DIGITAL TALENT SCHOLARSHIP 2019

Big Data Analytics

Praktek Membuat Instance Jupyter Notebook dari AWS Cloud (Amazon SageMaker) & Pengenalan Pemrograman Python Ke-1 of 2

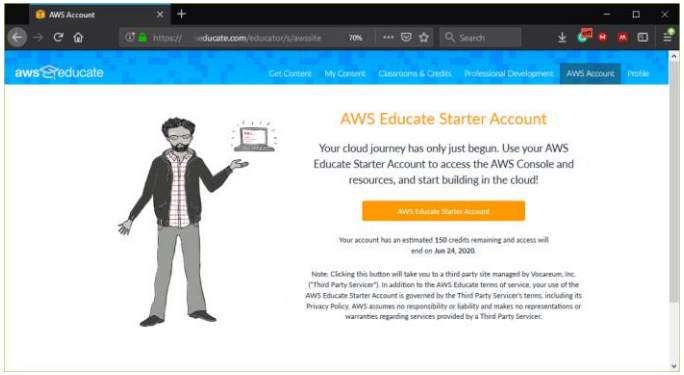
Oleh: Imam Cholissodin | imamcs@ub.ac.id, Putra Pandu Adikara, Sufia Adha Putri
Asisten: Guedho, Sukma, Anshori, Aang dan Gusti
Fakultas Ilmu Komputer (Filkom) Universitas Brawijaya (UB)

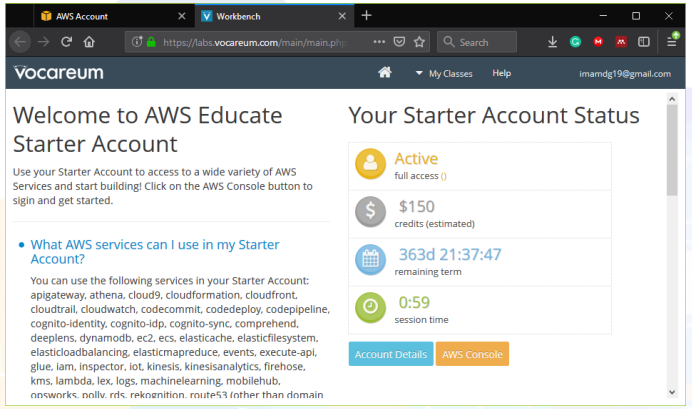
Pokok Bahasan

1. Buat Instance Jupyter Notebook dari AWS Cloud (Amazon SageMaker)
2. Pengenalan Pemrograman Python (run di SageMaker)
3. Tugas

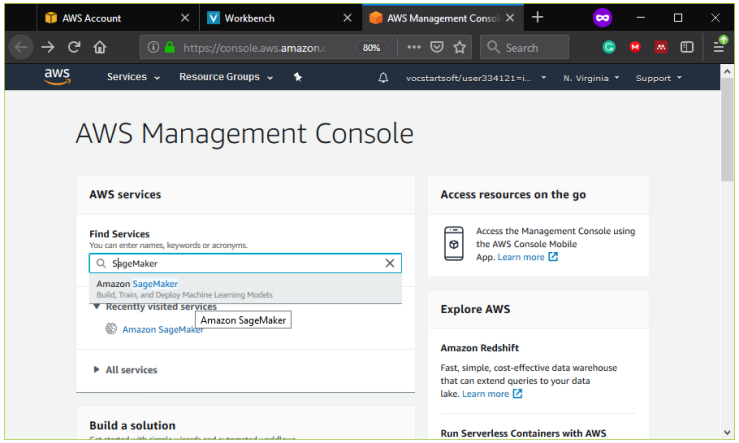
Buat SageMaker

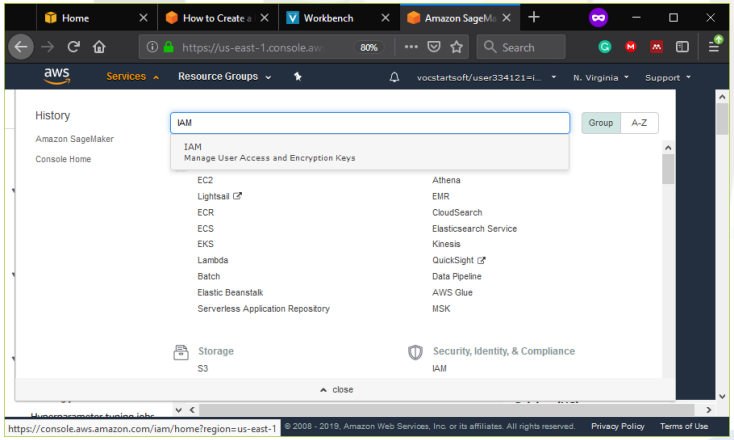
- Login Ke AWS Educate, Klik “AWS Educate Starter Account” > “AWS Console”





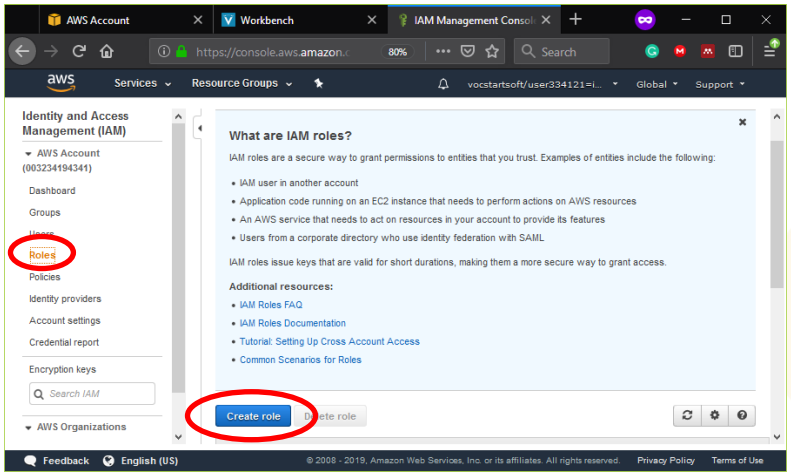
- Pada “WS Management Console” > ketik “SageMaker” > Klik “Amazon SageMaker” > Klik Services > ketik “IAM” > Klik “IAM Manage ..”

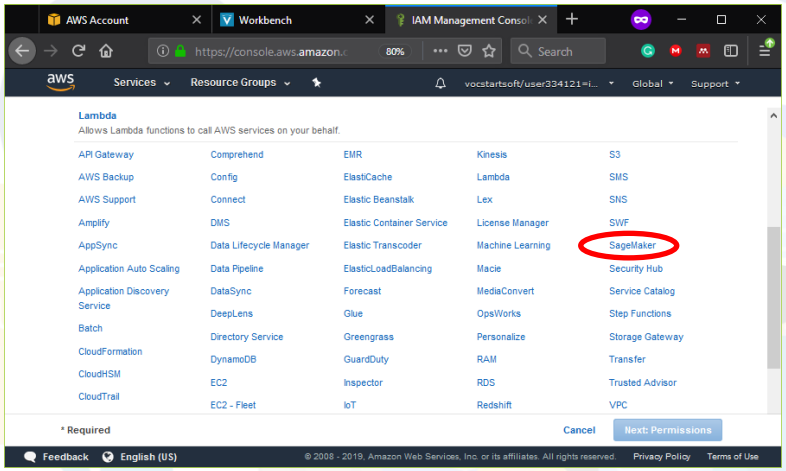




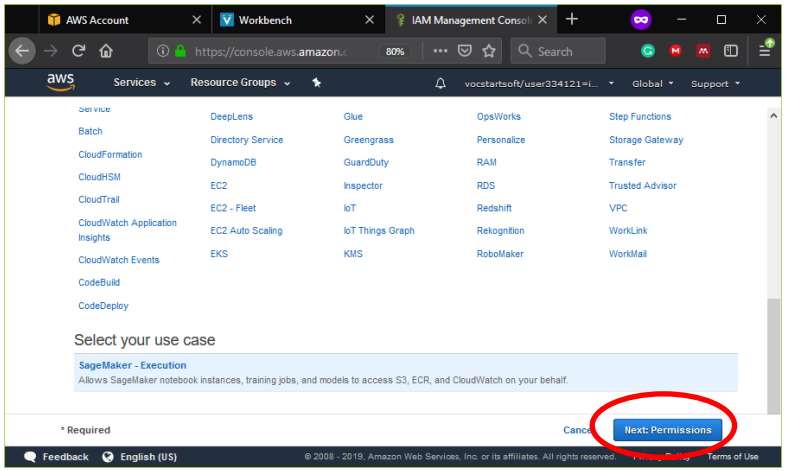
Buat SageMaker

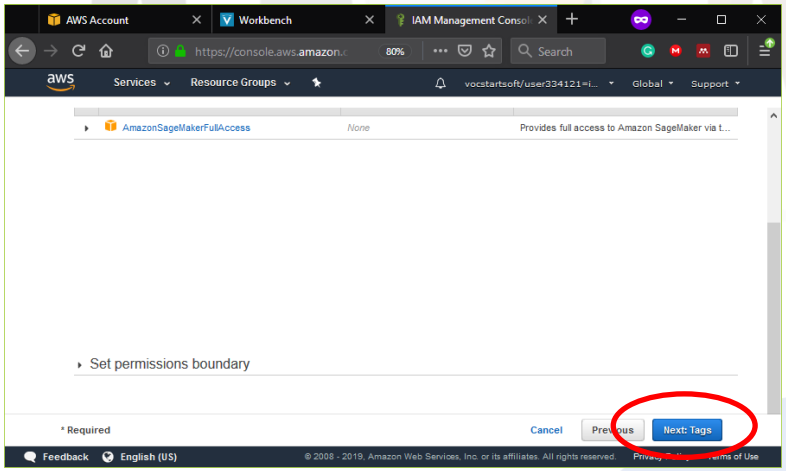
- Klik “Roles”, klik “Create role” > Klik “SageMaker”





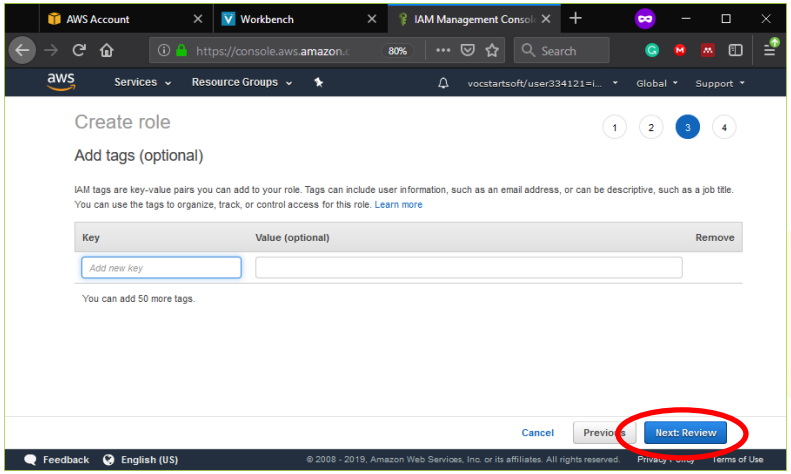
- Klik “Next: Permissions” > Klik “Next: Tags”

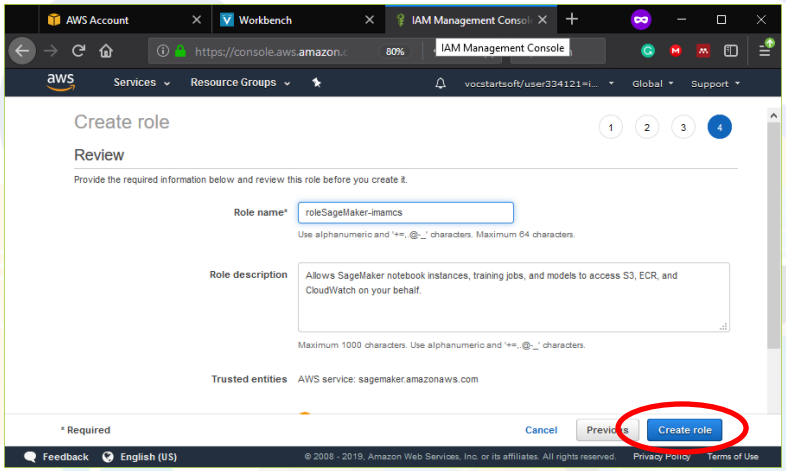




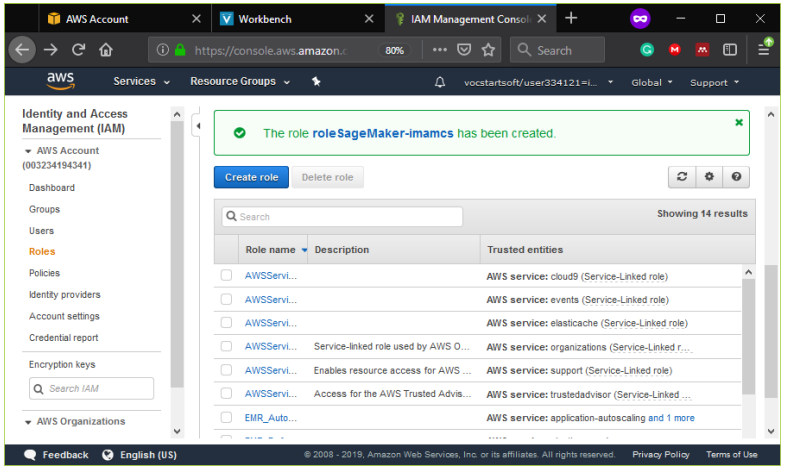
Buat SageMaker

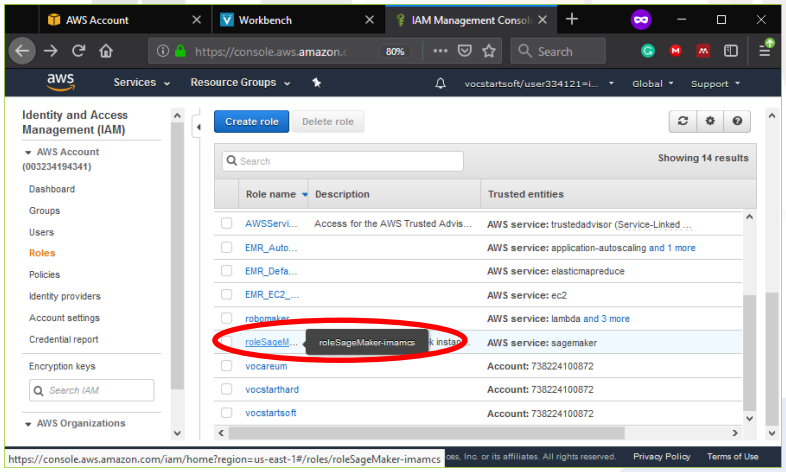
- Klik “Next: Review” > Masukkan Role name dll. > Klik “Create role”





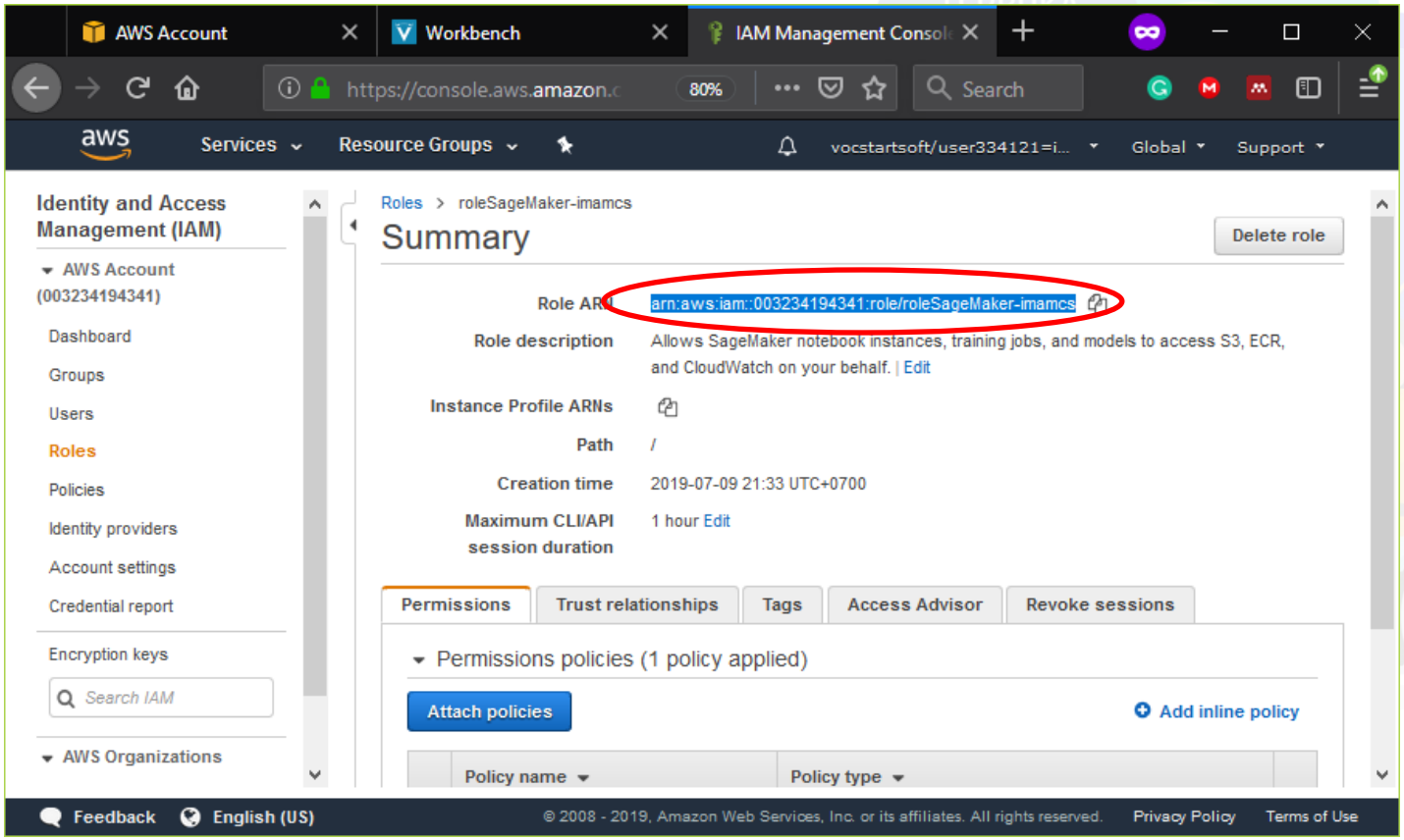
- The role “roleSageMaker-imamcs” has been created > Klik role yg telah dibuat





Buat SageMaker

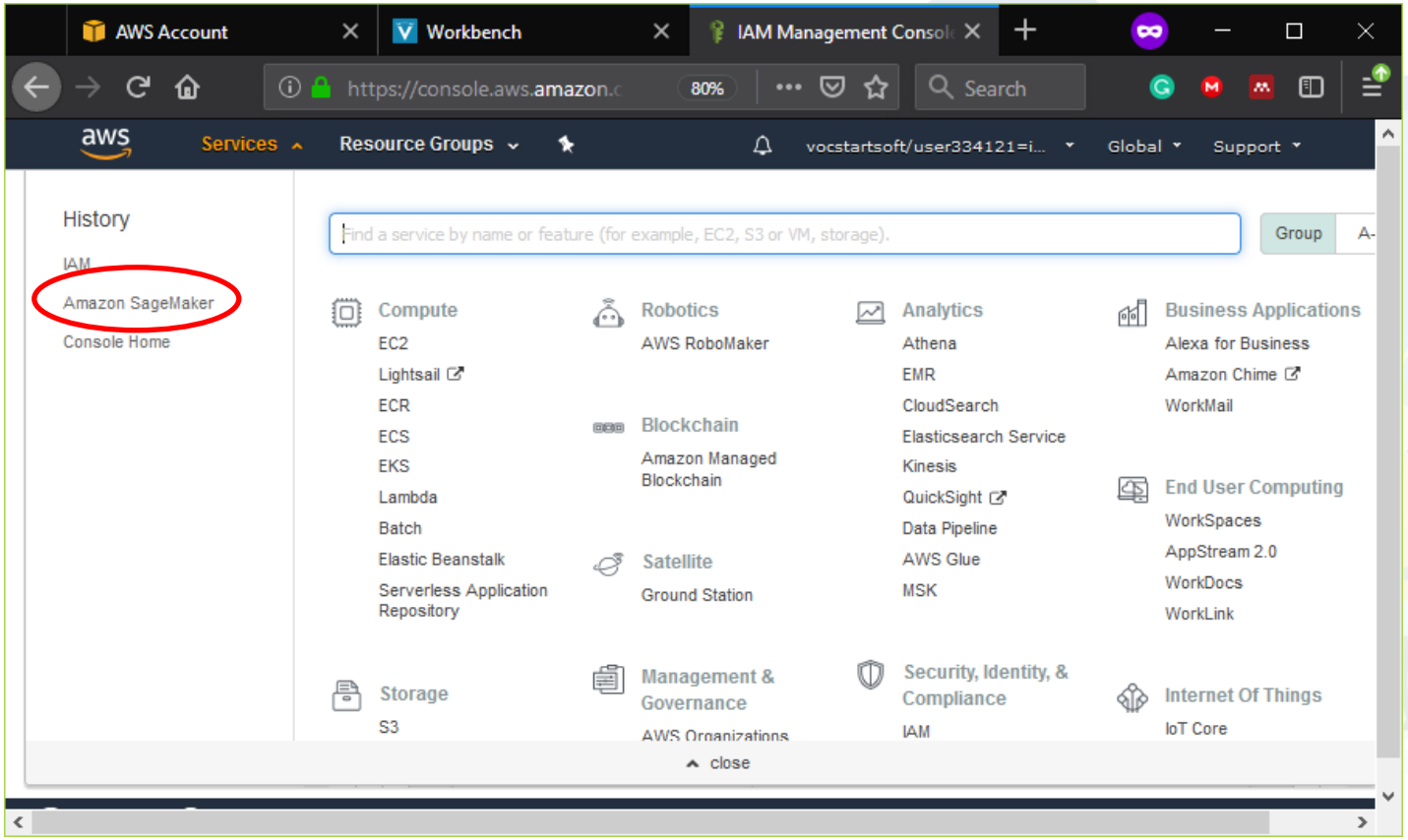
- Copy Role ARN > “arn:aws:iam::003234194341:role/roleSageMaker-imamcs” dengan klik icon copy



- Hasil copy tersebut akan dipaste pada saat membuat instance SageMaker

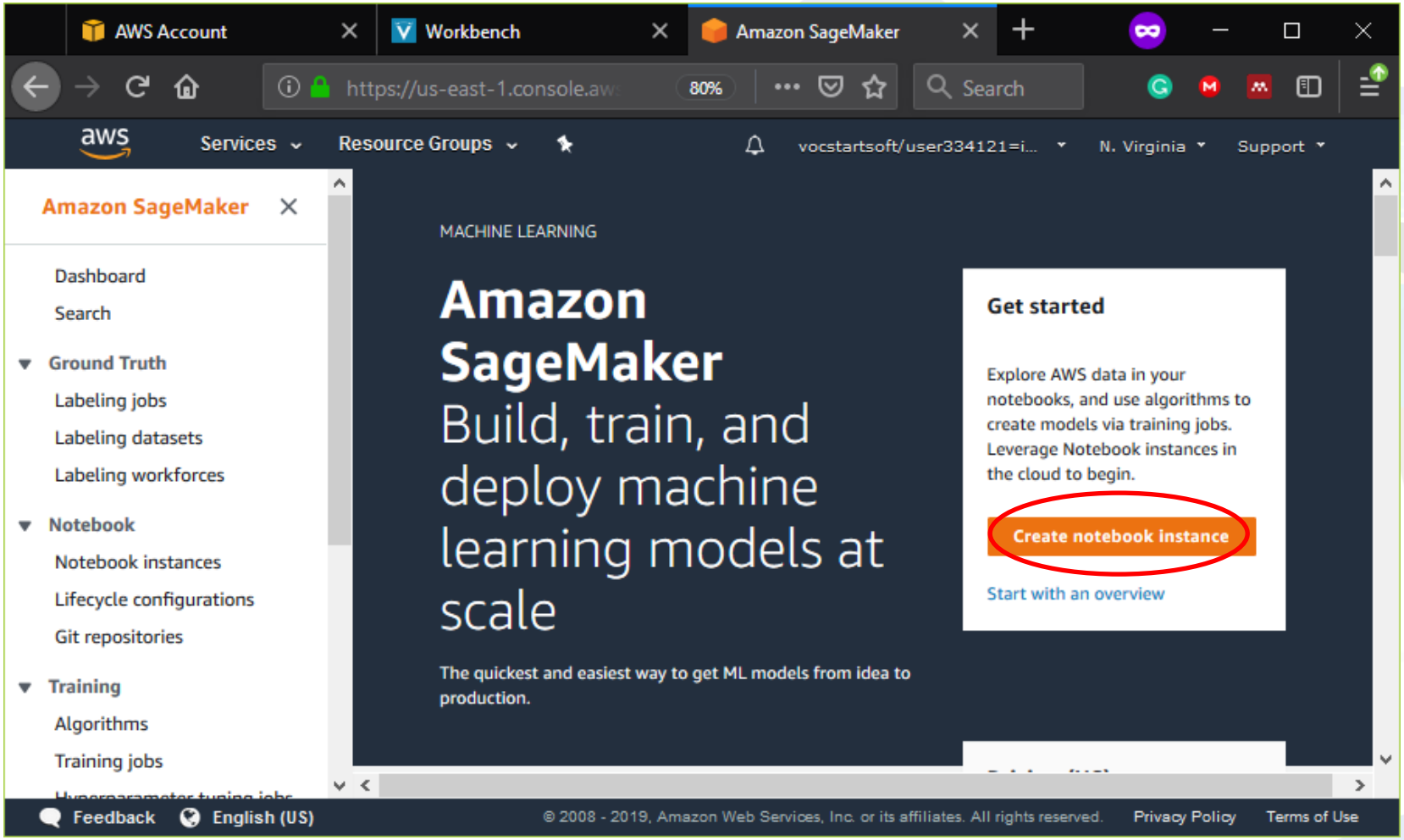
Buat SageMaker

- Klik “Services” > Klik “Amazon SageMaker”



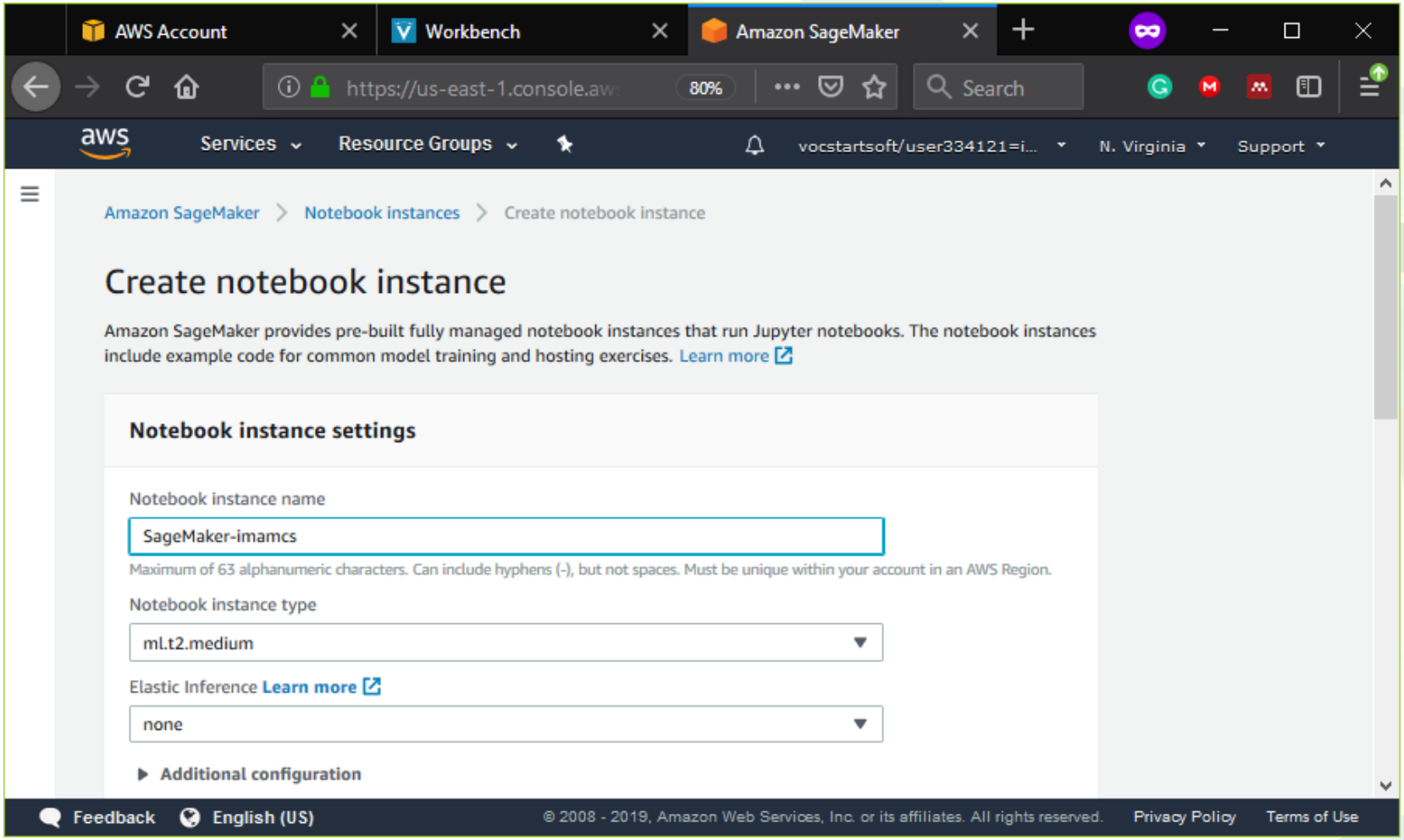
Buat SageMaker

- Klik “Create notebook instance”



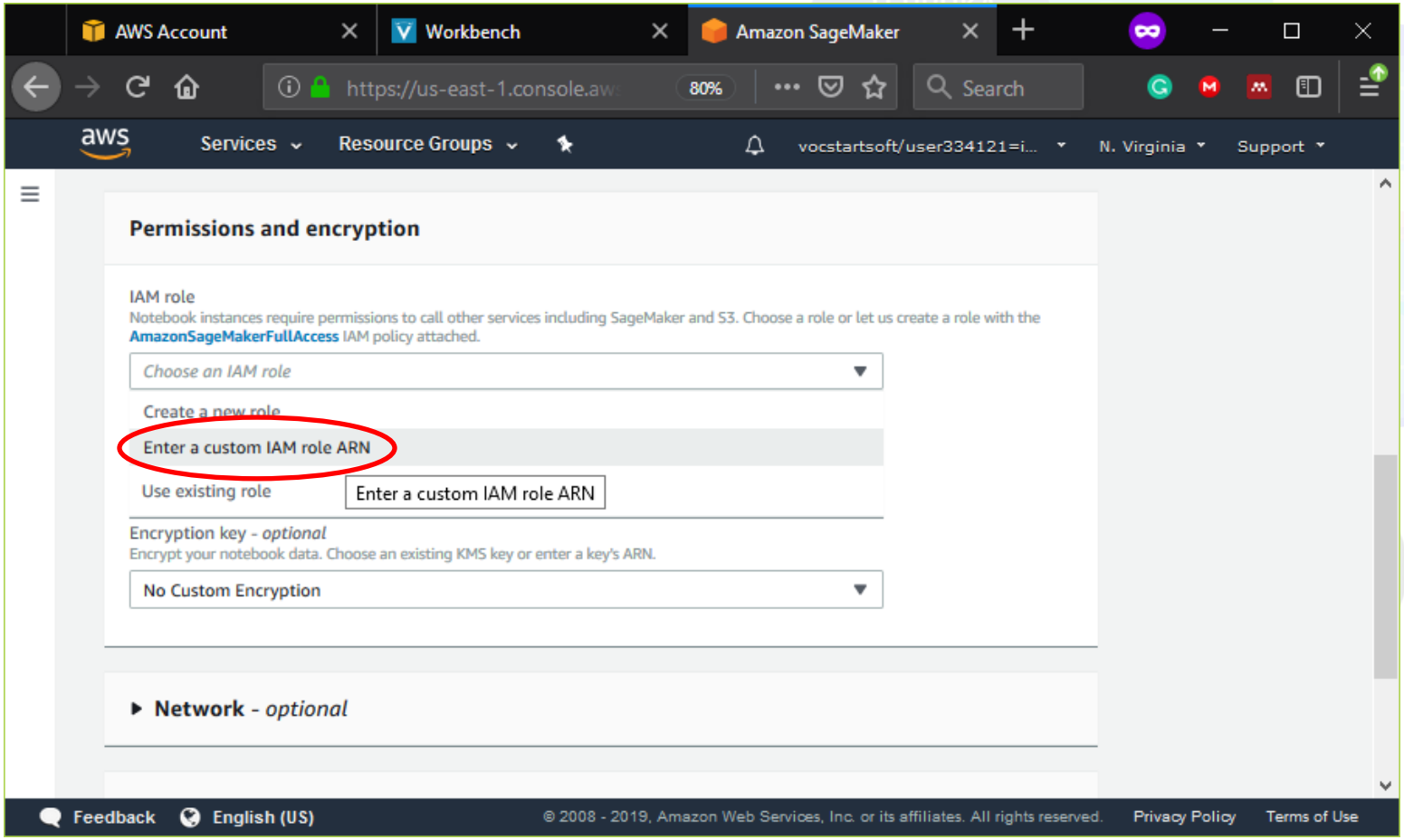
Buat SageMaker

- Masukkan Notebook instance nama, misal “SageMaker-imamcs”



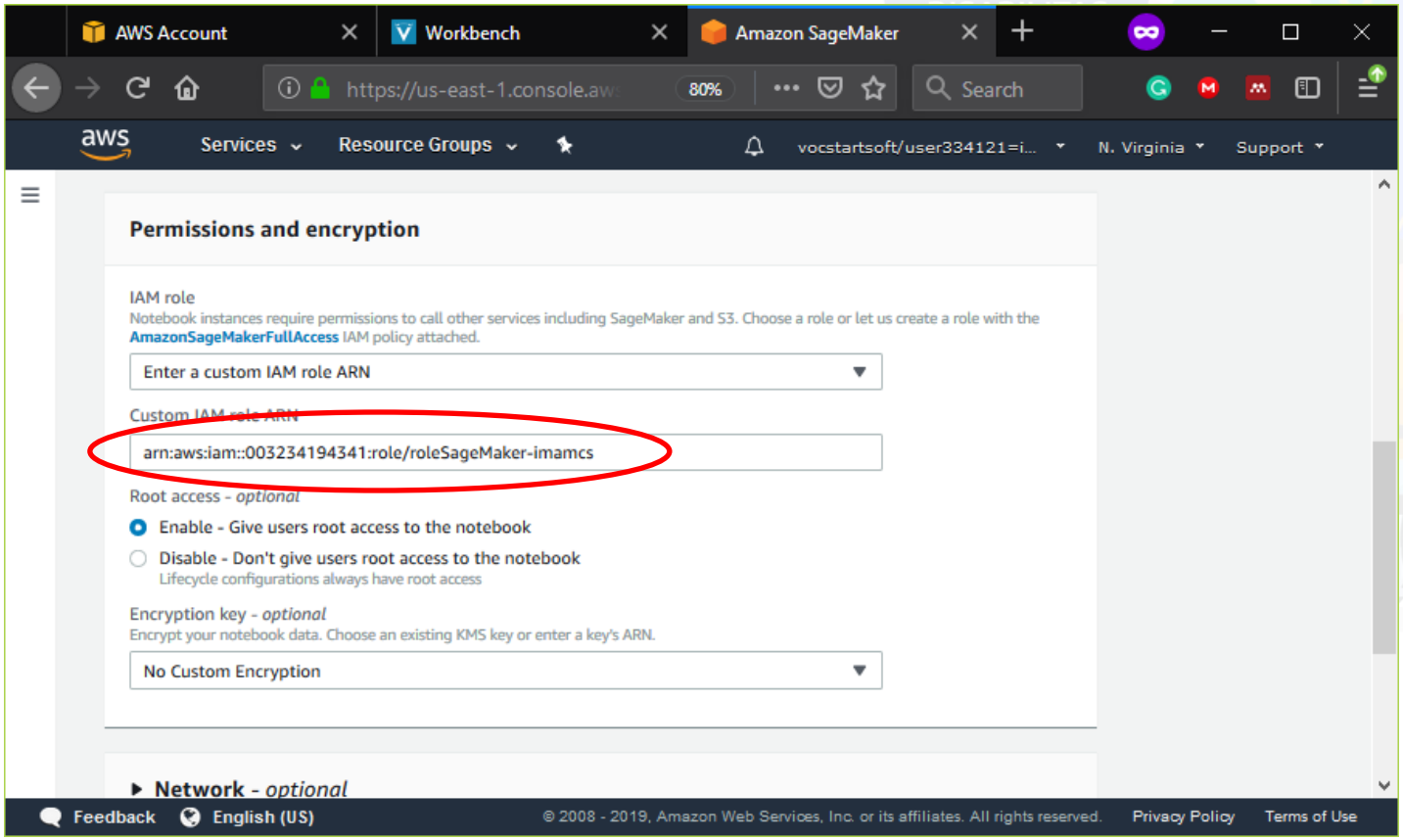
Buat SageMaker

- Scroll ke “Permissions and encryption” > klik “Choose IAM role” > klik “Enter a custom IAM role ARN”



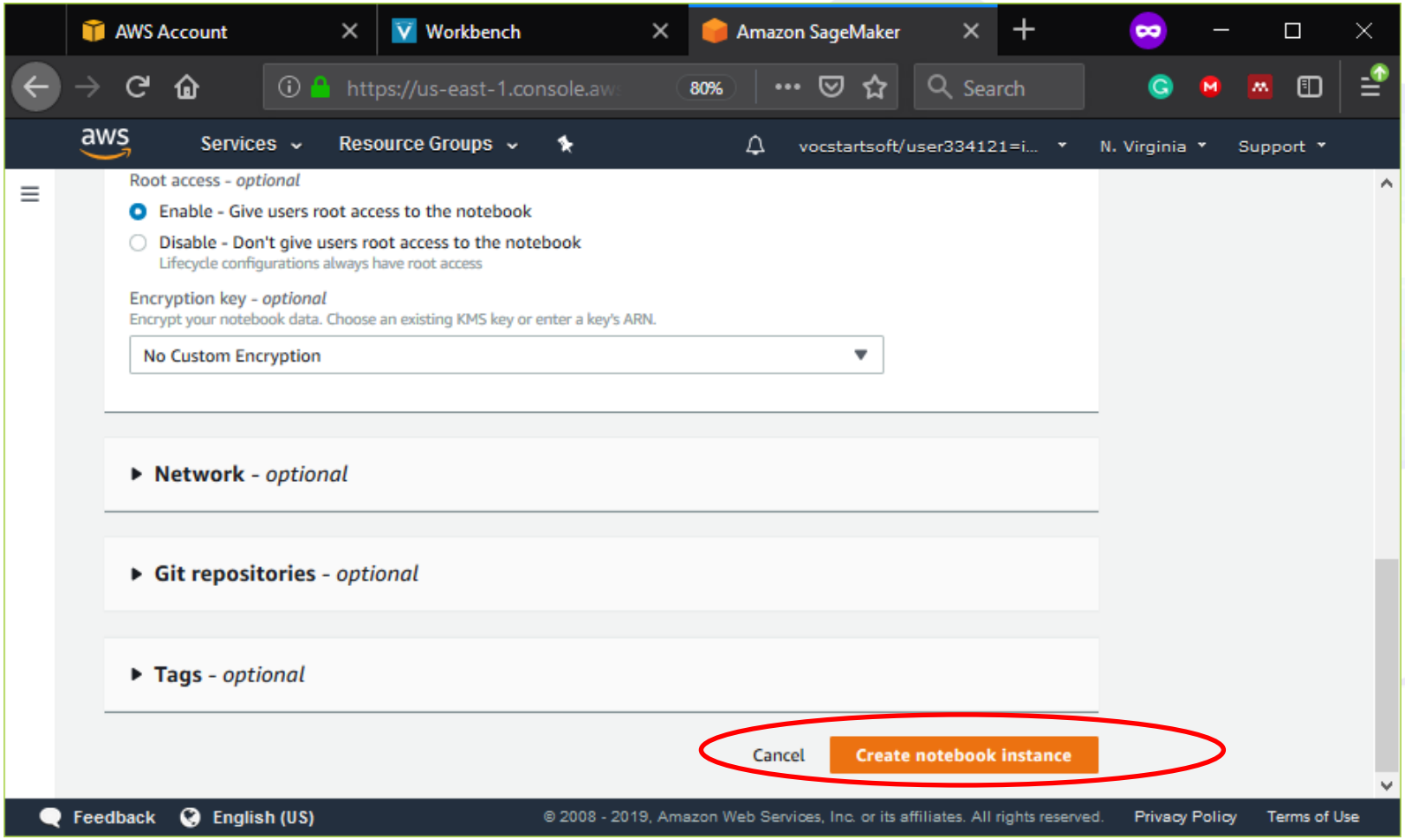
Buat SageMaker

- Paste “Role ARN”, misal “arn:aws:iam::003234194341:role/roleSageMaker-imamcs” ke “Custom IAM role ARN”



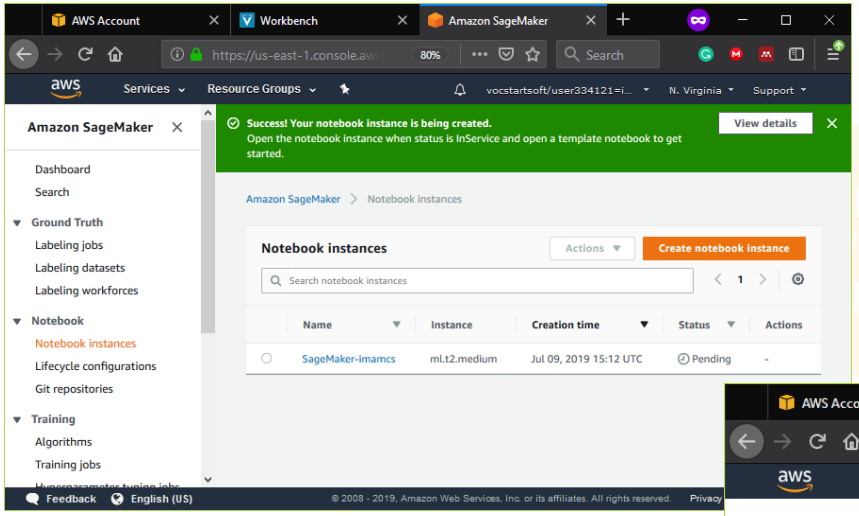
Buat SageMaker

- Scroll ke bawah > Klik “Create notebook Instance”

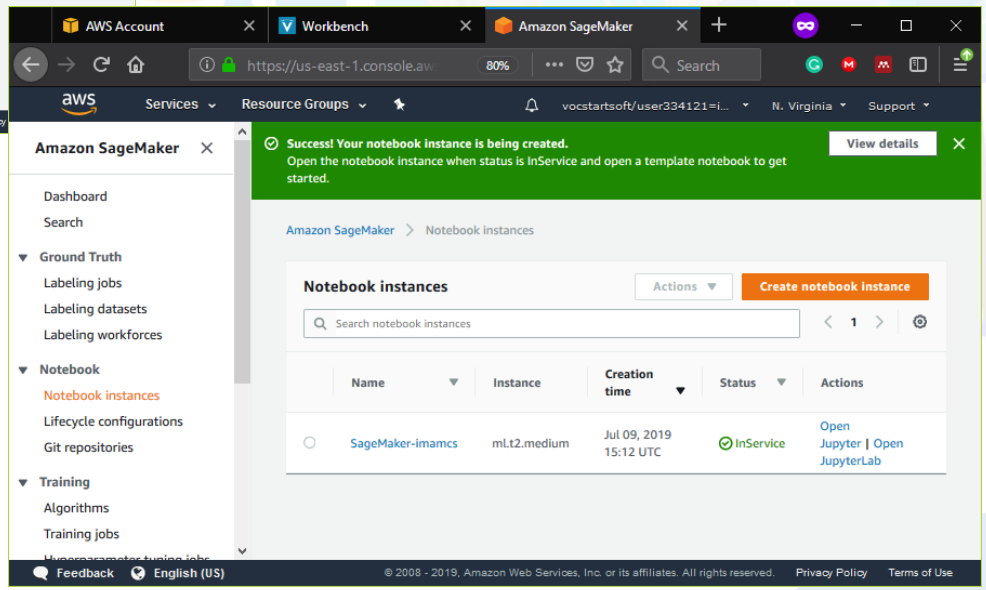


Buat SageMaker

- Notebook berhasil dibuat, perhatikan statusnya masih “Pending”, tunggu beberapa waktu sampai statusnya “InService”

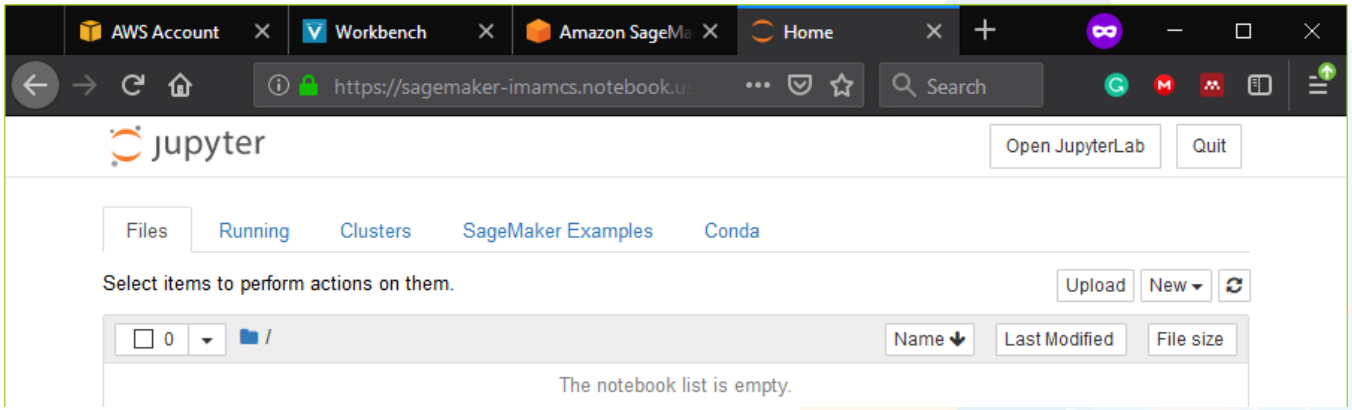


Klik “Open Jupyter”

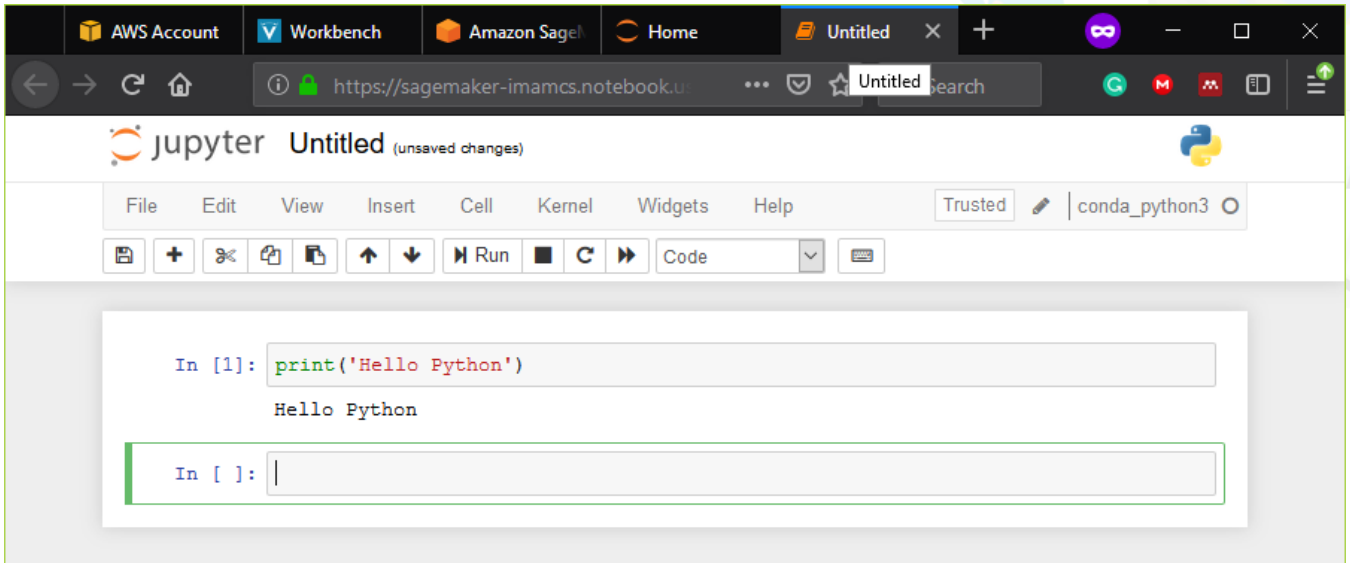


Buat SageMaker

- Jupyter Notebook siap untuk digunakan koding > Klik New > “conda_python3”

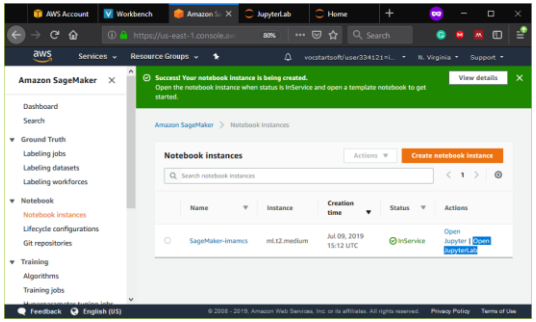


- Coba koding sederhana, berhasil :D

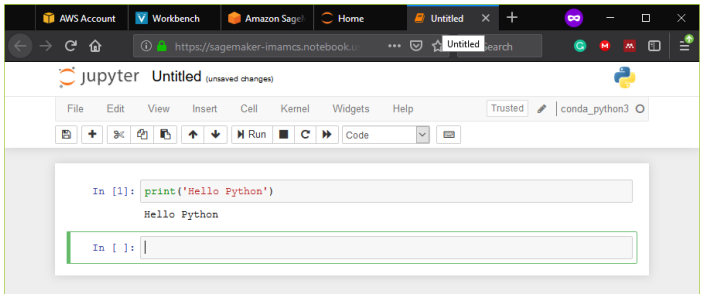


Open Jupyter vs JupyterLab

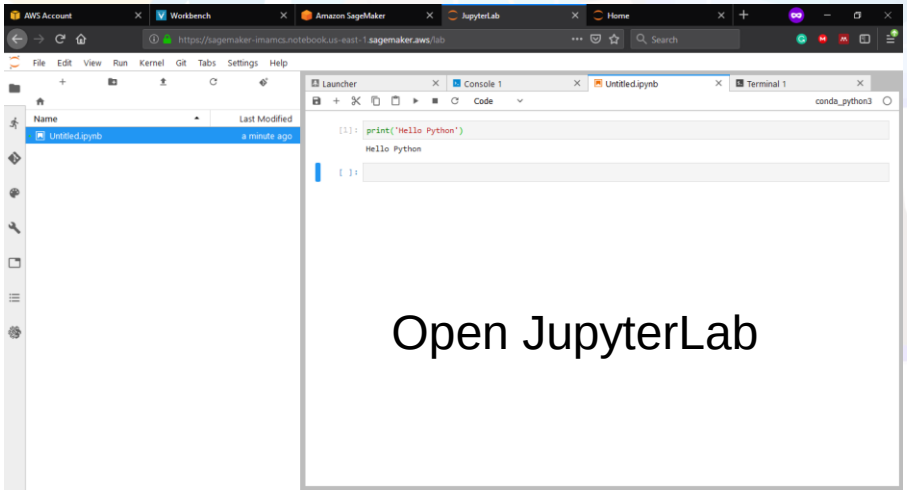
- Coba Klik “Open JupyterLab”



- “Open Jupyter“ vs “Open JupyterLab”



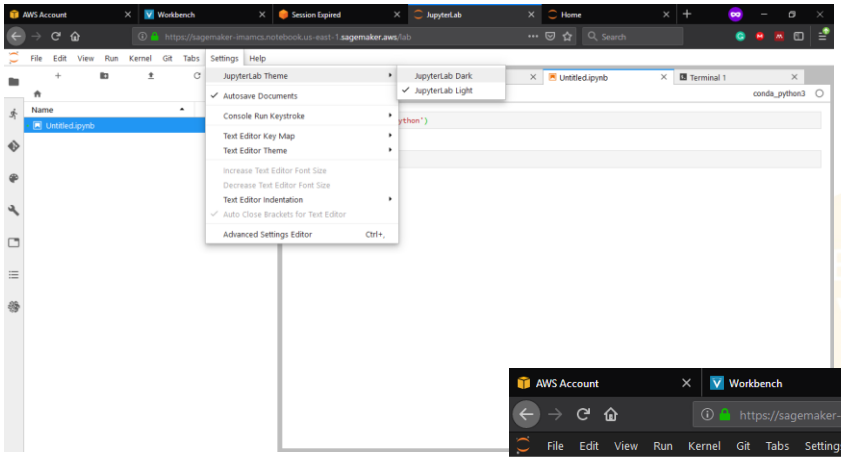
Open Jupyter

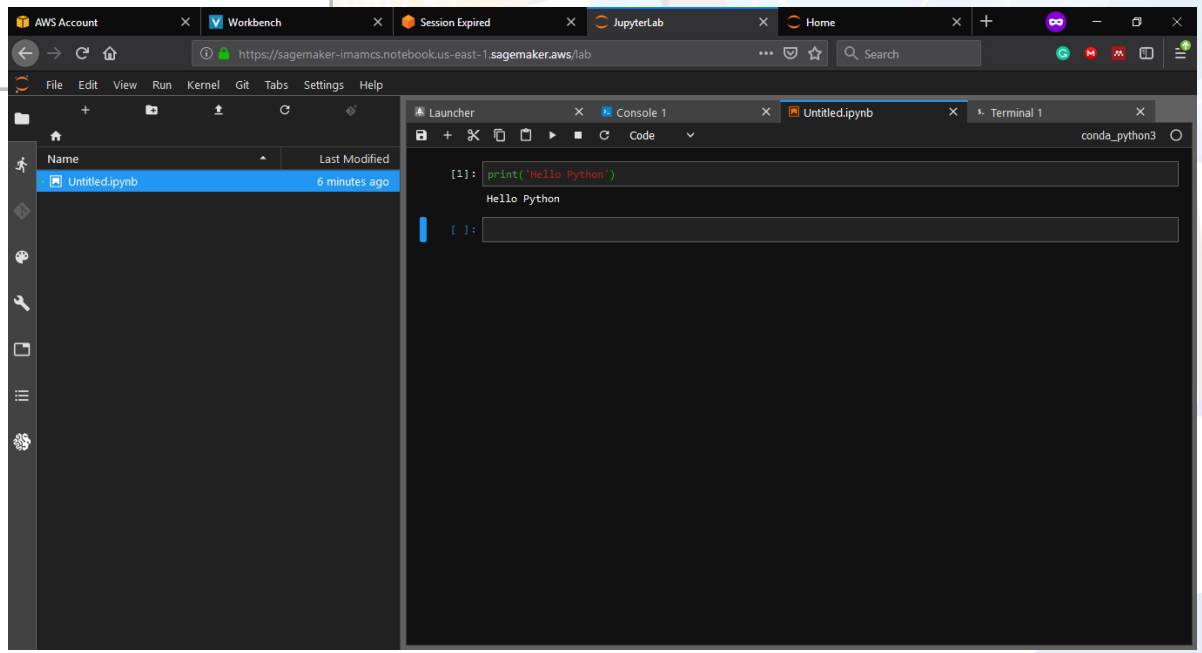


Open JupyterLab

Open Jupyter vs JupyterLab

- Coba ubah theme pada “JupyterLab” > Klik Setting > Klik “JupyterLab Dark”





Open Jupyter vs JupyterLab

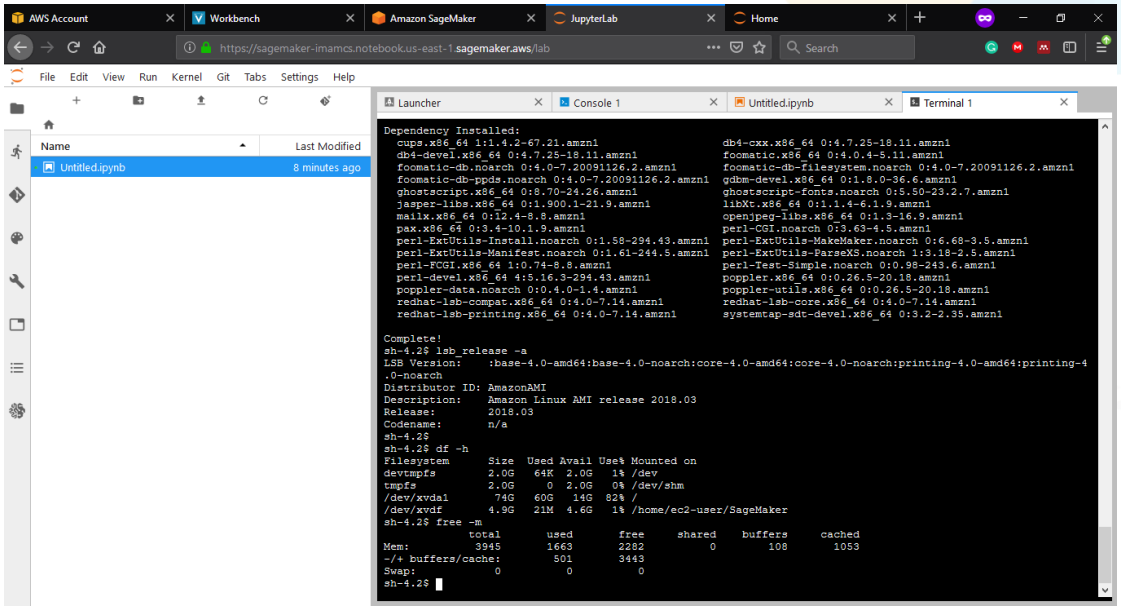
- Pada JupyterLab, coba Klik “File” > Terminal, coba ketikkan

```
$ sudo yum install tmux
$ sudo yum install redhat-lsb

Untuk cek linux yang digunakan, ketikkan
$ lsb_release -a
```

- Cek ukuran filesystem dan RAM

```
$ df -h
$ free -m
```



Open Jupyter vs JupyterLab

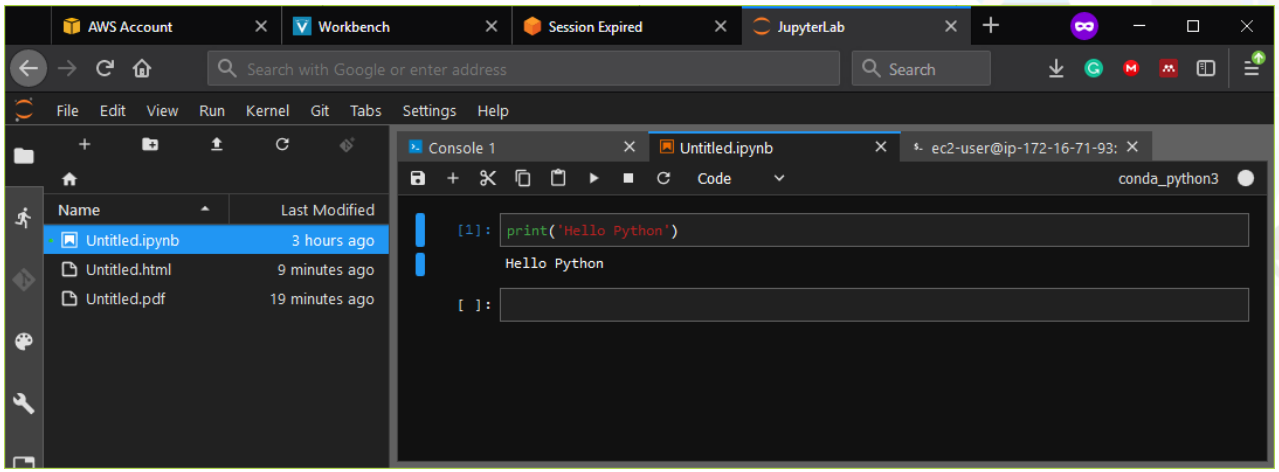
- Pada Terminal, coba ketikkan

Untuk dependensi convert file *.ipynb ke *.pdf, ketikkan
$ pip install nbconvert
$ sudo yum install texlive-*
Tekan enter, jika ada pertanyaan ketika install, ketik “y”, tekan enter kembali

Convert ipynb ke html
$ jupyter nbconvert --format=html SageMaker/Untitled.ipynb

$ wget https://downloads.wkhtmltopdf.org/0.12/0.12.5/wkhtmltox-0.12.5-1.centos6.x86_64.rpm
$ sudo yum install wkhtmltox-0.12.5-1.centos6.x86_64.rpm

Convert html to pdf (misal file Untitled.ipynb berada pada folder “SageMaker”)
$ wkhtmltopdf SageMaker/Untitled.html SageMaker/Untitled.pdf



Remote Linux dari SageMaker

- Untuk me-remote sagemaker via ssh misal dengan Putty/ZOC, Anda dapat menggunakan tunneling, misal yang paling sederhana dengan tool “ngrok” atau lainnya (link step by step “http://bit.ly/2Ldbwli” sebagai contoh untuk remote Linux dari Google CoLab).
- Untuk remote, di AWS, Anda nantinya bisa menggunakan **AWS Glue** untuk membuat titik akhir pengembang (developer endpoint).
- Di **AWS Glue** Anda membuat Sagemaker notebook yang dapat memberi Anda akses untuk terhubung ke python atau Scala Spark REPL, atau lainya melalui ssh, dan Anda juga dapat mengkoneksikan dengan perangkat IDE local Anda, misal PyCharm atau lainnya.
- Akses tersebut memungkinkan Anda untuk mengolah file dengan SFTP dan men-debug jarak jauh.
- Dan jika Anda perlu menginstal dependensi pada SageMaker notebook, Anda dapat menggunakan termina dari Jupyter untuk menginstal dependensi apa pun yang Anda inginkan.

filkom.ub.id

DIGITAL TALENT SCHOLARSHIP 2019

Big Data Analytics

Pengenalan Pemrograman Python

Oleh: Imam Cholissodin | imamcs@ub.ac.id, Ahmad Afif Supianto, Welly Purnomo
Asisten: Guedho Augnifico M. & Aditya Hari Bawono
Fakultas Ilmu Komputer (Filkom) Universitas Brawijaya (UB)

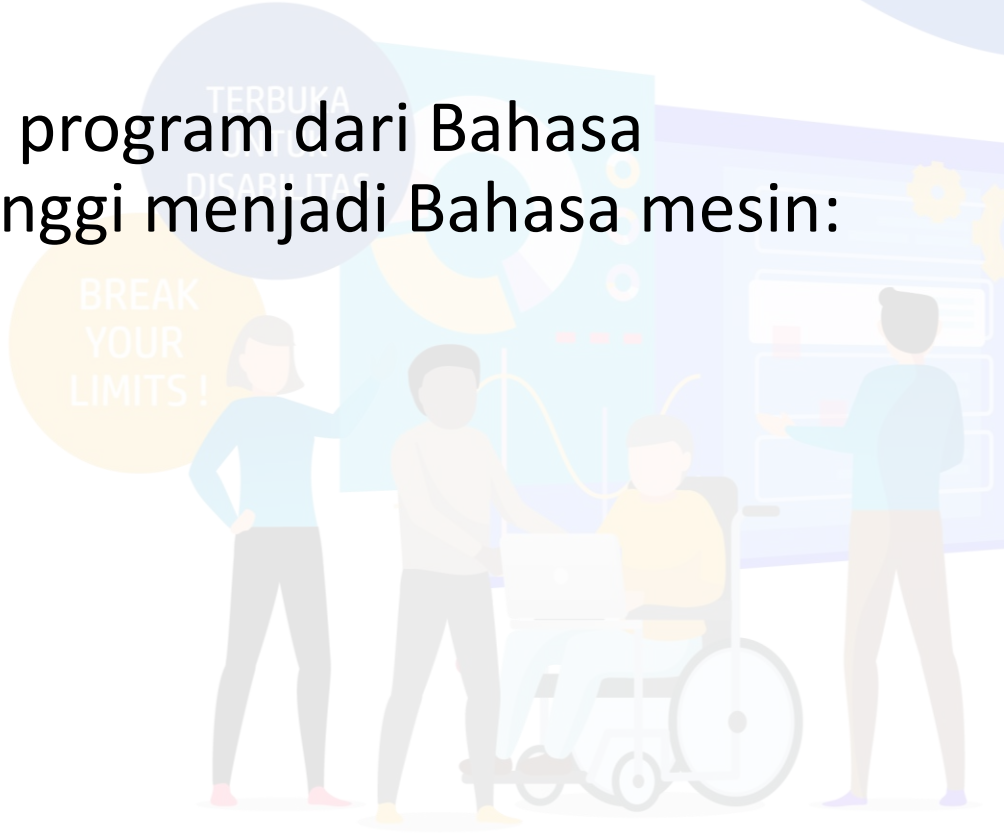
digitalent.kominfo.go.id

Programming - Absolute Basics

- Program membuat computer dapat digunakan
- Tanpa program, komputer tidak lebih dari sebuah benda
- Pemrograman adalah tindakan menulis program menggunakan bahasa pemrograman dalam bentuk perintah sesuai dengan yang diinginkan

Programming - Absolute Basics

- 2 cara untuk mengubah program dari Bahasa pemrograman tingkat tinggi menjadi Bahasa mesin:
  - Compilation
  - Interpretation



Pengenalan Python

- Python – Interpreted language. Python is free
- Widely-used, object-oriented, high-level programming language with dynamic semantics
- Python dibuat oleh Guido van Rossum

PYPL Popularity of Programming Language

Pierre Carbonnelle (2019)

Worldwide, Jun 2019 compared to a year ago:

Rank	Change	Language	Share	Trend
1		Python	28.08 %	+4.7 %
2		Java	20.51 %	-1.8 %
3		Javascript	8.29 %	-0.2 %
4	↑	C#	7.41 %	-0.5 %
5	↓	PHP	6.96 %	-1.2 %
6		C/C++	5.76 %	-0.4 %
7		R	4.15 %	-0.0 %
8		Objective-C	2.82 %	-0.6 %
9		Swift	2.36 %	-0.4 %
10		Matlab	1.95 %	-0.3 %
11	↑	TypeScript	1.69 %	+0.2 %
12	↓	Ruby	1.42 %	-0.3 %
13	↑↑↑	Kotlin	1.4 %	+0.5 %
14	↓	VBA	1.33 %	-0.1 %
15	↑↑	Go	1.19 %	+0.3 %
16	↓	Scala	1.13 %	-0.1 %
17	↓↓↓	Visual Basic	1.09 %	-0.1 %
18		Perl	0.59 %	-0.2 %

The `print()` Function - *Instructions*

Lihat efek dari kode berikut

```
1  print("The itsy bitsy spider climbed up the waterspout.")
2  print("Down came the rain and washed the spider out.")
```

```
1  print("The itsy bitsy spider\nclimbed up the waterspout.")
2  print()
3  print("Down came the rain\nand washed the spider out.")
```

```
1  print("My", "name", "is", sep="_", end="*")
2  print("Monty", "Python.", sep="*", end="*\n")
```

- Fungsi **`print`** merupakan built-in function

		Built-in Functions		
abs()	delattr()	hash()	memoryview()	set()
all()	dict()	help()	min()	setattr()
any()	dir()	hex()	next()	slice()
ascii()	divmod()	id()	object()	sorted()
bin()	enumerate()	input()	oct()	staticmethod()
bool()	eval()	int()	open()	str()
breakpoint()	exec()	isinstance()	ord()	sum()
bytearray()	filter()	issubclass()	pow()	super()
bytes()	float()	iter()	print()	tuple()
callable()	format()	len()	property()	type()
chr()	frozenset()	list()	range()	vars()
classmethod()	getattr()	locals()	repr()	zip()
compile()	globals()	map()	reversed()	__import__()

Tipe Data - Numerik

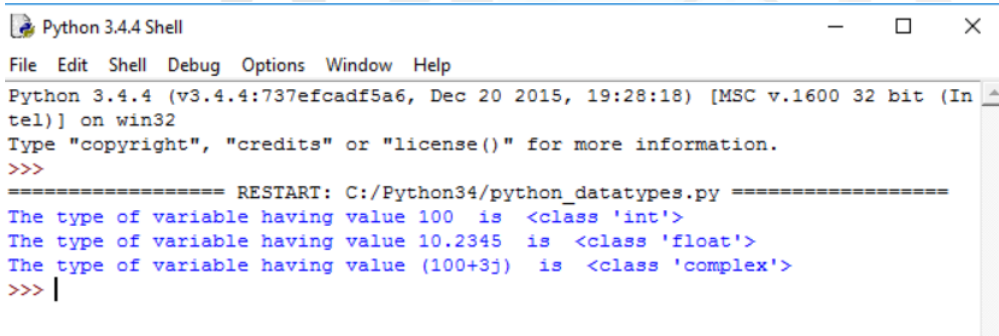
1. Int – bilangan bulat dengan panjang tidak terbatas
2. Long - merupakan long integers (ada pada Python 2, tidak digunakan lagi di Python 3)
3. Float - bilangan pecahan
4. Complex - bilangan kompleks

```
myComplex = 3 + 4j
```

```
#create a variable with integer value.
a=100
print("The type of variable having value", a, " is ",
type(a))

#create a variable with float value.
b=10.2345
print("The type of variable having value", b, " is ",
type(b))

#create a variable with complex value.
c=100+3j
print("The type of variable having value", c, " is ",
type(c))
```



Untuk memeriksa tipe data pada suatu variabel, kita bisa menggunakan fungsi type()

Tipe Data - *String*

1. Char: karakter, contoh 'R'
2. *String*: kumpulan karakter, contoh "aku lagi makan"

```
nama = "Ivan"
jenis_kelamin = 'L'
alamat = """
    Jl. Suka Karya, No 32. RT Kode,
    Kelurahan Mawar, Jakarta
"""
agama = 'islam'
```

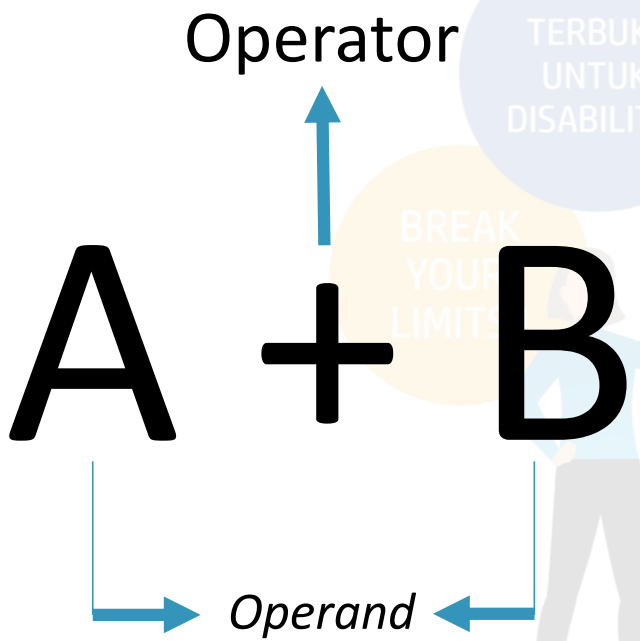
Penulisan tipe data teks harus diapit dengan tanda petik. Bisa menggunakan petik tunggal ('...'), ganda ("...")

Tipe Data - *Boolean*

- Tipe data *Boolean* adalah tipe data yang hanya memiliki dua nilai yaitu True dan False
- Pada penulisan True dan False, huruf pertamanya harus kapital

```
bergerak = True
nyala = 1 #sebenarnya tipenya int, tapi bisa juga menjadi bool
```

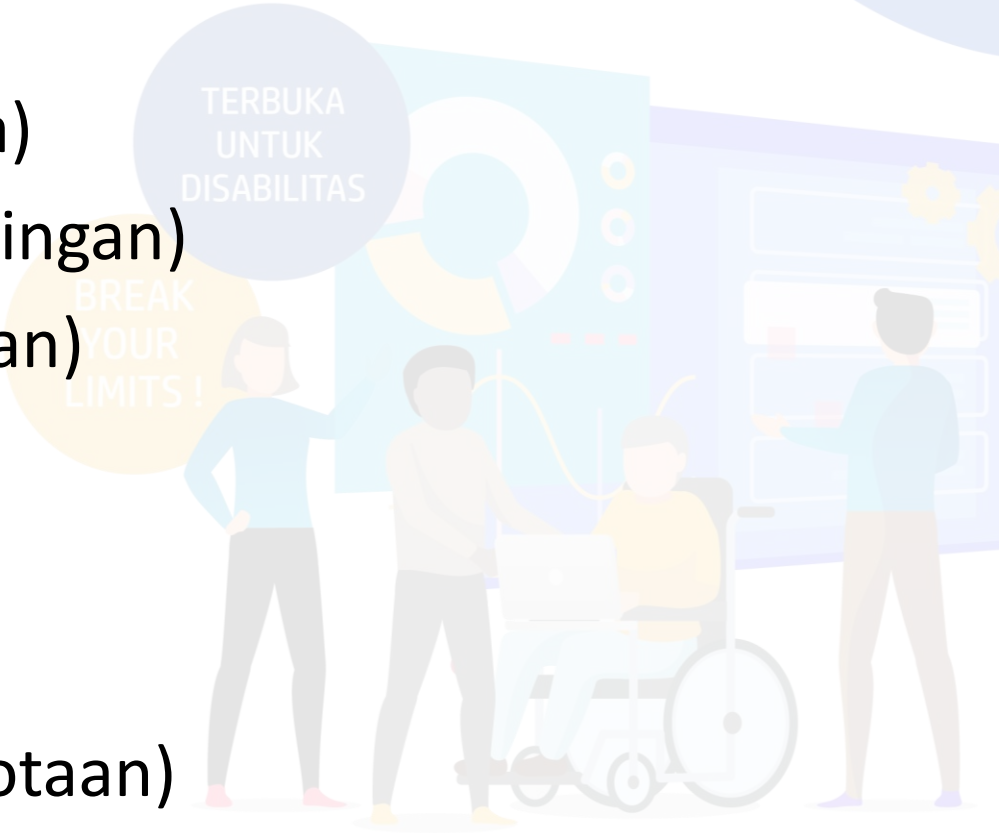
Operators - Data Manipulation Tools



- Ingat: data dan operator ketika dikoneksikan bersama membentuk **expressions**

Kategori Operator

1. Arithmetic (aritmetika)
2. Comparison (perbandingan)
3. Assignment (penugasan)
4. Logical (logika)
5. Bitwise
6. Identity (identitas)
7. Membership (keanggotaan)



Operator - *Arithmetic*

Python menyediakan akses ke *library* untuk membantu menyelesaikan perhitungan matematika yang kompleks

Operator	Nama dan Fungsi	Contoh
+	Penjumlahan, menjumlahkan 2 buah operand	x + y
–	Pengurangan, mengurangkan 2 buah operand	x – y
*	Perkalian, mengalikan 2 buah operand	x * y
/	Pembagian, membagi 2 buah operand	x / y
**	Pemangkatan, memangkatkan bilangan	x **y
//	Pembagian bulat, menghasilkan hasil bagi tanpa koma	x // y
%	Modulus, menghasilkan sisa pembagian 2 bilangan	x % y

Operator - *Comparison*

Membandingkan suatu nilai dengan nilai lainnya dan menampilakan hasil perbandingan (True dan False)

Operator	Nama dan Fungsi	Contoh
>	Lebih besar dari – Hasilnya True jika nilai sebelah kiri lebih besar dari nilai sebelah kanan	x > y
<	Lebih kecil dari – Hasilnya True jika nilai sebelah kiri lebih kecil dari nilai sebelah kanan	x < y
==	Sama dengan – Hasilnya True jika nilai sebelah kiri sama dengan nilai sebelah kanan	x == y
!=	Tidak sama dengan – Hasilnya True jika nilai sebelah kiri tidak sama dengan nilai sebelah kanan	x != y
>=	Lebih besar atau sama dengan – Hasilnya True jika nilai sebelah kiri lebih besar atau sama dengan nilai sebelah kanan	x >= y
<=	Lebih kecil atau sama dengan – Hasilnya True jika nilai sebelah kiri lebih kecil atau sama dengan nilai sebelah kanan	x <= y

Operator – *Assignment*

Operator *assignment* menempatkan nilai dalam suatu variable

Operator	Penjelasan	Contoh
=	Menugaskan nilai yang ada di kanan ke operand yang ada di sebelah kiri	c = a + b menugaskan a + b ke c
+=	Menambahkan operand yang di kanan dengan operand yang ada di kiri dan hasilnya di tugaskan ke operand yang di kiri	c += a sama dengan c = c + a
-=	Mengurangi operand yang di kanan dengan operand yang ada di kiri dan hasilnya di tugaskan ke operand yang di kiri	c -= a sama dengan c = c + a
*=	Mengalikan operand yang di kanan dengan operand yang ada di kiri dan hasilnya di tugaskan ke operand yang di kiri	c *= a sama dengan c = c * a
/=	Membagi operand yang di kanan dengan operand yang ada di kiri dan hasilnya di tugaskan ke operand yang di kiri	c /= a sama dengan c = c * a
**=	Memangkatkan operand yang di kanan dengan operand yang ada di kiri dan hasilnya ditugaskan ke operand yang di kiri	c **= a sama dengan c = c ** a
//=	Melakukan pembagian bulat operand di kanan terhadap operand di kiri dan hasilnya disimpan di operand yang di kiri	c //= a sama dengan c = c // a
%=	Melakukan operasi sisa bagi operand di kanan dengan operand di kiri dan hasilnya di simpan di operand yang di kiri	c %= a sama dengan c = c % a

Operator – *Logical*

Operator logika adalah operator yang digunakan untuk melakukan operasi logika

Operator	Penjelasan	Contoh
and	Hasilnya adalah True jika kedua operandnya bernilai benar	x and y
or	Hasilnya adalah True jika salah satu atau kedua operandnya bernilai benar	x or y
not	Hasilnya adalah True jika operandnya bernilai salah (kebalikan nilai)	not x

Operator – *Bitwise*

- Operator bitwise adalah operator yang melakukan operasi bit terhadap operand
- Operator ini beroperasi bit per bit sesuai dengan namanya
- Sebagai misal, angka 2 dalam bit ditulis 10 dalam notasi biner dan angka 7 ditulis 111
- Pada tabel di bawah ini, misalkan x = 10 (0000 1010) dalam biner dan y = 4 (0000 0100) dalam biner

Operator	Nama	Contoh
&	Bitwise AND	x& y = 0 (0000 0000)
|	Bitwise OR	x | y = 14 (0000 1110)
~	Bitwise NOT	~x = -11 (1111 0101)
^	Bitwise XOR	x ^ y = 14 (0000 1110)
>>	Bitwise right shift	x>> 2 = 2 (0000 0010)
<<	Bitwise left shift	x<< 2 = 40 (0010 1000)

Operator – Identitas

- Operator identitas adalah operator yang memeriksa apakah dua buah nilai (atau variabel) berada pada lokasi memori yang sama

Operator	Penjelasan	Contoh
is	True jika kedua operand identik (menunjuk ke objek yang sama)	x is True
is not	True jika kedua operand tidak identik (tidak merujuk ke objek yang sama)	x is not True

Operator – Membership

- Operator keanggotaan adalah operator yang digunakan untuk memeriksa apakah suatu nilai atau variabel merupakan anggota atau ditemukan di dalam suatu data (string, list, tuple, set, dan dictionary)

Operator	Penjelasan	Contoh
in	True jika nilai/variabel ditemukan di dalam data	5 in x
not in	True jika nilai/variabel tidak ada di dalam data	5 not in x

Operator Precedence

Jika bekerja dengan multiple operator, penting untuk menentukan operator mana yang diproses pertama kali

Table 6-9 Python Operator Precedence

Operator	*Description*
`()`	You use parentheses to group expressions and to override the default precedence so that you can force an operation of lower precedence (such as addition) to take precedence over an operation of higher precedence (such as multiplication).
`**`	Exponentiation raises the value of the left operand to the power of the right operand.
`~ + -`	Unary operators interact with a single variable or expression.
`* / % //`	Multiply, divide, modulo, and floor division.
`+ -`	Addition and subtraction.
`>> <<`	Right and left bitwise shift.
`&`	Bitwise AND.
`^ |`	Bitwise exclusive OR and standard OR.
`<= < > >=`	Comparison operators.
`== !=`	Equality operators.
`= %= /= //= -= += *= **=`	Assignment operators.
`Is` `is not`	Identity operators.
`In` `not in`	Membership operators.

Variabel dan Tipe Data

- Variabel merupakan tempat menyimpan data, sedangkan tipe data adalah jenis data yang terseimpan dalam variabel.
- Karakter pada nama variabel bersifat **sensitif (*case-sensitif*).** Artinya huruf besar dan kecil dibedakan. Misalnya, variabel_Ku dan variabel_ku, keduanya adalah variabel yang berbeda
- Nama variabel tidak boleh menggunakan kata kunci yang sudah ada dalam Python seperti `if`, `while`, `for`, dsb.


```
nama_variabel = <nilai>
```

```
variabel_ku = "ini isi variabel"
variabel2 = 20
```

Input Output – *Reading Keyboard Input*

- Fungsi input ([prompt]) meminta input atau masukan dari *user*
- Bila kita menginput bilangan, misalnya *integer* lewat fungsi `input()`, maka hasil input tersebut adalah *string* dan bukan *integer*. Kita harus mengubahnya terlebih dahulu menjadi tipe *integer* menggunakan fungsi `int()`

```
name = input('Masukkan nama: ')
print(name)
```

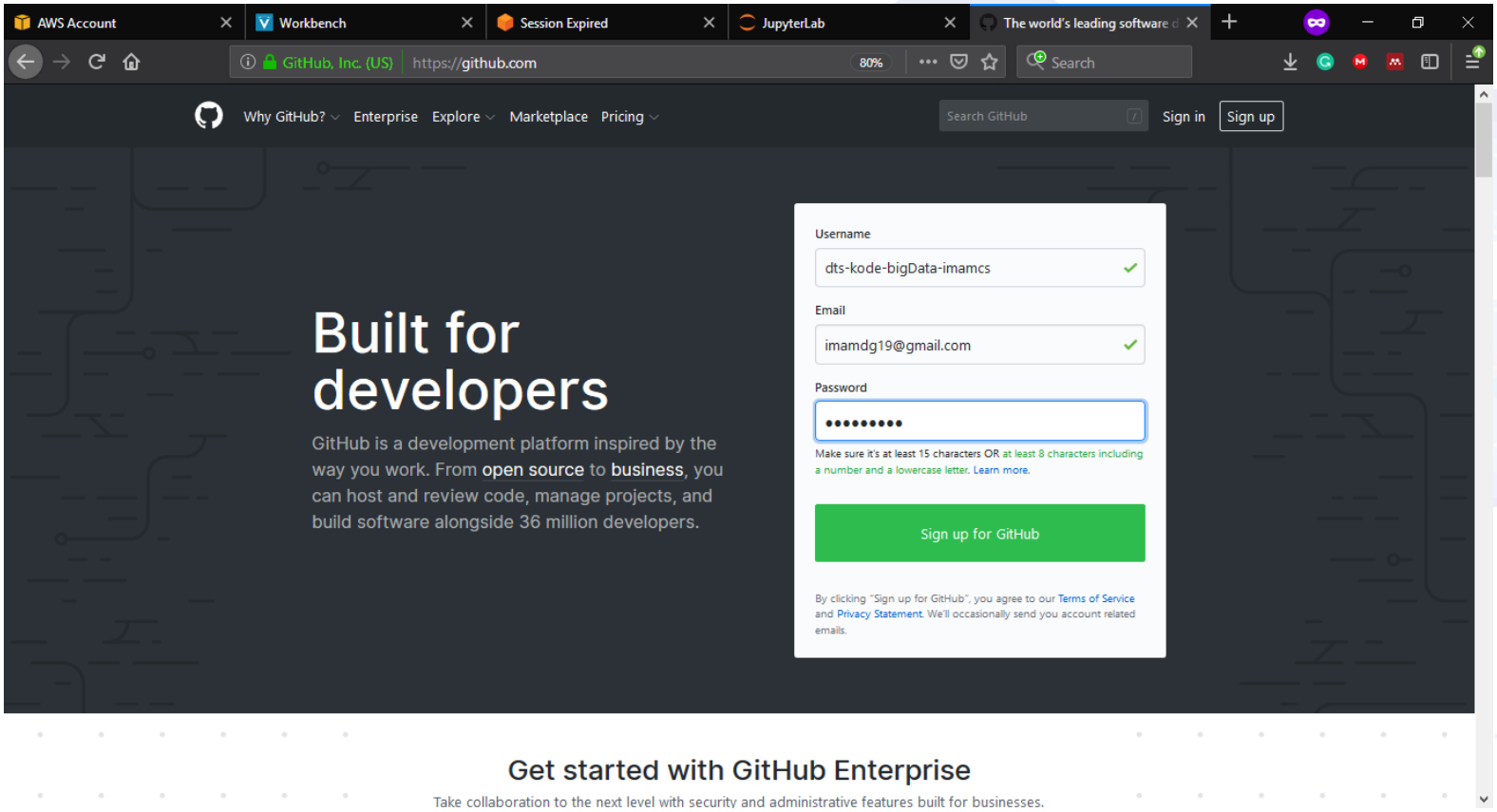
```
Masukkan nama: budi
budi
```

```
panjang = input('Masukkan nilai panjang: ')
lebar = input('Masukkan nilai lebar: ')
luas = int(panjang) * int(lebar)
print("Luas =", luas)
```

```
Masukkan nilai panjang: 7
Masukkan nilai lebar: 8
Luas = 56
```

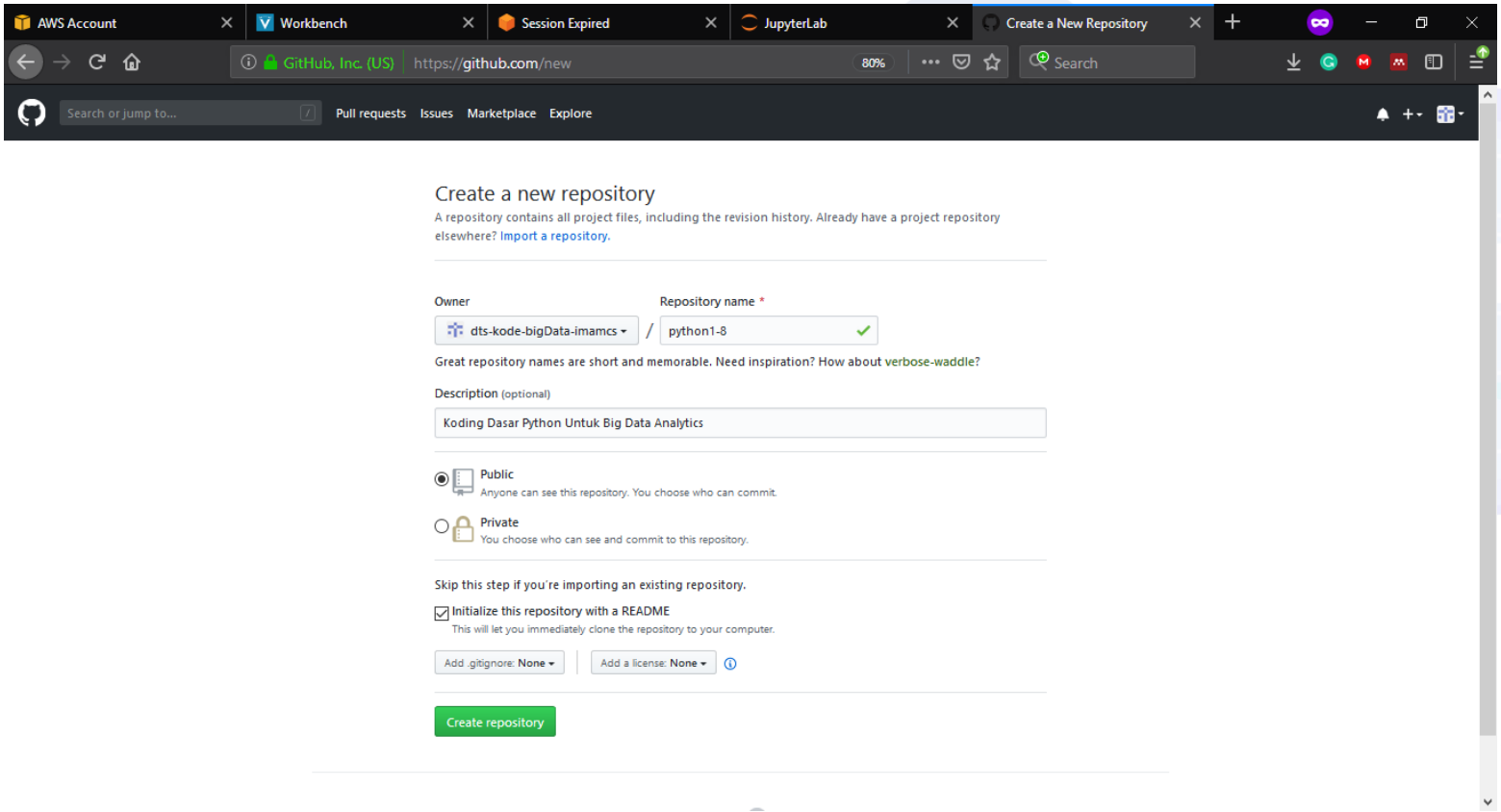
Clone kode dari github ke Jupyter Lab

- Register ke github (jika sudah punya akun github, silahkan langsung login)



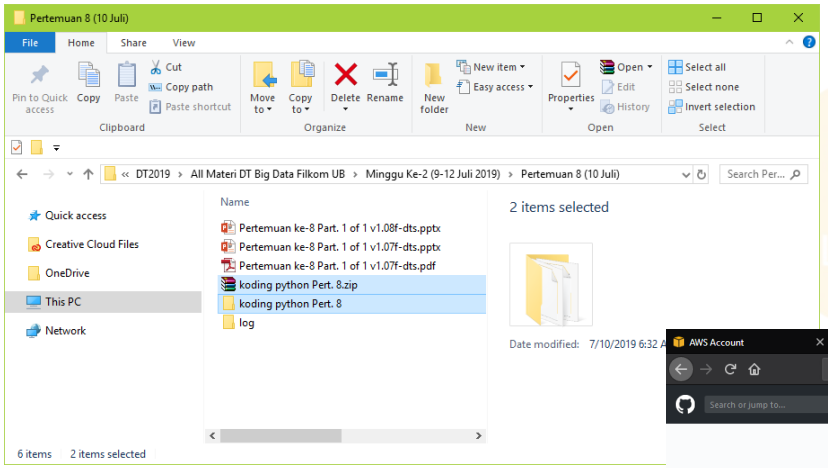
Clone kode dari github ke Jupyter Lab

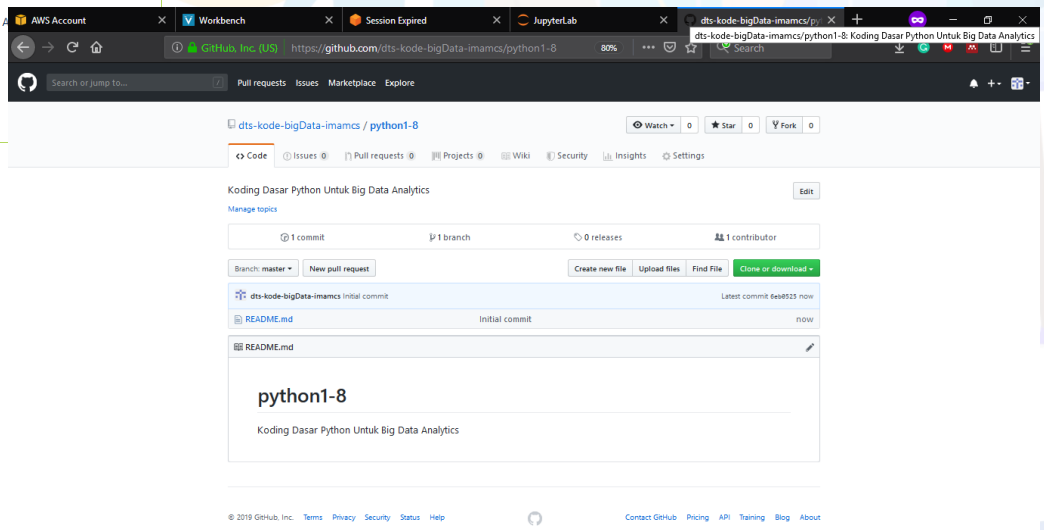
- Create repository, misal “python1-7”



Clone kode dari github ke Jupyter Lab

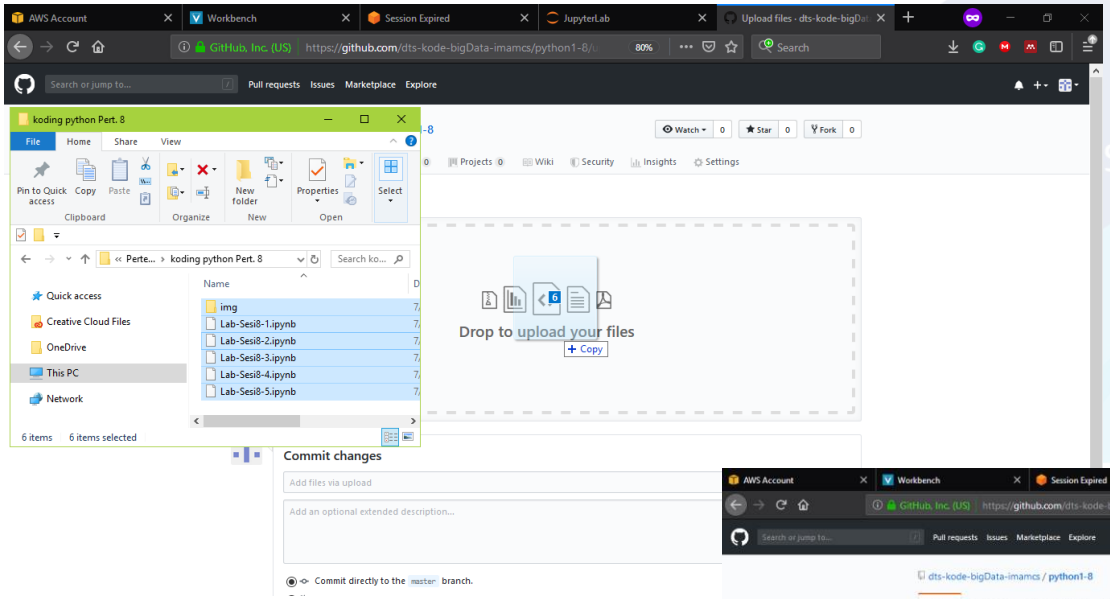
- Download file “koding python Pert. 7.zip” dari link “http://bit.ly/2YKicRs”, lalu ekstrak di lokal komputer anda > Upload hasil ekstrak tersebut ke github Anda > dengan klik “Upload files”



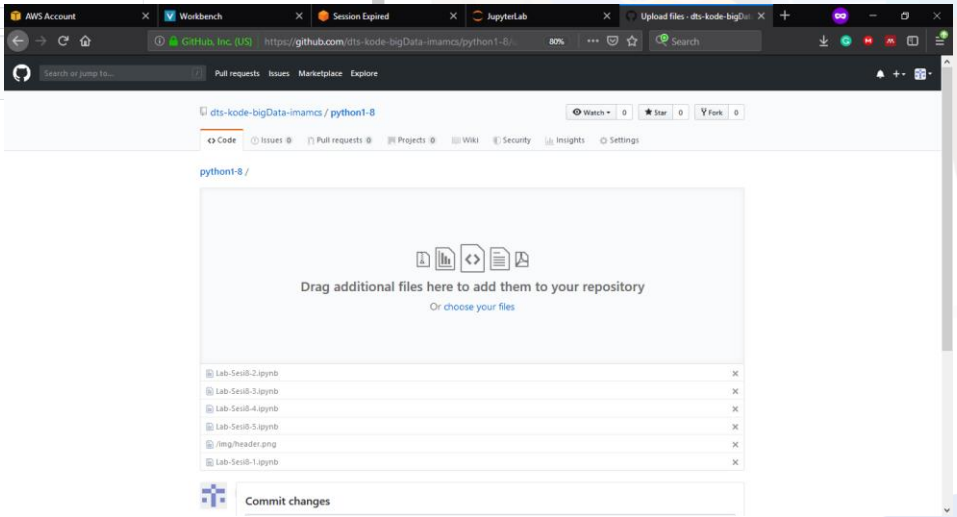


Clone kode dari github ke Jupyter Lab

- Drag & Drop file (tunggu beberapa waktu) > Done > Klik “Commit Change”

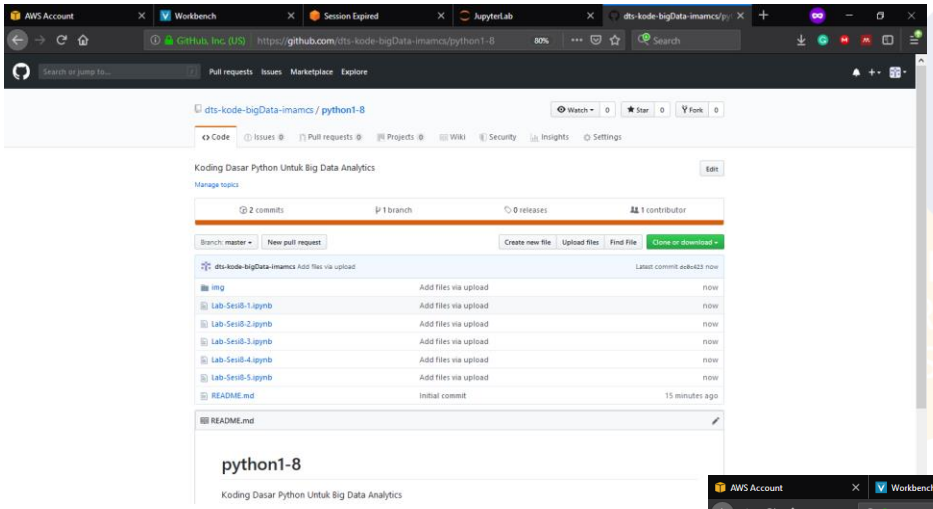


Done!

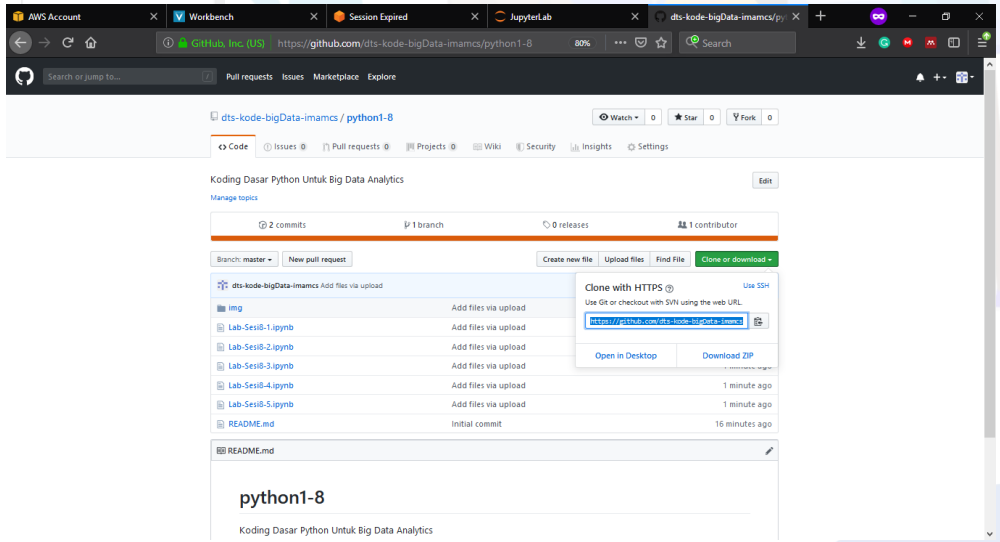


Clone kode dari github ke Jupyter Lab

- Hasil Klik “Commit Change” > Klik “Clone or download”

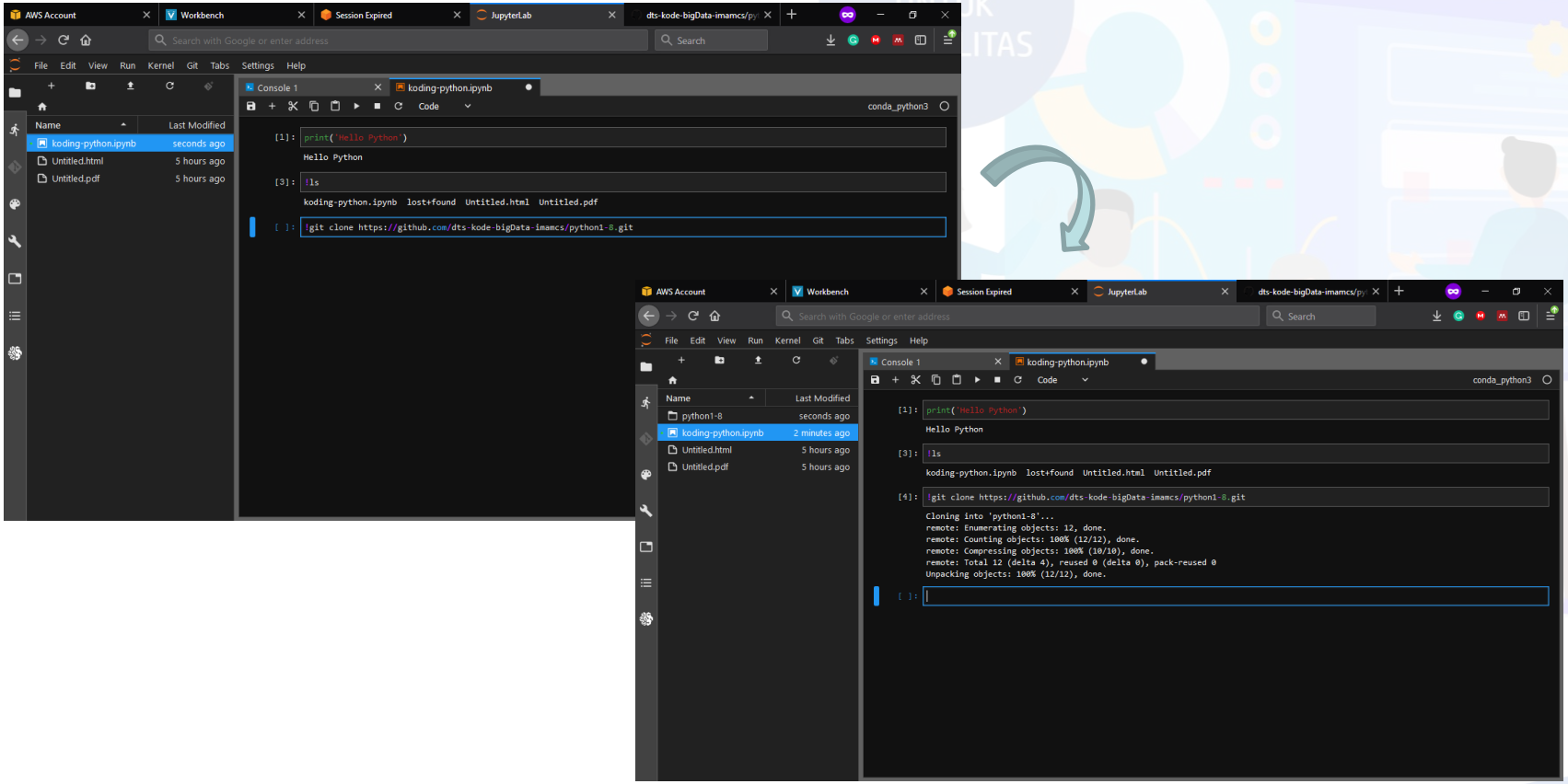


Copy link yang muncul, misal:
https://github.com/dts-kode-bigData-imamcs/python1-8.git



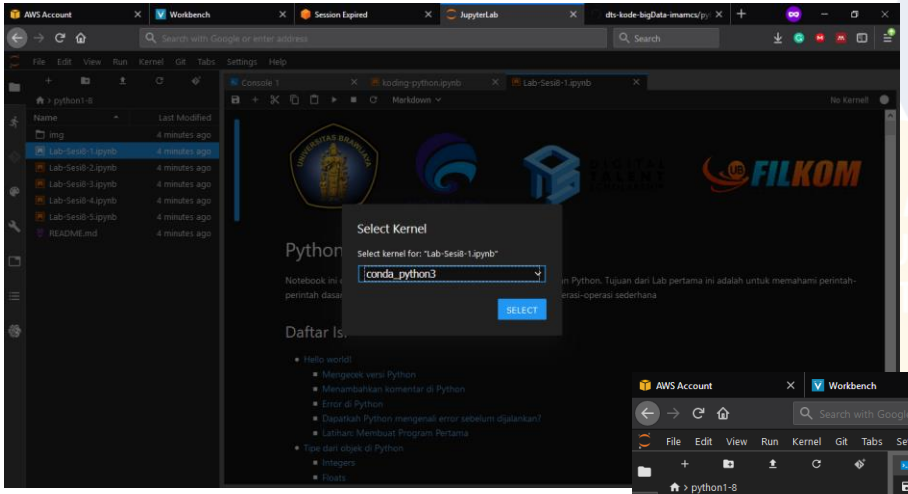
Clone kode dari github ke Jupyter Lab

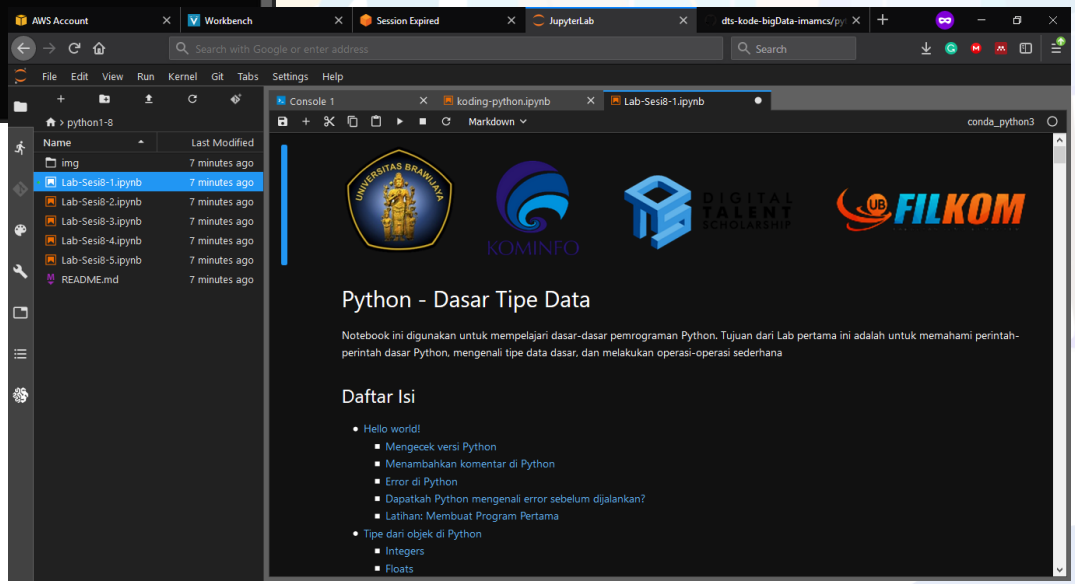
- Pada JupyterLab, ketikkan "!git clone https://github.com/dts-kode-bigData-imamcs/python1-8.git", lalu tekan shift + enter, maka otomatis koding github Anda sudah ter-*clone* ke JupyterLab.



Clone kode dari github ke Jupyter Lab

- Pada JupyterLab, buka folder “python1-8”, buka setiap file *.ipynb, dan set dengan kernel “conda_python3” > Klik SELECT. Happy Coding :D

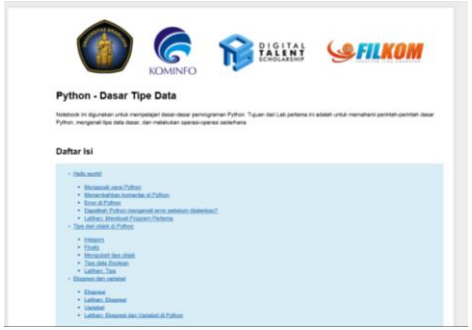




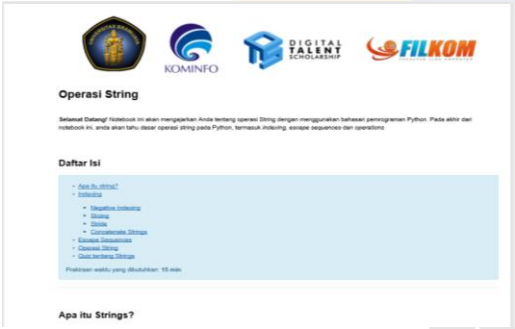
Latihan langsung di Kelas Ke-1 & Pembahasan
Link kode “http://bit.ly/2YKicRs”

Silahkan dicoba dijalankan dengan Jupyter notebook yang Anda buat sebelumnya di Ubuntu 16.04 atau dengan SageMaker notebook (JupyterLab) yang baru Anda buat hari ini.

Lab-Sesi8-1

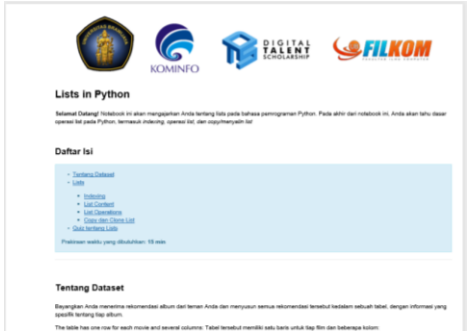


Lab-Sesi8-2

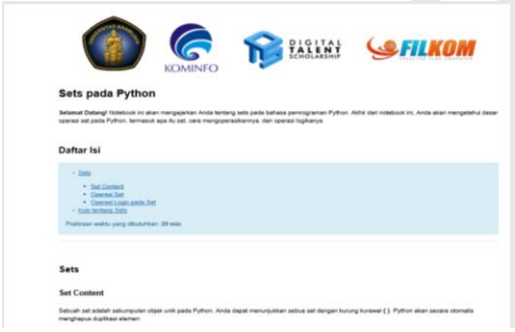


Lab-Sesi8-3

Lab-Sesi8-4



Lab-Sesi8-5



Latihan langsung di Kelas Ke-2 & Pembahasan

1. Buatlah code untuk menyelesaikan problem matematika sederhana berikut:

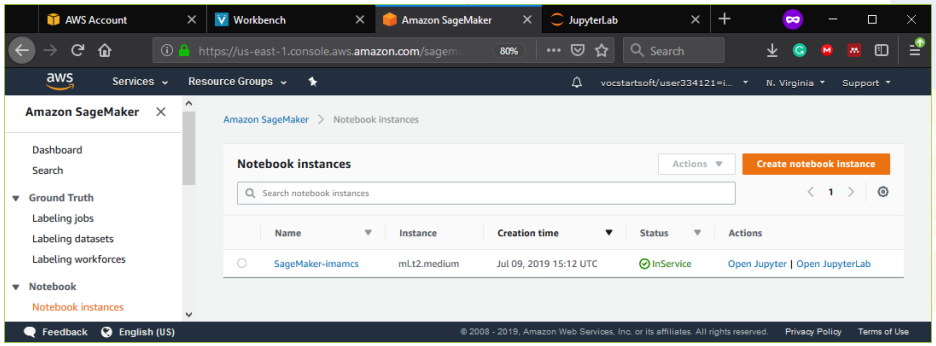
$$c = \sqrt{a^2 + b^2}$$

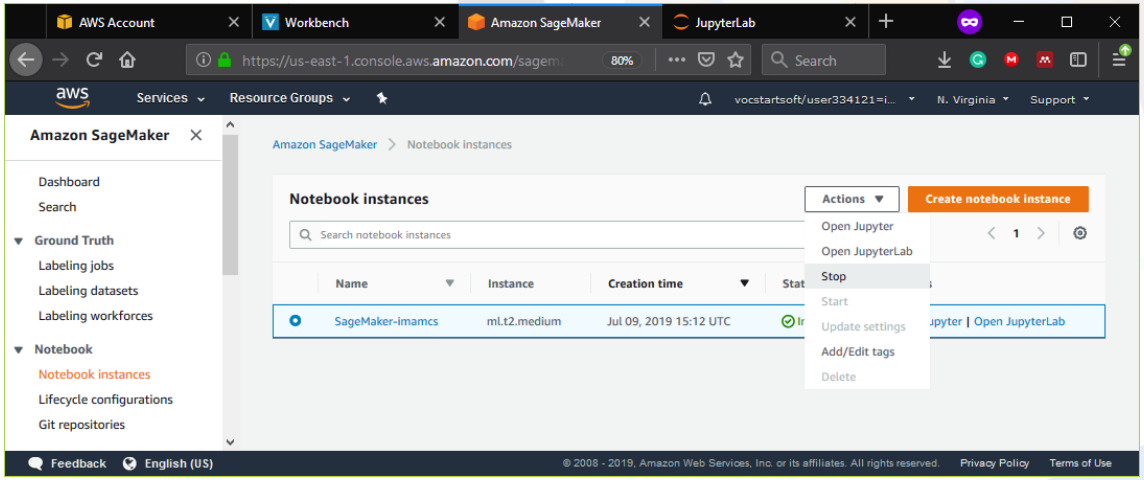
2. Buatlah program yang dapat membandingkan 2 buah nilai apakah sama besar, lebih kecil atau tidak

3. Buatlah program Kalkulator sederhana

Cara Stop SageMaker

- Klik “Services” > Pilih “Amazon SageMaker” > Klik “Notebook instances” > Klik simbol lingkaran di samping “SageMaker-imamcs” > Klik “Actions” > Pilih “Stop”





Tugas Individu

1. Buatlah rangkuman materi di atas dengan cara berikut:
   - Lakukan ulang materi di atas seperti di *all* slide, termasuk melampirkan file *.ipynb yang sudah di-*convert* ke pdf dari hasil "Latihan langsung di Kelas Ke-1 dan Latihan langsung di Kelas Ke-2"

lalu simpan dalam satu file PDF dengan nama file
"[Nama Lengkap Mhs]-[Pert. Ke-2.1/..]"

Dan dari semua tugas dalam 1 minggu di-merger, dengan nama file seperti:
"[Nama Lengkap Mhs]-[Minggu Ke-1/2/..]", lalu cek plagiasi diturnitin dari hasil merger tersebut
> Register ke turnitin
> Masukkan **id class**: 21563495 & **enrool key**: filkomub9302

filkom.ub.ac.id

DIGITAL TALENT SCHOLARSHIP 2019

Big Data Analytics

Terimakasih

Oleh: Imam Cholissodin | imamcs@ub.ac.id, Putra Pandu Adikara, Sufia Adha Putri
Asisten: Guedho, Sukma, Anshori, Aang dan Gusti
Fakultas Ilmu Komputer (Filkom) Universitas Brawijaya (UB)

Materi Tambahan

- Menyelesaikan course “PY0101EN” pada cognitiveclass.ai --> Optional
  - Module 1 Lab – Write your first Python code!
  - Module 1 Lab – String Operations