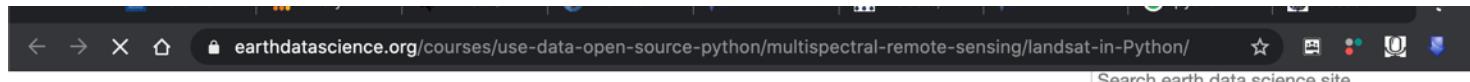


Dasar Python

Kenapa Python?

Remote Sensing



Tutorials Courses Workshops Tools Blog About

Earth Data Analytics Online Certificate

[Enroll now! Learn more.](#)

Home/Courses /Use data open source python /Multispectral remote sensing /Work with Landsat Remote Sensing Data in Python

Intermediate earth data science textbook

UNITS

SECTION 1

TIME SERIES DATA IN PYTHON

Chapter 1: Time Series Data in Pandas

Chapter 1.5: Flood Returns Period Analysis in Python

SECTION 2

INTRO TO SPATIAL VECTOR DATA IN PYTHON

Chapter 2: Spatial Data in Python

Chapter 3: Processing Spatial Vector Data in Python

Lesson 1. Work with Landsat Remote Sensing Data in Python

Leah Wasser

Learn How to Work With Landsat Multispectral Remote Sensing Data in Python - Intermediate earth data science textbook course module

Welcome to the first lesson in the [Learn How to Work With Landsat Multispectral Remote Sensing Data in Python](#) module. Learn how to work with Landsat multi-band raster data stored in .tif format in Python using Rasterio

[IN THIS CHAPTER](#)

Search earth data science site

Earth Lab

Tutorials Courses Workshops Tools Blog About

Earth Data Analytics Online Certificate

[Enroll now! Learn more.](#)

Home/Tags /Lessons: Use Remote sensing data in R or Python

Lessons: Use Remote sensing data in R or Python

About Remote Sensing Data

Remote sensing is the science of studying things without touching them. You can use remote sensing systems, to study how Earth systems change over time. For example, scientists use, high powered cameras, not unlike the camera in your smartphone, mounted on airplanes and satellites to capture images of the earth as it changes over time. Other sensors such as lidar (light detection and ranging) are used to collect height data which can be used to measure how trees and forests and even development changes over time.

Active vs Passive Remote Sensing

There are two types of remote sensing sensors: active and passive sensors. Passive sensors measure existing energy, often from the sun. The camera in your smartphone or iPad is an example of a passive remote sensing sensor. To capture a picture, this camera records sunlight, reflected off objects. In contrast,

BY SCIENCE TOPIC

remote sensing: 66
landsat 23
modis 19
lidar 27
nasa 4
multispectral-remote-sensing 10

earth science: 59

social science: 7

time series: 22

data exploration and analysis: 40

spatial data and gis: 102

reproducible science and programming: 208

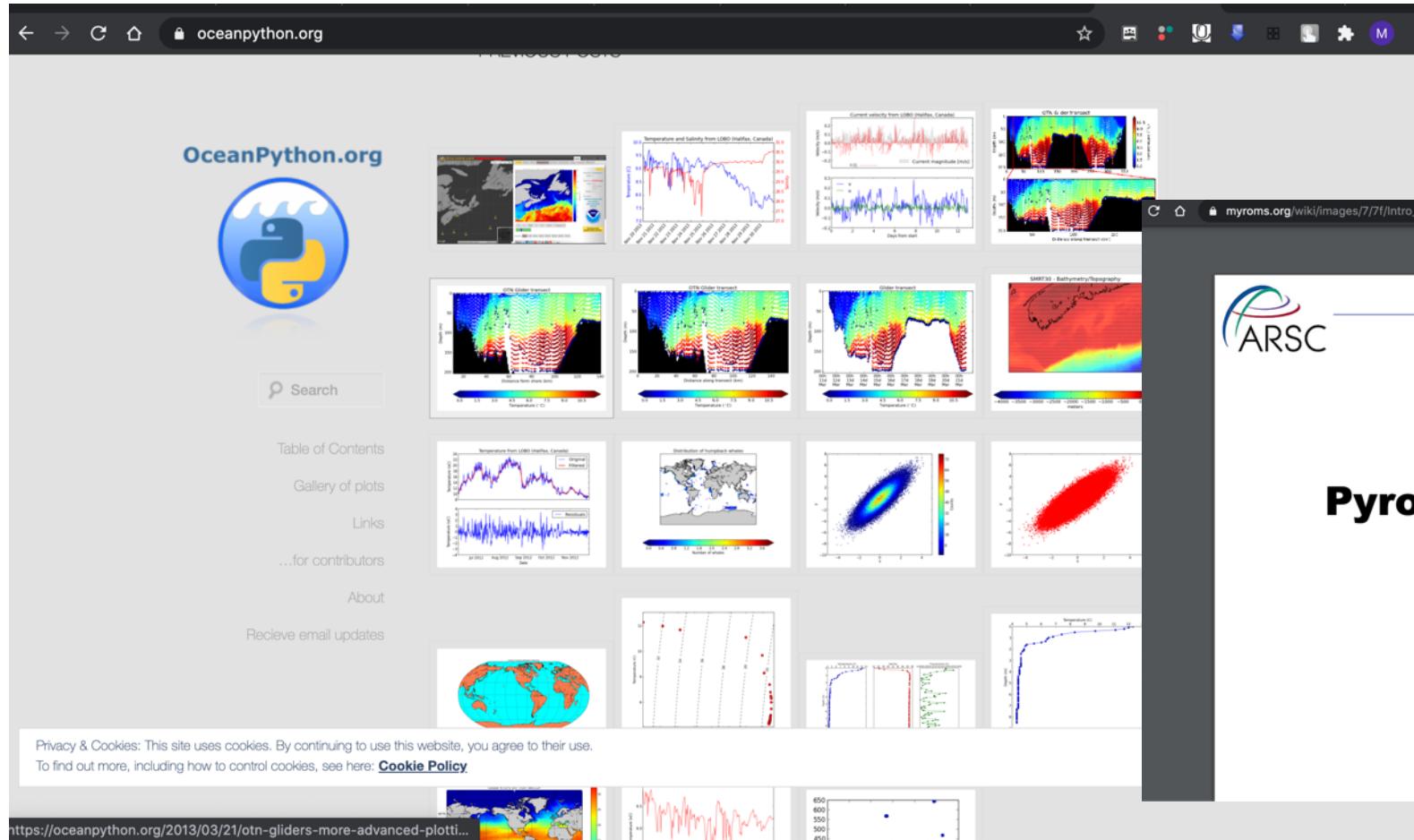
find and manage data: 24

GIS

The screenshot shows the ArcGIS Desktop interface. At the top, there are links for ArcGIS Desktop, ArcGIS Pro, and ArcMap. Below that is a navigation bar with tabs: Home, Get Started, Map, Analyze (which is highlighted in blue), Manage Data, Tools, and Extensions. Underneath the navigation bar, the main content area has a breadcrumb trail: Analyze > Python. To the left, there is a sidebar with a list of topics, and at the bottom, there are two buttons: "What is Python?" and "Essential Python vocabulary".

The screenshot shows a course page from Esri Academy. At the top, there is a navigation bar with links for Products, Solutions, Support & Services, News, and About. Below that is another navigation bar with links for Esri Academy, About, Catalog (which is highlighted in blue), and other categories like Course Catalog, Courses by Schedule, Courses by Location, New and Retired Training, and Learning Plans. A banner at the top right says "Get the latest information on training and certification during coronavirus disease". The main content area features a large image with the text "Web Course" and "Python for Everyone". Below the image, there is a "Sign In" button and social sharing icons. Text on the page includes "Duration: 3 Hours, 30 Minutes" and "Cost: Free". At the bottom, there is a section titled "About this Course".

Oceanography



myroms.org/wiki/images/7/71/intro_pyroms.pdf

ARSC

Pyroms – Python for ROMS

Kate Hedstrom
Frederic Castruccio
Bob Torgerson

Marine Biology

C 🔍 forbio.uio.no/events/courses/2019/Python%20for%20biologists%20-%20Advanced.html

The screenshot shows the ForBio website with the following details:

- ForBio Logo:** A green stylized tree icon followed by the text "ForBio".
- Text:** "Forskerskole i Biosystematikk" and "Research School in Biosystematics".
- Navigation Bar:** Home, Membership, Courses and events (highlighted), Course grants, Newsletter, About.
- Section:** Courses and events (highlighted).
- Section:** ForBio courses (highlighted).
- List:** 2021, 2020, 2019, 2018, 2017, 2016.
- Section:** Python for biologists - Advanced.
- Text:** Time and place: Dec. 4, 2019 12:00 PM–7:00 PM, Gothenburg Global Bi... Carl Skottsbergs gata 22B, Gothenburg.
- Logos:** NEIC logo (Nordic E-infrastructures Collaboration), LIFEWA logo, and ForBio logo.

Search

pythonforbiologists.com/upcoming-workshops/introduction-to-python-for-biologists-online-course-13th-24th-jul... ☆ 🔍

BIOLOGICAL DATA EXPLORATION
WITH
PYTHON, PANDAS AND SEABORN

Clean, filter, reshape and visualize complex biological datasets using the scientific Python stack

**Now published:
*Biological Data Exploration***

A complete guide to cleaning, manipulating and visualizing complex biological datasets with Python.

CLICK HERE FOR MORE DETAILS

The book cover has a blue and red geometric pattern background. The title "BIOLOGICAL DATA EXPLORATION" is at the top in white, with "WITH PYTHON, PANDAS AND SEABORN" below it. The subtitle "Clean, filter, reshape and visualize complex biological datasets using the scientific Python stack" is at the bottom. The authors' names are listed on the back cover.

Underwater Acoustics

arpyp.readthedocs.io/en/latest/uwapm.html

arpyp
latest

Search docs

- Signal processing
- Communications
- Beamforming and array processing
- Stable distributions
- Geographical coordinates
- Underwater acoustics
- Underwater acoustic propagation modeling
- Plotting utilities
- Common utilities
- Digital Towed Array
- ROMANIS
- High frequency data acquisition system
- Unet modem & network stack



Read the Docs v: latest

Docs » Underwater acoustic propagation modeling

Edit

Underwater acoustic propagation modeling

Underwater acoustic propagation modeling toolbox.

This toolbox currently uses the Bellhop acoustic propagation model. For this model to work [acoustic toolbox](#) must be installed on your computer and *bellhop.exe* should be in your PA'

Sample Jupyter notebook

For usage examples of this to see [Bellhop notebook](#).

`arpyp.uwapm.arrivals_to_impulse_response(arrivals, fs, abs_time=False)`

Convert arrival times and coefficients to an impulse response.

Parameters:

- `arrivals` – arrivals times (s) and coefficients
- `fs` – sampling rate (Hz)
- `abs_time` – absolute time (True) or relative time (False)

Returns: impulse response

If `abs_time` is set to True, the impulse response is placed such that the zero time corresponds to the first arrival time.

journals.plos.org/plosone/article?id=10.1371/journal.pone.0144610

pyAudioAnalysis: An Open-Source Python Library for Audio Signal Analysis

Theodoros Giannakopoulos

Published: December 11, 2015 • <https://doi.org/10.1371/journal.pone.0144610>

Article	Authors	Metrics	Comments	Media Coverage
Abstract	Introduction	Metrics	Comments	Media Coverage

Abstract

Audio information plays a rather important role in the increasing digital content that is available today, resulting in a need for methodologies that automatically analyze such content: audio event recognition for home automation and surveillance systems, speech recognition, music information retrieval, multimodal analysis (e.g. audio-visual analysis of online videos for content-based recommendation), etc. This paper presents pyAudioAnalysis, an open-source Python library that provides a wide range of audio analysis procedures including: feature extraction, classification of audio signals, supervised and unsupervised segmentation and content visualization. pyAudioAnalysis is licensed under the Apache License and is available at GitHub (<https://github.com/yiannak/pyAudioAnalysis>). Here we present the theoretical background behind the wide range of the implemented methodologies, along with evaluation

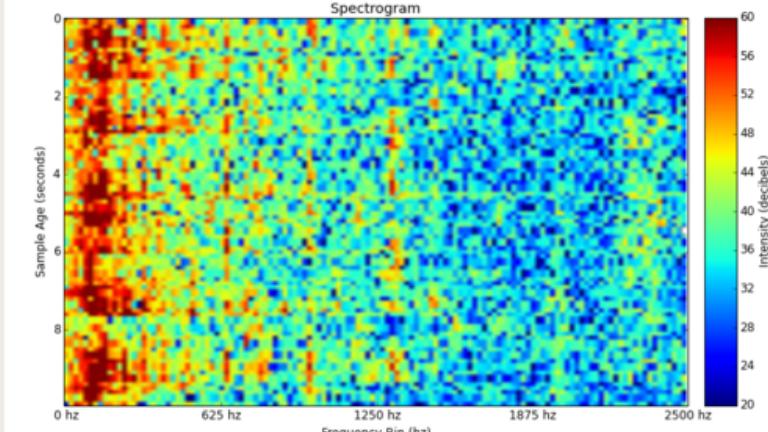
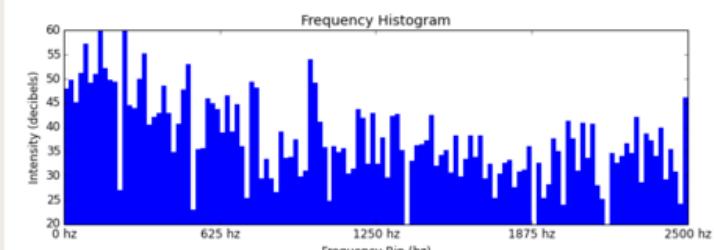
Spectrogram v1.0

by Tony DiCola

Device
Serial Port: ttyACM0
Open

Device Parameters
FFT Size: 256
Sample Rate: 5000 Hz
Modify

Graphs
Intensity Min: 20 dB
Intensity Max: 60 dB



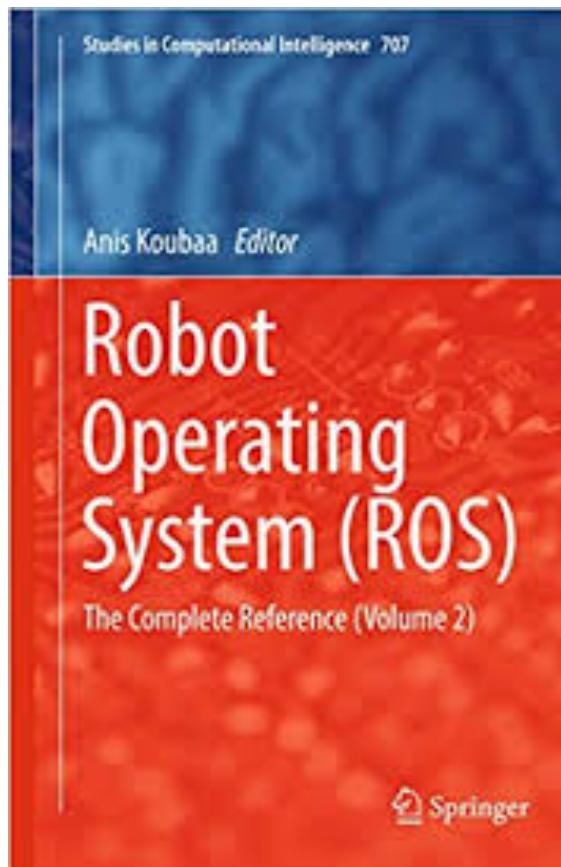
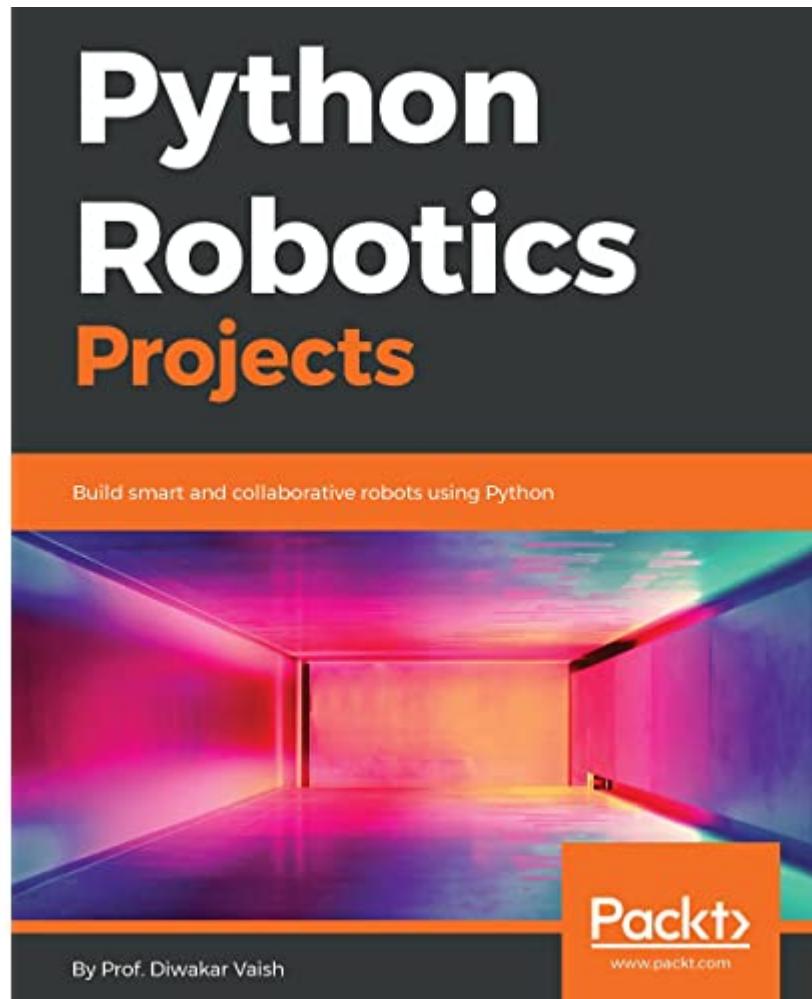
211,581 View 18 Share

Download PDF
Print Share

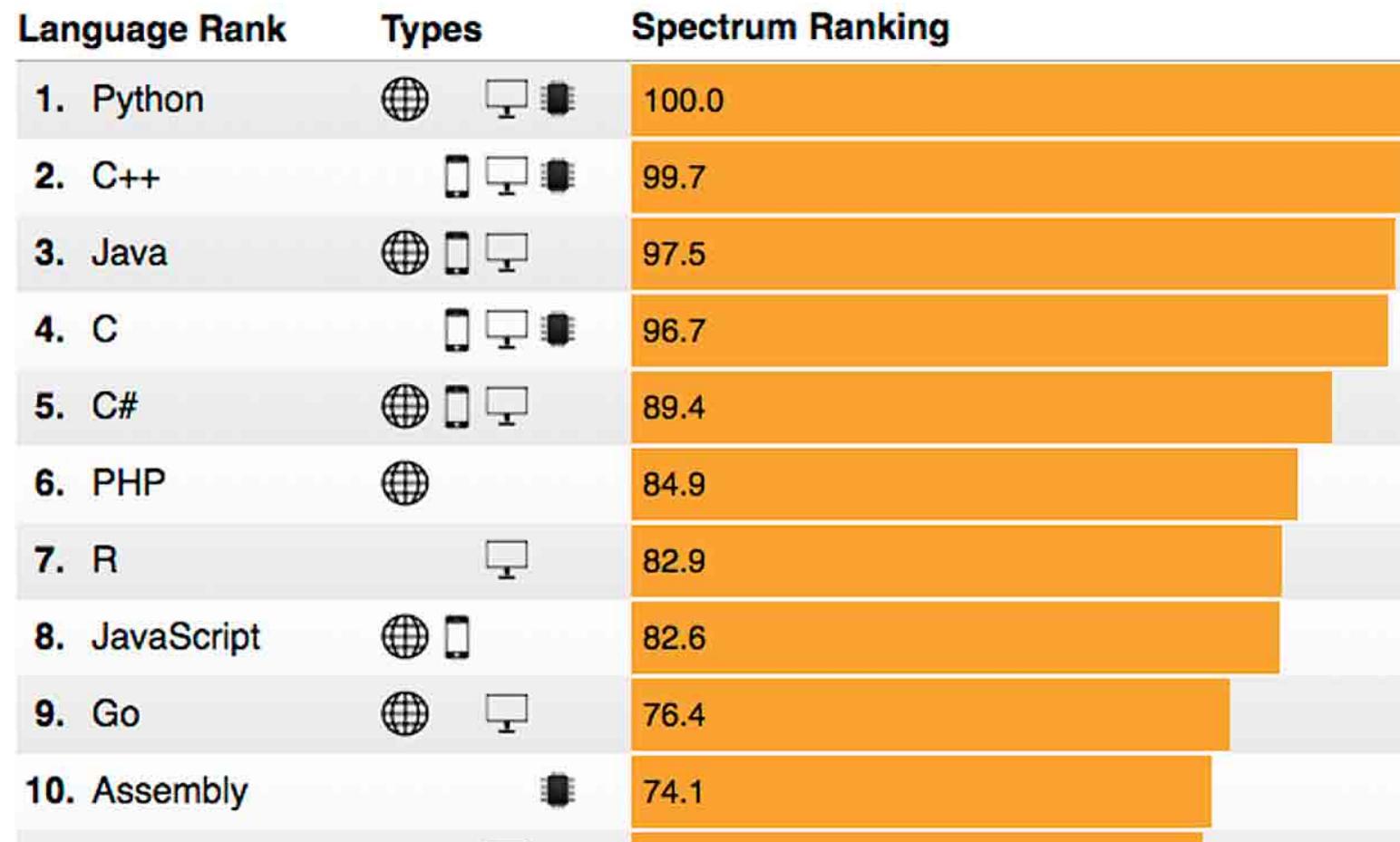
Check for updates
ADVERTISEMENT

PLOS ONE
NEW COLLECTION Plastics in the

Instrumentasi dan Robotics



Python anywhere

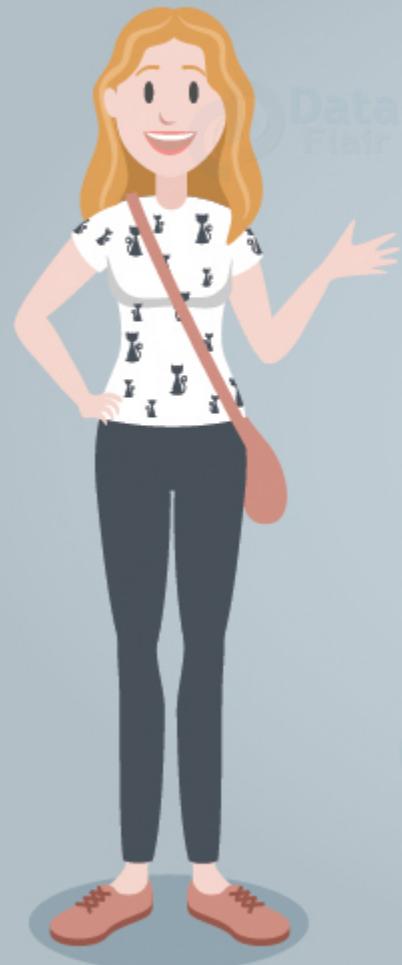


Contoh Aplikasi yang menggunakan Python:

Google
Youtube
Google App Engine
Pixar
BitTorrent



7 Reasons Why You Should Learn Python



01

Perfect For Rookies

02

Community

03

Career Opportunities

04

Python in Web Development

05

Python in Artificial Intelligence and Machine Learning

06

Raspberry Pi

07

Startups and Corporates- Python for Both

Textbook Python di Kuliah Ini



Cara penulisan variabel

1. Karakter pertama harus berupa huruf atau garis bawah/underscore (_)
2. Karakter selanjutnya dapat berupa huruf, garis bawah/underscore(_) atau angka
3. Karakter pada nama variabel bersifat sensitif (case-sensitif).
4. Menuliskan variabel dengan suatu nilai dengan cara menambahkan tanda sama dengan (=) diikuti dengan nilai yang ingin dimasukan.

Macam-macam Tipe Data

Tipe Data	Keterangan	Contoh
Boolean	Menyatakan benar(True) yang bernilai 1, atau salah(False) yang bernilai 0 (1 bit)	True atau False
String	Menyatakan karakter/kalimat bisa berupa huruf angka, dll (diapit tanda " atau ')	"Ayo belajar Python"
Integer	Menyatakan bilangan bulat (16 bit)	10
Float	Menyatakan bilangan yang mempunyai koma (32 bit)	3.14
long	Perhitungan diluar range nilai integer ataupun float (32 bit)	1/10000000000000000000000000000000 000000
Complex	Menyatakan pasangan angka real dan imajiner	1 + 5j
List	Data untaian yang menyimpan berbagai tipe data dan isinya bisa diubah-ubah	['xyz', 786, 2.23]
Tuple	Data untaian yang menyimpan berbagai tipe data tapi isinya tidak bisa diubah	('xyz', 768, 2.23)
Dictionary	Data untaian yang menyimpan berbagai tipe data berupa pasangan penunjuk dan nilai	{'nama': 'adi','id':2}

Arithmetric Operators

Operator	Keterangan
+	Menambahkan dua obyek
-	Mengurangi obyek dengan obyek yang lain
*	Perkalian
**	Pangkat
/	Pembagian
//	Pembagian bulat ke bawah
%	Sisa hasil bagi (modulus)

Comparison Operators

Operator	Keterangan
<code>==</code>	Jika masing-masing memiliki nilai yang sama, maka kondisi bernilai benar atau True
<code>!=</code>	Akan menghasilkan nilai kebalikan dari kondisi sebenarnya
<code><></code>	Akan menghasilkan nilai kebalikan dari kondisi sebenarnya
<code>></code>	Jika nilai kiri lebih besar dari nilai kanan, maka kondisi menjadi benar
<code><</code>	Jika nilai kiri lebih kecil dari nilai kanan, maka kondisi menjadi benar
<code>>=</code>	Jika nilai kiri lebih besar dari nilai kanan, atau sama, maka kondisi menjadi benar
<code><=</code>	Jika nilai kiri lebih kecil dari nilai kanan, atau sama, maka kondisi menjadi benar

Assignment Operators

Operator	Keterangan
=	Memberikan nilai di kanan ke dalam variabel yang berada di sebelah kiri.
+=	Memberikan nilai variabel dengan nilai variabel itu sendiri ditambah dengan nilai di sebelah kanan.
-=	Memberikan nilai variabel dengan nilai variabel itu sendiri dikurangi dengan nilai di sebelah kanan.
*=	Memberikan nilai variabel dengan nilai variabel itu sendiri dikali dengan nilai di sebelah kanan.
/=	Memberikan nilai variabel dengan nilai variabel itu sendiri dibagi dengan nilai di sebelah kanan.
%=	Memberikan nilai variabel dengan nilai variabel itu sendiri dibagi dengan nilai di sebelah kanan. Yang diambil nantinya adalah sisa baginya.
**=	Memberikan nilai variabel dengan nilai variabel itu sendiri dipangkatkan dengan nilai di sebelah kanan.
//=	Membagi bulat sebelah kiri operator dengan sebelah kanan operator kemudian hasilnya diisikan ke operan sebelah kiri.

Bitwise Operators

Operator	Keterangan
&	Operator biner AND, memeriksa apakah nilai di sebelah kiri dan nilai sebelah kanan mempunyai angka biner 1 di setiap bit. Jika keduanya bernilai 1 maka bit hasil operasi akan bernilai 1.
	Operator biner OR, memeriksa apakah nilai di sebelah kiri dan nilai sebelah kanan mempunyai angka biner 1 di setiap bit. Jika salah satunya bernilai 1 maka bit hasil operasi akan bernilai 1.
^	Operator biner XOR, memeriksa apakah nilai di sebelah kiri dan nilai sebelah kanan mempunyai angka biner 1 di setiap bit. Jika keduanya bernilai 1 maka bit hasil operasi akan bernilai 0.
~	Operasi membalikkan angka bitwise dari dari x, menghasilkan $-x-1$
<<	Memberikan nilai variabel dengan nilai variabel itu sendiri dibagi dengan nilai di sebelah kanan. Yang diambil nantinya adalah sisa baginya.
>>	Operator penggeser biner ke kiri, deret bit akan digeser ke kiri sebanyak n kali.

Logical Operators

Operator	Keterangan
and	Jika kedua operan bernilai True, maka kondisi akan bernilai True. Selain kondisi tadi maka akan bernilai False.
or	Jika salah satu atau kedua operan bernilai True maka kondisi akan bernilai True. Jika keduanya False maka kondisi akan bernilai False.
not	Membalikkan nilai kebenaran pada operan misal jika asalnya True akan menjadi False dan begitupun sebaliknya.

Membership Operators

Operator	Keterangan
in	Memeriksa apakah nilai yang dicari berada pada list atau struktur data python lainnya. Jika nilai tersebut ada maka kondisi akan bernilai True.
not in	Memeriksa apakah nilai yang dicari tidak ada pada list atau struktur data python lainnya. Jika nilai tersebut tidak ada maka kondisi akan bernilai True.

Identity Operators

Operator	Keterangan
is	Memeriksa apakah nilai di sebelah kiri memiliki identitas memori yang sama dengan nilai di sebelah kanan operan. Jika sama maka kondisi bernilai True.
is not	Memeriksa apakah nilai yang dicari tidak ada pada list atau struktur data python lainnya. Jika nilai tersebut tidak ada maka kondisi akan bernilai True.

List

- List adalah struktur data yang menyimpan koleksi data, anda dapat menyimpan sequence / rangkaian nilai menggunakan list.
- Data untaian yang menyimpan berbagai tipe data dan isinya bisa **diubah-ubah**.
- **Data = [nilai1, nilai2, nilai3]**

Metode-metode yang terdapat di dalam list

- **contoh_list.append('item 1')**, akan menambahkan string '**item 1**' kedalam list **contoh_list**.
- **contoh_list.remove('item 1')**, akan menghapus string '**item 1**' kedalam list **contoh_list**. Atau dapat menggunakan **del contoh_list[index]** untuk menghapus index yang diinginkan (0, 1,2,.....)
- **contoh_list.sort()**, akan mengurutkan item yang ada di dalam **contoh_list**.

Tuple

- Tuple mirip dengan list namun tuple bersifat **immutable** (**tidak bisa diubah setelah didefinisikan**).
- **Data = ('item1','item2','item3')**

Dictionary

Dictionary seperti buku alamat, dengan buku alamat anda bisa mencari alamat atau detail kontak hanya menggunakan nama orang yang anda cari.

Data = {key1: value1, key2: value2}.

Bentuk if – else – elif

Bentuk umum perintah if – else – elif

if (kondisi 1) :

statemen 1

elif (kondisi 2) :

statemen 2

else:

statemen 3

try...except...finally

try:

statemen 1

except :

statemen 2

finally:

statemen 3

Perintah for

Bentuk umum perintah *for* :

for (variabel) in (objek) :
statemen

while

Perintah while pada Python merupakan perintah yang paling umum digunakan untuk proses iterasi. Konsep sederhana dari perintah while adalah ia akan mengulang mengeksekusi statemen dalam blok while selama nilai kondisinya benar dan ia akan keluar atau tidak melakukan eksekusi blok statemen jika nilai kondisinya salah.

Bentuk umum statemen while :

while (kondisi_memenuhi) :
statemen

NEXT:

FUNCTION AND CLASSES