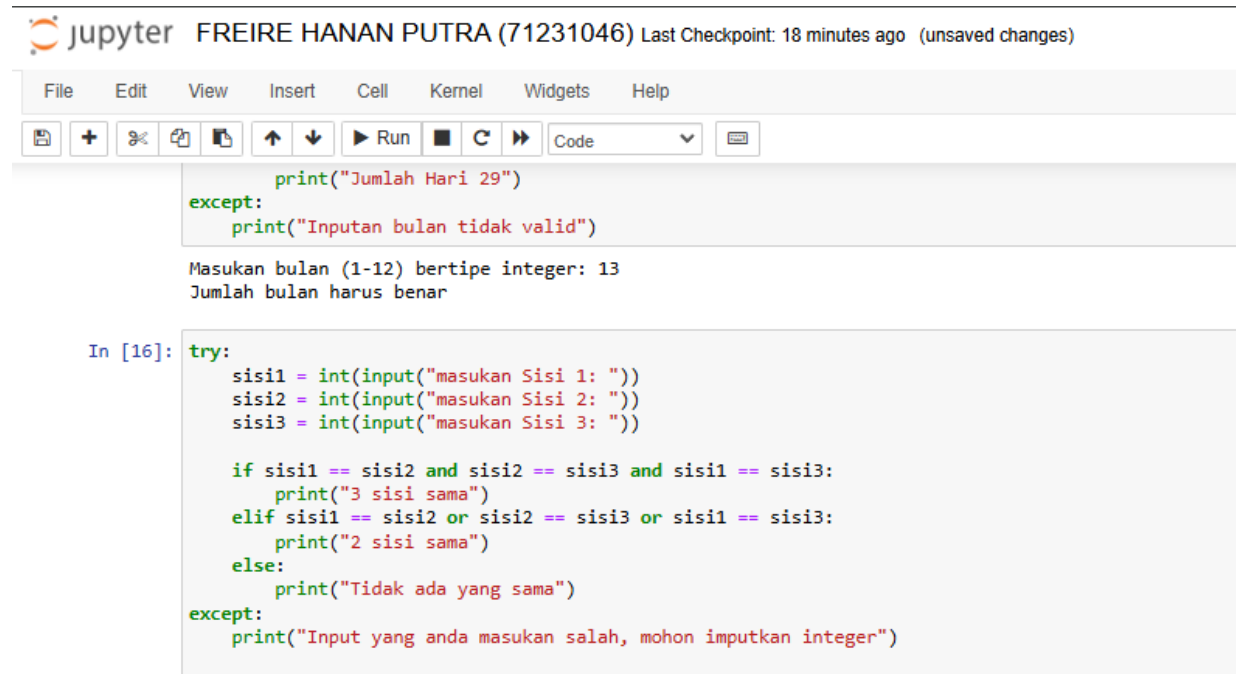
```
Masukan bulan (1-12) bertipe integer: 13
Jumlah bulan harus benar
```

Penjelasan:

- Pertama saya buat fitur try untuk mengecek apakah tipe data yang diinputkan user sama dengan tipe data program agar dapat tetap berjalan.
- Setelah itu saya buat kondisi untuk mengecek apakah pada bulan yang diinputkan user adalah 1 atau 3 atau 4 atau 7 atau 8. Jika salah satu true maka print jumlah hari adalah 31
- Jika tidak ada yang memenuhi maka lanjut mengecek apakah yang diinputkan user adalah 4 atau 6 atau 9 atau 11. Jika salah satunya true maka print 30.
- Setelah itu saya juga membuat jika bulan lebih dari 12 maka print jumlah bulan harus benar
- Jika semua kondisi tidak terpenuhi maka jumlah hari yang di tuju adalah bulan 2 maka print jumlah hari 29

SOAL 4

Code:



The screenshot shows a Jupyter Notebook interface. At the top, the username 'FREIRE HANAN PUTRA (71231046)' and 'Last Checkpoint: 18 minutes ago (unsaved changes)' are visible. The interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for saving, adding cells, and running code. The code cell contains a snippet of Python code with an exception handler. The output of the code is displayed below the cell, showing a prompt for a month and the resulting number of days.

```
print("Jumlah Hari 29")
except:
    print("Inputan bulan tidak valid")

Masukan bulan (1-12) bertipe integer: 13
Jumlah bulan harus benar
```

```
In [16]: try:
        sisi1 = int(input("masukan Sisi 1: "))
        sisi2 = int(input("masukan Sisi 2: "))
        sisi3 = int(input("masukan Sisi 3: "))

        if sisi1 == sisi2 and sisi2 == sisi3 and sisi1 == sisi3:
            print("3 sisi sama")
        elif sisi1 == sisi2 or sisi2 == sisi3 or sisi1 == sisi3:
            print("2 sisi sama")
        else:
            print("Tidak ada yang sama")
    except:
        print("Input yang anda masukan salah, mohon imputkan integer")
```

Penerapan:

```
masukan Sisi 1: 12
masukan Sisi 2: 43
masukan Sisi 3: 43
2 sisi sama
```

```
masukan Sisi 1: 423
masukan Sisi 2: tiga
Input yang anda masukan salah, mohon imputkan integer
```

Penjelasan:

- Pertama saya membuat inputkan untuk sisi 1, 2 dan 3 dengan tipe data integer.
- Setelah itu saya membuat pemisalan jika sisi 1 = dengan sisi 2 dan sisi 2 = sisi 3 dan sisi 3 = sisi 1. Jika semua kondisi tersebut terpenuhi maka print 3 sisi sama.
- Selanjutnya saya juga buat pemisalan jika sisi 1 = sisi 2 atau sisi 2 = sisi 3 atau sisi 3 = sisi 1. Artinya jika salah satu kondisi terpenuhi maka ada 2 sisi yang sama.
- Jika semua kondisi tidak terpenuhi maka tidak ada yang sama.