



Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

NIM	71230977
Nama Lengkap	MICHAEL HOSEA
Minggu ke / Materi	01 / Pengantar Python

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.

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
YOGYAKARTA
2024

BAGIAN 1: MATERI MINGGU INI (40%)

Pada Minggu kali ini (7 Februari 2024), Materinya berorientasi dengan pengenalan ulang Bahasa Python yang sudah pernah dipelajari pada semester 1.

Kenapa Python?

Python merupakan bahasa pemrograman level tinggi yang interpreted, mendukung Object Oriented Programming (OOP) yang dimana merupakan metode pemrograman yang fokus atau berorientasi pada objek. Selain itu memiliki sifat dynamic semantics. Python juga merupakan salah satu bahasa pemrograman yang paling banyak dipakai setelah Javascript. Secara umum Python merupakan salah satu bahasa pemrograman yang paling populer di dunia karena aturan dan sintaksnya yang sederhana untuk dipelajari bagi pemula.



Gambar 1 : Logo Python

Bukti paling mudah bahwa bahasa pemrograman Python merupakan bahasa yang ramah pemula dapat dilihat dari perbandingan source code program Hello World jika dibandingkan dengan Java dan C.

Pada bahasa pemrograman Java:

```
1 public class Main {
2     public static void main(String[] args){
3         System.out.println("Hello World!");
4     }
5 }
```

Gambar 2 : Print "Hello World" Menggunakan Java

Pada bahasa pemrograman C:

```
1 #include <stdio.h>
2
3 int main(void){
4     printf("Hello world");
5 }
```

Gambar 3 : Print "Hello World" Menggunakan C

Pada bahasa pemrograman Python:

```
1 print("Hello World!");
```

Gambar 4 : Print "Hello World" Menggunakan Python

Selain pemrograman yang ramah bagi pemula, Python memiliki banyak kelebihannya antara lain:

1. Pustaka bawaan dari Python sudah bervariasi dan mencakup banyak aspek dasar yang umumnya dibutuhkan oleh seorang programmer.
2. Open Source sehingga dapat digunakan bebas tanpa perlu membayar lisensi.
3. Aktif dikembangkan sehingga fitur dan kemampuannya selalu meningkat/menambah.
4. Sintaksnya yang sangat sederhana dan sangat menyerupai Bahasa Inggris.

Walaupun memiliki banyak kelebihan, ternyata Python juga memiliki berbagai kekurangan antara lain:

1. Belum dapat mendukung pembuatan aplikasi pada platform Android dan iOS.
2. Konsumsi memory yang besar, sehingga tidak cocok digunakan untuk kasus yang besar.
3. Kecepatan proses Python lebih lambat jika dibandingkan bahasa pemrograman seperti C.

Editor untuk Python serta Penginstalan

Untuk menulis suatu program, kita memerlukan editor untuk menulis source code. Semakin canggih suatu editor, biasanya akan sangat mempermudah dalam pembuatan program. Berikut ini merupakan beberapa contoh Editor yang biasanya digunakan untuk membuat program dalam bahasa pemrograman Python, antara lain:

1. Visual Studio Code + Python Extension (<https://code.visualstudio.com>)
2. PyCharm (<https://www.jetbrains.com/pycharm/>)
3. Spyder (<https://www.spyder-ide.org/>)
4. ActivePython (<https://www.activestate.com/products/activepython/>)
5. IDLE (<https://docs.python.org/3/library/idle.html>)

Anaconda

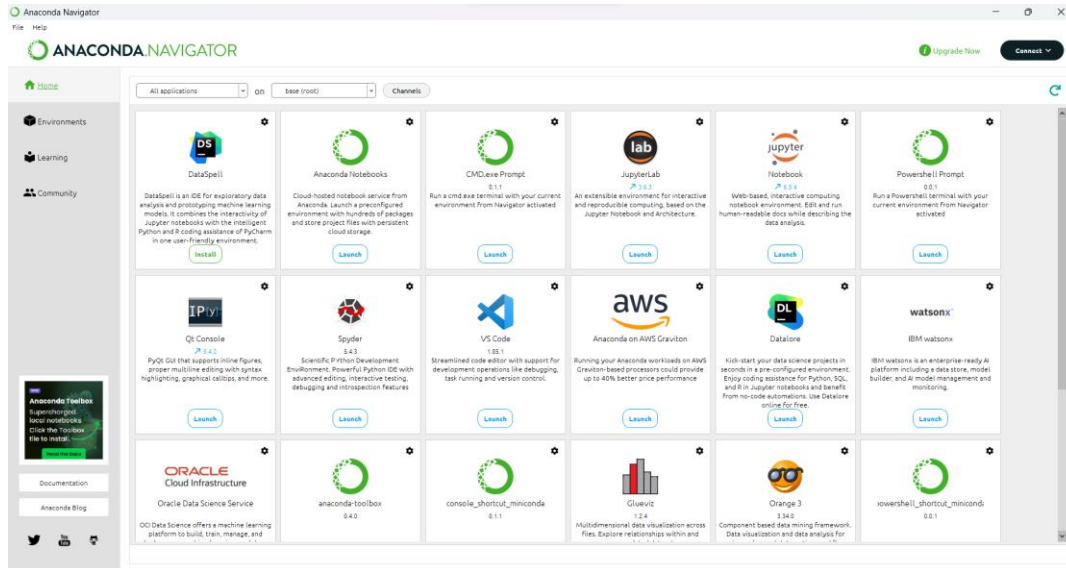
- Penginstalan Anaconda

Instalasi Python cukup mudah, tetapi ada beberapa hal yang perlu diperhatikan. Sebelum memasang Python, pastikan Anda menentukan versi yang akan diinstal dan pilih jenis yang sesuai dengan kebutuhan Anda. Ada banyak versi Python yang tersedia di internet, termasuk Anaconda. Anaconda adalah distribusi Python yang fokus pada komputasi ilmiah dan dirancang untuk membuatnya lebih mudah. Anda dapat mengunduh Anaconda secara gratis dari anaconda.com/products/distribution dan menginstalnya dengan mudah di komputer Anda. Untuk memasang Anaconda, langkah pertama adalah mengunduh installer dari situs web Anaconda, lalu ikuti petunjuk yang disediakan dalam installer tersebut.



Gambar 5: Logo Anaconda

- **Pengoperasian Anaconda**

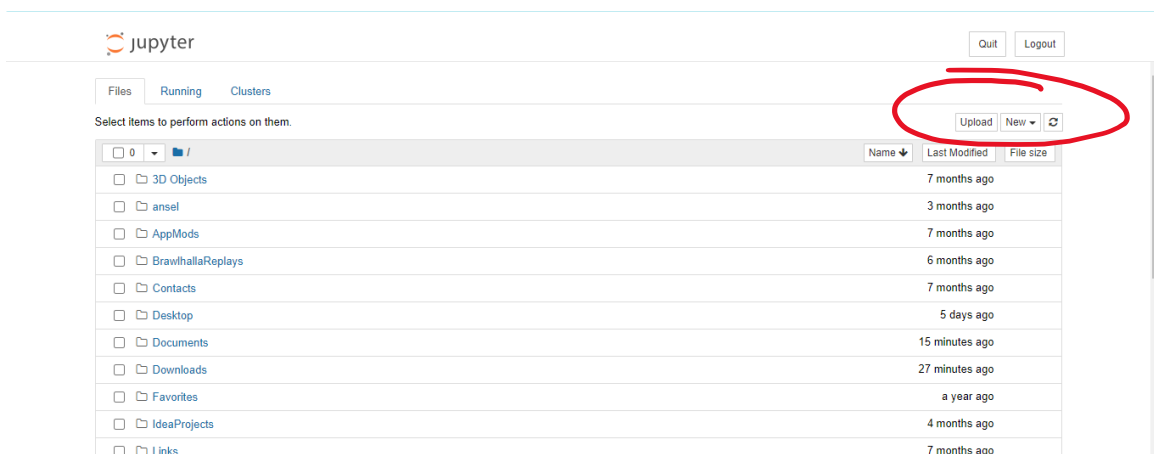


Gambar 6: Tampilan Antarmuka Anaconda Navigator

Gambaran awal saat Anda membuka Anaconda terlihat pada gambar di atas. Selain itu, terdapat berbagai aplikasi lain seperti Jupyter Lab, Jupyter Notebook, dan sejumlah aplikasi lainnya yang dapat digunakan. Anaconda terkenal karena kemudahannya; hanya dengan beberapa klik, Anda dapat menggunakan aplikasi yang diinginkan.

Jupyter Notebook

Saat pertama kali membuka Jupyter Notebook, langkah awal yang perlu dilakukan adalah membuat atau mengunggah notebook baru. Ini dapat dilakukan dengan mengklik menu yang terletak di bagian kanan atas.



Gambar 7: Tampilan Antarmuka Jupyter Notebook

Latihan Mandiri

Link Repository Latihan Mandiri : <https://github.com/Mhoseaaa/alpro-week1>

Link pendukung untuk Latihan :

- <https://revou.co/panduan-teknis/matplotlib-python>
- <https://matplotlib.org/stable/tutorials>
- <https://numpy.org/doc/stable/>

Latihan 1.1

A. Source Code

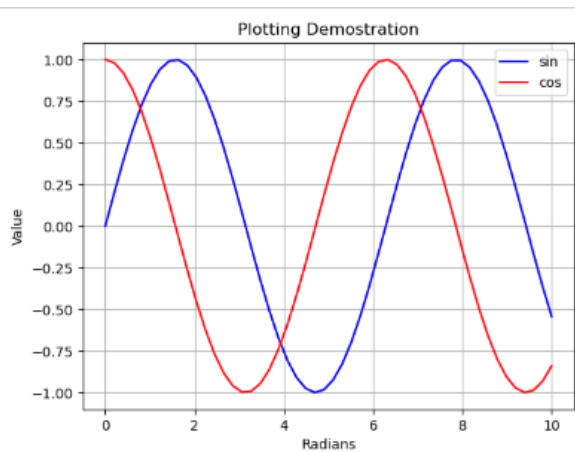
```
%matplotlib inline

import matplotlib.pyplot as plt
import numpy as np

x = np.linspace(0,10)
y = np.sin(x)
z = np.cos(x)

plt.plot(x,y,'b',x,z,'r')
plt.xlabel('Radians');
plt.ylabel('Value');
plt.title('Plotting Demostration')
plt.legend(['sin','cos'])
plt.grid()
```

B. Output



C. Penjelasan

1. `%matplotlib inline`

Baris ini adalah perintah khusus Jupyter Notebook yang menyematkan grafik langsung di dalam notebook.

2. `import matplotlib.pyplot as plt`

Baris ini mengimpor modul `matplotlib.pyplot`, yang menyediakan fungsi untuk membuat dan menyesuaikan plot di Python. Biasanya dialias sebagai `plt` untuk singkatnya.

3. `import numpy as np`

Baris ini mengimpor pustaka `numpy`, yang menawarkan alat yang canggih untuk perhitungan numerik dan manipulasi array. Seringkali dialias sebagai `np`.

4. `x = np.linspace(0, 10)`

Baris ini membuat array nilai yang berjarak sama antara 0 dan 10 (inklusif) menggunakan `numpy.linspace`. Ini menyimpan nilai-nilai ini dalam variabel `x`, yang akan berfungsi sebagai data sumbu x untuk plot.

5. `y = np.sin(x)`

Baris ini menghitung sinus dari setiap nilai dalam array `x` menggunakan `numpy.sin` dan menyimpan hasilnya dalam variabel `y`. Nilai-nilai ini akan mewakili data sumbu y plot pertama.

6. `z = np.cos(x)`

Baris ini secara serupa menghitung kosinus dari setiap nilai dalam array `x` menggunakan `numpy.cos` dan menyimpan hasilnya dalam variabel `z`. Nilai-nilai ini akan mewakili data sumbu y plot kedua.

7. `plt.plot(x, y, 'b', x, z, 'r')`

Baris ini membuat plot yang sebenarnya:

Ini memanggil fungsi `plt.plot` untuk menghasilkan dua garis pada plot yang sama.

Baris pertama diplot menggunakan nilai dalam `x` untuk sumbu x dan nilai dalam `y` untuk sumbu y, dengan warna biru ('b').

Baris kedua diplot menggunakan nilai `x` yang sama tetapi nilai dalam `z` untuk sumbu y, dengan warna merah ('r').

8. `plt.xlabel('Radians')`

Baris ini menetapkan label untuk sumbu x menjadi "Radians" menggunakan `plt.xlabel`.

9. `plt.ylabel('Value')`

Baris ini menetapkan label untuk sumbu y menjadi "Value" menggunakan `plt.ylabel`.

10. `plt.title('Plotting Demonstration')`

Baris ini menetapkan judul plot menjadi "Plotting Demonstration" menggunakan `plt.title`.

11. plt.legend(['sin', 'cos'])

Baris ini menambahkan legenda ke plot menggunakan plt.legend, menunjukkan garis mana yang sesuai dengan fungsi sinus dan cosinus.

12. plt.grid()

Baris ini menambahkan kisi ke plot menggunakan plt.grid, memberikan referensi visual untuk nilai-nilai pada kedua sumbu.

SOAL 2

A. Source Code

```
# Gerard membeli emas 25 gram dengan harga Rp. 650.000/gram. Jika sekarang
# harga emas menjadi Rp. 685.000/gram,
# • Berapa keuntungan yang didapat oleh Gerard (dalam Rp dan dalam %)?
# • Jika Gerard kemudian membeli lagi 15 gram emas dengan harga Rp. 685.000, maka
# Gerard sekarang memiliki total 40 gram emas. Jika kemudian harga emas naik lagi
# menjadi Rp. 715.000, berapa keuntungan yang didapat oleh Gerard (dalam Rp dan dalam %)?
|
# Emas dan Harga
# E untuk Emas, HEL untuk Harga Emas Lama, dan HEB untuk Harga Emas Baru
E = 25
HEL = 650000
HEB = 685000

# Keuntungan
# PRP untuk Profit secara RP/Rupiah dan PP untuk Profit secara persen
PRP = (HEB * E) - (HEL * E)
PP = float((PRP * 100) / (HEL * E))

print("Emas Gerard      : ", E, "gram")
print("Profit (Nominal)  : ", "Rp", PRP)
print("Profit (Persen)   : ", PP, "%")

# Penambahan Emas dan Kenaikan Harga Emas
# TE untuk Tambahan Emas, TET untuk Total Emas Terkumpul, dan KHE untuk Kenaikan Harga Emas
TE = 15
TET = TE + E
KHE = 715000

# TPN untuk Total Profit Nominal, TPP untuk Total Profit Persen
TPN = (TET * KHE) - (HEL * E) - (TE * HEB)
TPP = float((TPN * 100) / ((E * HEL) + (TE * HEB)))

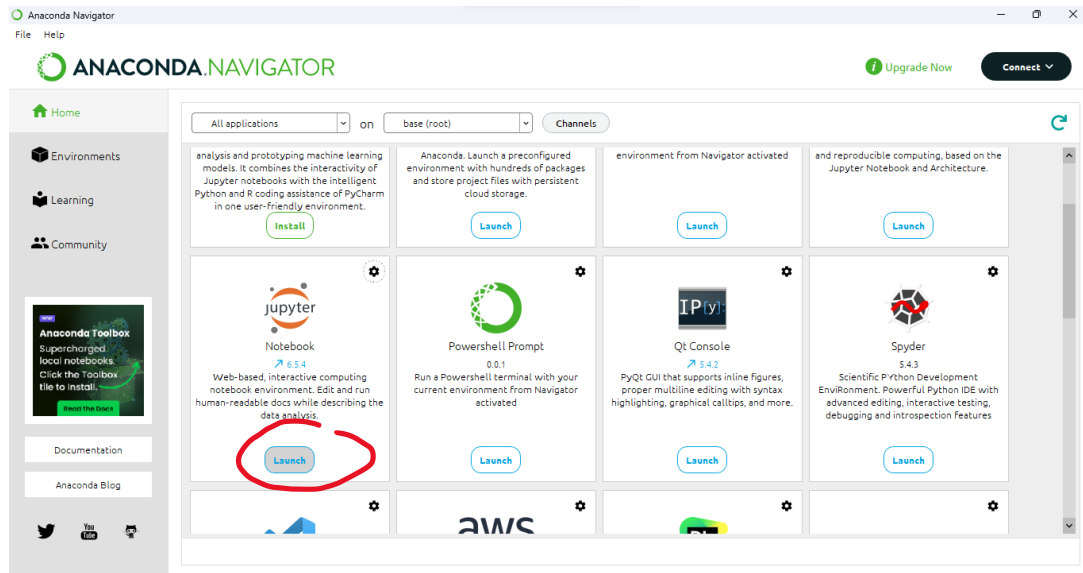
print("Total Emas Gerard : ", TET, "gram")
print("Profit (Nominal)   : ", "Rp", TPN)
print("Profit (Persen)    : ", TPP, "%")
```

B. Output

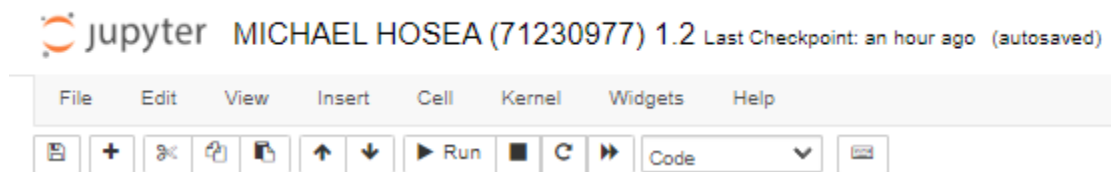
```
Emas Gerard      : 25 gram
Profit (Nominal)  : Rp 875000
Profit (Persen)   : 5.384615384615385 %
Total Emas Gerard : 40 gram
Profit (Nominal)  : Rp 2075000
Profit (Persen)   : 7.822808671065033 %
```

C. Result

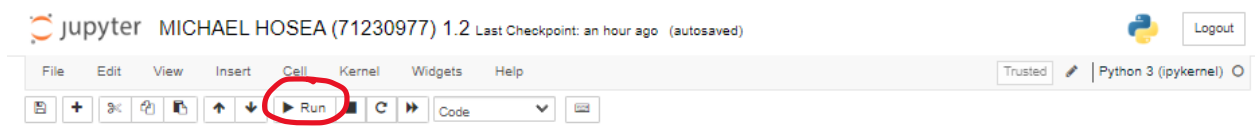
1. Membuka Anacoda Navigator, lalu memilih pilihan Jupyter Notebook.



2. Membuat file baru, mengganti judul menjadi nama beserta NIM.



3. Mulai membuat code dengan mendeklarasikan Emas dan Harganya
4. $PRP = (HEB * E) - (HEL * E)$ digunakan untuk mencari profit harga dengan nominal (berdasarkan Rupiah)
5. $PP = \text{float}((PRP * 100) / (HEL * E))$ digunakan untuk mencari profit harga dengan bentuk persen.
6. Pada Result nomor 4 dan 5, code tersebut merupakan rumus untuk mencari bunga Dimana pada nomor 4 harus mencari profitnya secara rupiah dengan mengalikan (Harga Emas Baru x Jumlah Emas) – (Harga Emas Lama x Jumlah Emas)
7. Sedangkan pada Result nomor 5, menjelaskan bagaimana cara mencari profit secara persen dengan mengalikan hasil PRP dengan 100% lalu dibagi oleh Total Harga Lama.
8. $TPN = (TET * KHE) - (HEL * E) - (TE * HEB)$
9. $TPP = \text{float}((TPN * 100) / ((E * HEL) + (TE * HEB)))$
10. Result nomor 9-10 juga sama saja menjelaskan apa yang dijelaskan pada Result nomor 4 dan 5, mencari total profit secara nominal (menggunakan rupiah) dan mencari total profit dengan menggunakan persen.
11. Setelah selesai memasukkan code, menjalankan program dengan mengeklik button Run.



12. Lalu melakukan save dengan menekan tombol CTRL + S secara bersamaan.

SOAL 3

A. Source Code

```
In [16]: # Erika memiliki uang 200 juta rupiah dan ingin disimpan di deposito Pasti Cuan sampai uangnya menjadi
# minimal 400 juta, Berapa Lama waktu yang dibutuhkan?
# Catatan: bunga 10% per-tahun.

# variabel
jumlah_uang_awal = 200_000_000 # dalam rupiah
target_uang = 400_000_000 # dalam rupiah
bunga_tahunan = 0.10 # 10%

# Inisialisasi variabel tambahan
total_uang = jumlah_uang_awal
tahun = 0

# Loop sampai total uang mencapai target
while total_uang < target_uang:
    bunga = total_uang * bunga_tahunan
    total_uang += bunga
    tahun += 1

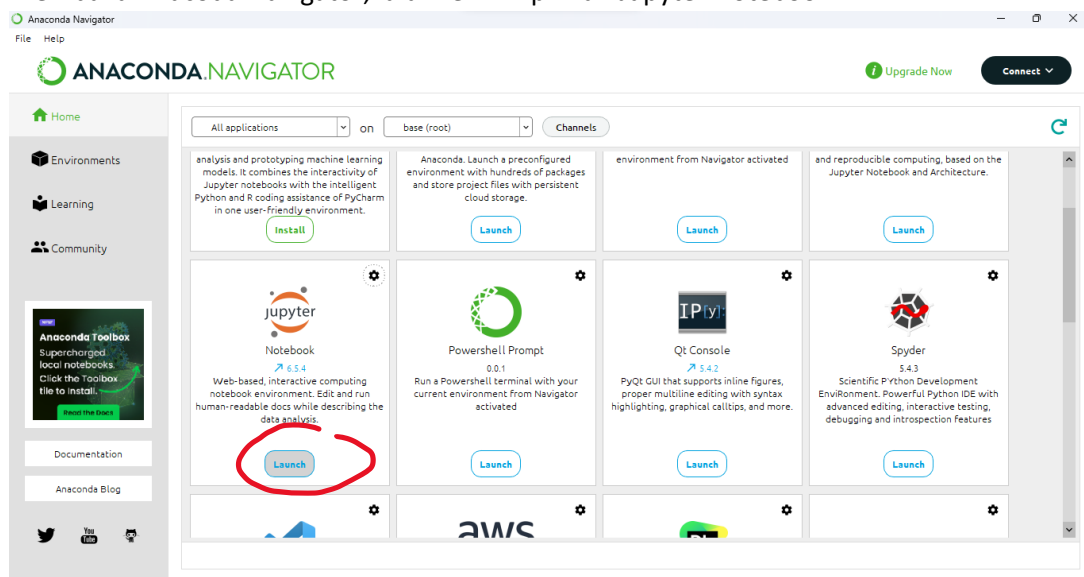
# Output
print("Jumlah tahun yang dibutuhkan : {}".format(tahun), "tahun")
print("Jumlah uang yang terkumpul : Rp{:, .2f}".format(total_uang))
```

B. Output

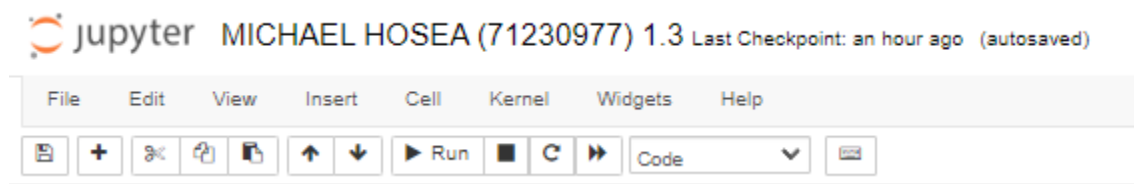
```
Jumlah tahun yang dibutuhkan : 8 tahun
Jumlah uang yang terkumpul : Rp428,717,762.00
```

C. Penjelasan

1. Membuka Anaconda Navigator, lalu memilih pilihan Jupyter Notebook.

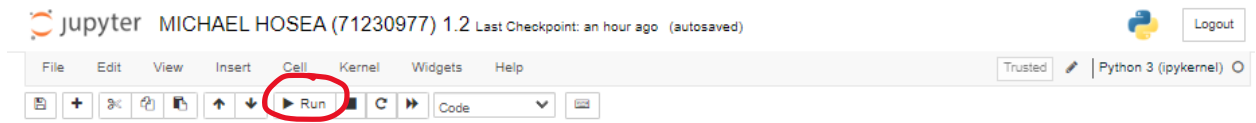


2. Membuat file baru, mengganti judul menjadi nama beserta NIM.



3. Membuat variable seperti jumlah uang awal, target, serta bunga tahunan.

4. Mulai melakukan pengodingan yang akan dipecah line per line dibawah ini.
5. `while total_uang < target_uang`
Ini adalah awal dari loop while. Loop ini akan terus berjalan selama `total_uang` masih kurang dari `target_uang`.
6. `bunga = total_uang * bunga_tahunan`
Baris ini digunakan untuk menghitung bunga yang diperoleh dari jumlah uang saat ini (`total_uang`) dengan menggunakan persentase bunga tahunan (`bunga_tahunan`).
7. `total_uang += bunga`
Baris ini menambahkan jumlah bunga yang diperoleh (`bunga`) ke dalam `total_uang`, sehingga `total_uang` sekarang mencerminkan total uang setelah menerima bunga.
8. `tahun += 1`
Baris ini menambahkan 1 ke variabel `tahun`, yang digunakan untuk menghitung berapa lama waktu yang dibutuhkan untuk mencapai target jumlah uang.
9. Setelah selesai memasukan code, menjalankan program dengan mengeklik button Run.



10. Lalu melakukan save dengan menekan tombol CTRL + S secara bersamaan.