USER MANUEL

Team Name:

- Omedi

Project Name:

- Assistant Developed Using Natural Language Processing Techniques

Team Members:

- Dilan Kaya
- Osman Melih Tolunay
- Umut Şentürk
- Mustafa Körükmez

Main Contact for the Instructor:

- Mustafa Körükmez

Instructor:

- Ahmet Çağdaş Seçkin

Table of Contents

I.REQUIRED APPLICATIONS FOR INSTALLATION

i.For Admin ;

- ✓ Python in version 3 and above
- ✓ New generation ide
- ✓ GitHub Desktop

ii.For User;

- ✓ A compiler to run on a Conda base
- ✓ Python in version 3 and above
- ✓ Anaconda

II.TECHNICAL SPECIFICATIONS OF THE ENVIRONMENT TO BE INSTALLED

✓ For mobile devices and tablets it must be above android 4.0. For other mobile operating systems, it must be 2008-2010 or higher. Apart from that, it can be written for any mobile operating system and can be used with Termux, one of the Linux envoriment systems, which is very easy to activate on android devices, especially in the content of the device.

✓ For desktop and laptop, the device must be capable of removing at least Python3.0 version. There should also be an audio input that can exchange audio.

III.SYSTEM FEATURES

✓ To help our daily life by performing main operations with voice commands of the device when we cannot reach our device or have sufficient information to use the device.

✓ Can search in the browser with voice command and get the result of the search by voice

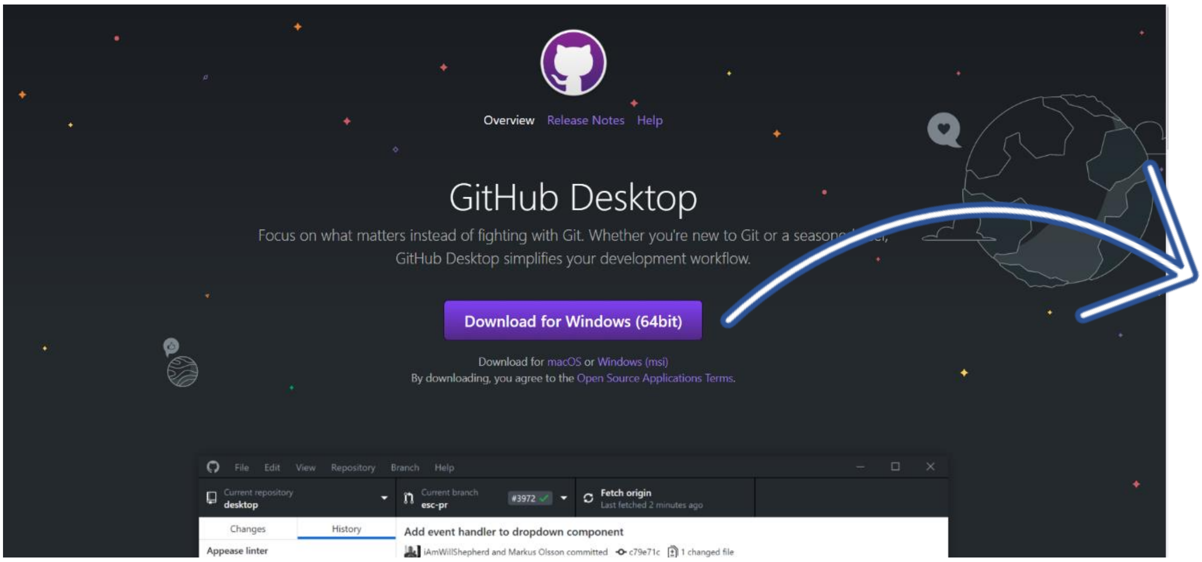
✓ You can open integrated applications with a voice command and note with a voice command.

IV.MAIN SCENARIO

✓ The user to operate in a virtual environment by giving voice commands.

V. APPLICATIONS THAT NEED TO BE INSTALLED FOR THE ADMIN

i.GitHub Desktop Setup



GitHub Desktop

Focus on what matters instead of fighting with Git. Whether you're new to Git or a seasone... GitHub Desktop simplifies your development workflow.

Download for macOS or Windows (msi)

By downloading, you agree to the Open Source Applications Terms.

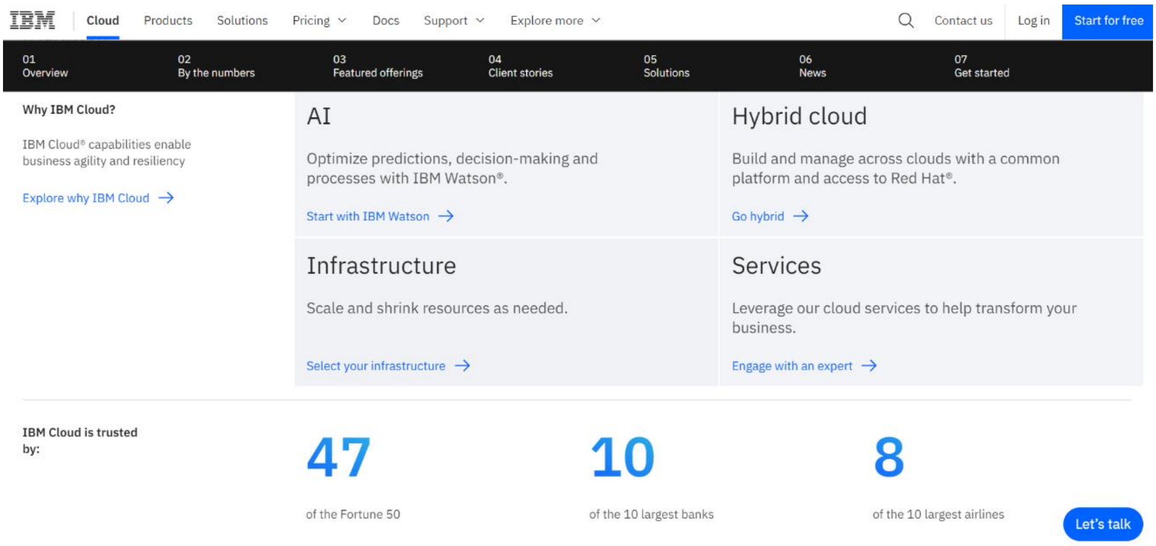
Appease linter

Add event handler to dropdown component

iAmWillShepherd and Markus Olsson committed c79e71c 1 changed file

CLICK FOR [DOWNLOAD]

✓ First, we download GitHub Desktop so that we can always control our open-source project and add new actions to it. Also, since the most important goal of the project is Linguistics, we are connecting our GitHub Desktop with IBM Cloud.



Why IBM Cloud?

IBM Cloud® capabilities enable business agility and resiliency

Explore why IBM Cloud →

AI

Optimize predictions, decision-making and processes with IBM Watson®.

Start with IBM Watson →

Hybrid cloud

Build and manage across clouds with a common platform and access to Red Hat®.

Go hybrid →

Infrastructure

Scale and shrink resources as needed.

Select your infrastructure →

Services

Leverage our cloud services to help transform your business.

Engage with an expert →

IBM Cloud is trusted by:

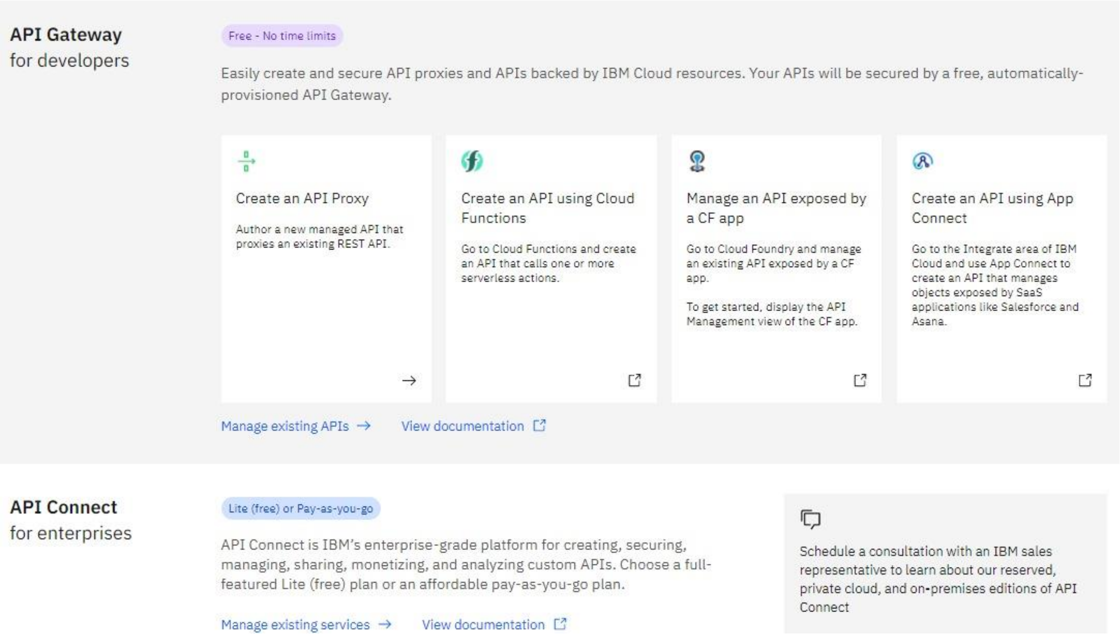
47 of the Fortune 50

10 of the 10 largest banks

8 of the 10 largest airlines

✓ Through to IBM Cloud, the language model can be rewritten for the region and the language used by the person who cloned the project from GitHub and wants to use it in his vehicle. With the written language model, we can easily perform the morphologically based linguistics operation that makes sense of what the user says.

ii.How to Include IBM Cloud in the Project?



API Gateway
for developers

Free - No time limits

Easily create and secure API proxies and APIs backed by IBM Cloud resources. Your APIs will be secured by a free, automatically-provisioned API Gateway.

Create an API Proxy

Author a new managed API that proxies an existing REST API.

Create an API using Cloud Functions

Go to Cloud Functions and create an API that calls one or more serverless actions.

Manage an API exposed by a CF app

Go to Cloud Foundry and manage an existing API exposed by a CF app.

To get started, display the API Management view of the CF app.

Create an API using App Connect

Go to the Integrate area of IBM Cloud and use App Connect to create an API that manages objects exposed by SaaS applications like Salesforce and Asana.

Manage existing APIs →   View documentation

API Connect
for enterprises

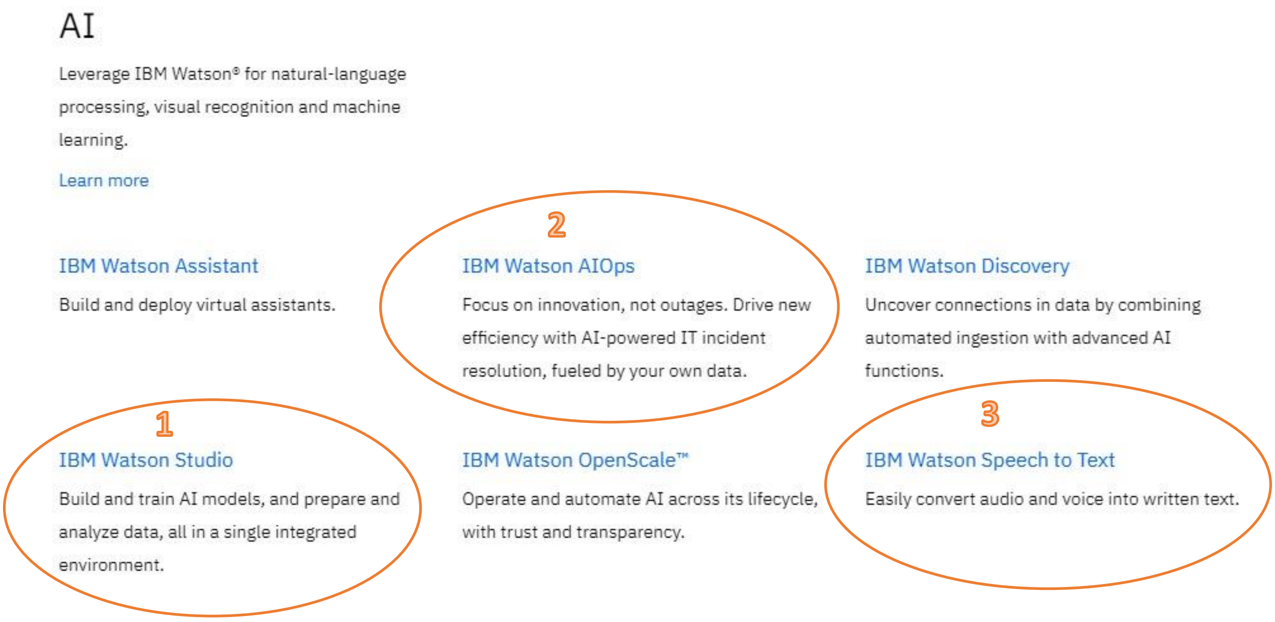
Lite (free) or Pay-as-you-go

API Connect is IBM's enterprise-grade platform for creating, securing, managing, sharing, monetizing, and analyzing custom APIs. Choose a full-featured Lite (free) plan or an affordable pay-as-you-go plan.

Manage existing services →   View documentation

Schedule a consultation with an IBM sales representative to learn about our reserved, private cloud, and on-premises editions of API Connect

- ✓ After subscribing to IBM Cloud, we choose APIs related to our project.



AI

Leverage IBM Watson® for natural-language processing, visual recognition and machine learning.

Learn more

IBM Watson Assistant

Build and deploy virtual assistants.

2

IBM Watson AIOps

Focus on innovation, not outages. Drive new efficiency with AI-powered IT incident resolution, fueled by your own data.

IBM Watson Discovery

Uncover connections in data by combining automated ingestion with advanced AI functions.

1

IBM Watson Studio

Build and train AI models, and prepare and analyze data, all in a single integrated environment.

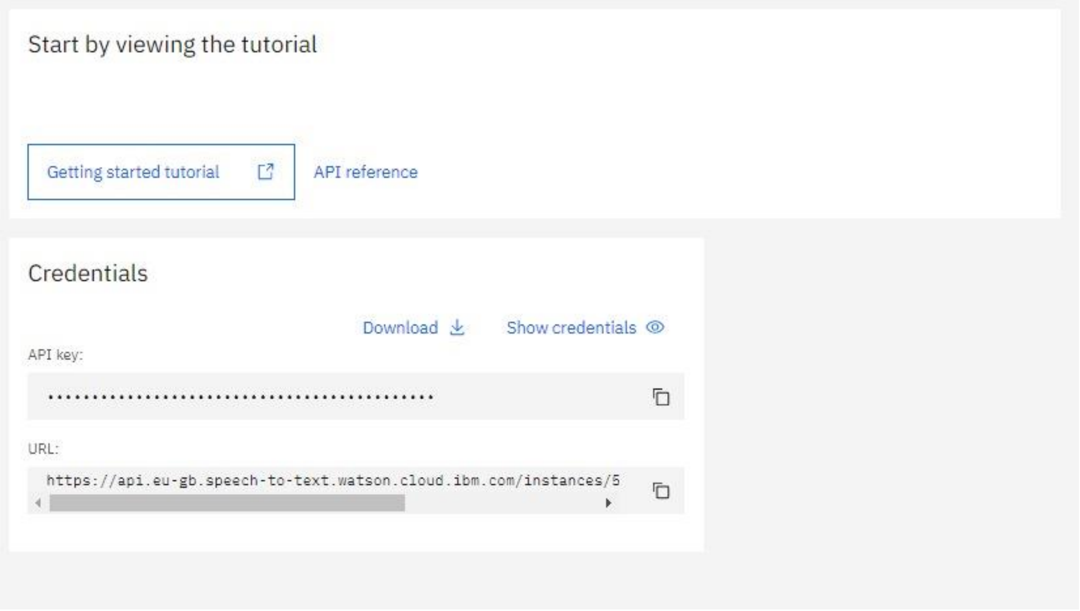
IBM Watson OpenScale™

Operate and automate AI across its lifecycle, with trust and transparency.

3

IBM Watson Speech to Text

Easily convert audio and voice into written text.



Start by viewing the tutorial

Credentials

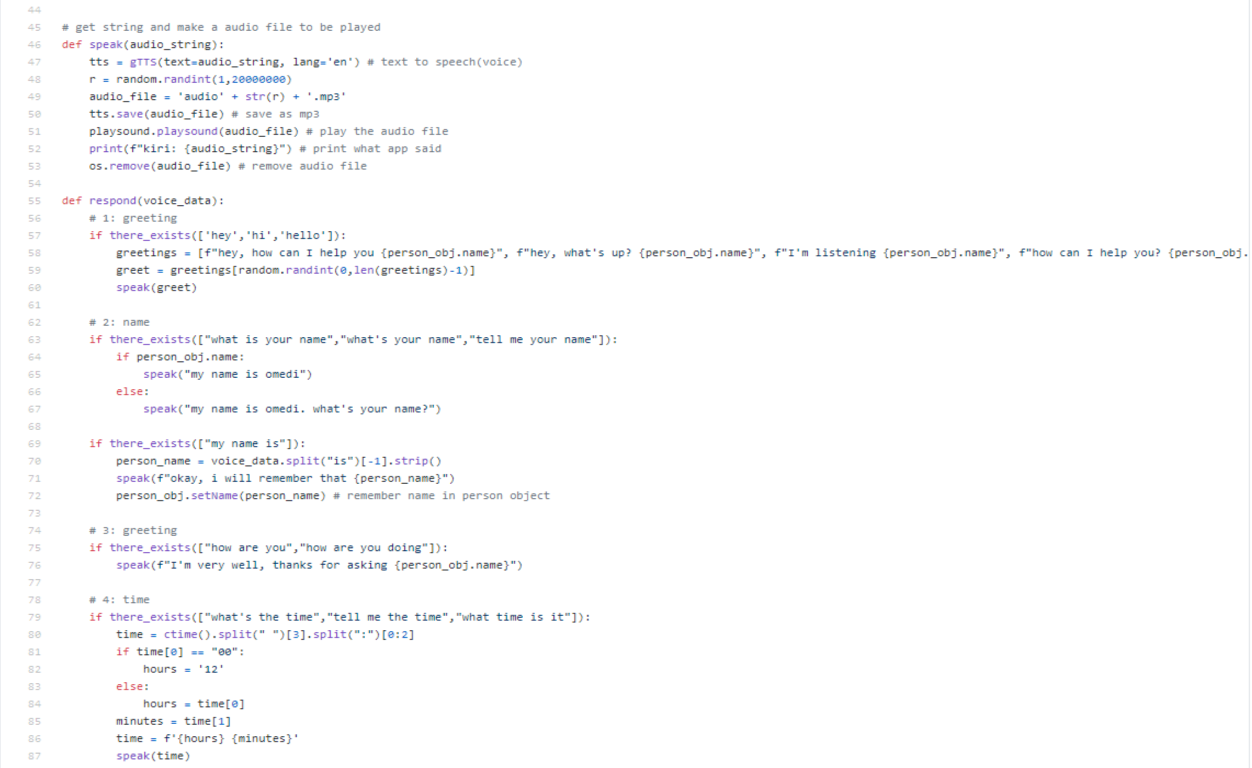
API key:

••

URL:

```
https://api.eu-gb.speech-to-text.watson.cloud.ibm.com/instances/5
```

✓ In order to include the APIs, we have selected above into our project, it is sufficient for each to include their own API identity and the custom API password that will be given to the user-admin, as shown in the figure.



```
# get string and make a audio file to be played
def speak(audio_string):
    tts = gTTS(text=audio_string, lang='en') # text to speech(voice)
    r = random.randint(1,20000000)
    audio_file = 'audio' + str(r) + '.mp3'
    tts.save(audio_file) # save as mp3
    playsound.playsound(audio_file) # play the audio file
    print(f"kiri: {audio_string}") # print what app said
    os.remove(audio_file) # remove audio file

def respond(voice_data):
    # 1: greeting
    if there_exists(['hey','hi','hello']):
        greetings = [f"hey, how can I help you {person_obj.name}", f"hey, what's up? {person_obj.name}", f"I'm listening {person_obj.name}", f"how can I help you? {person_obj.
        greet = greetings[random.randint(0,len(greetings)-1)]
        speak(greet)

    # 2: name
    if there_exists(["what is your name","what's your name","tell me your name"]):
        if person_obj.name:
            speak("my name is omedi")
        else:
            speak("my name is omedi. what's your name?")

    if there_exists(["my name is"]):
        person_name = voice_data.split("is")[-1].strip()
        speak(f"okay, i will remember that {person_name}")
        person_obj.setName(person_name) # remember name in person object

    # 3: greeting
    if there_exists(["how are you","how are you doing"]):
        speak(f"I'm very well, thanks for asking {person_obj.name}")

    # 4: time
    if there_exists(["what's the time","tell me the time","what time is it"]):
        time = ctime().split(" ")[3].split(":")[0:2]
        if time[0] == "00":
            hours = '12'
        else:
            hours = time[0]
        minutes = time[1]
        time = f'{hours} {minutes}'
        speak(time)
```

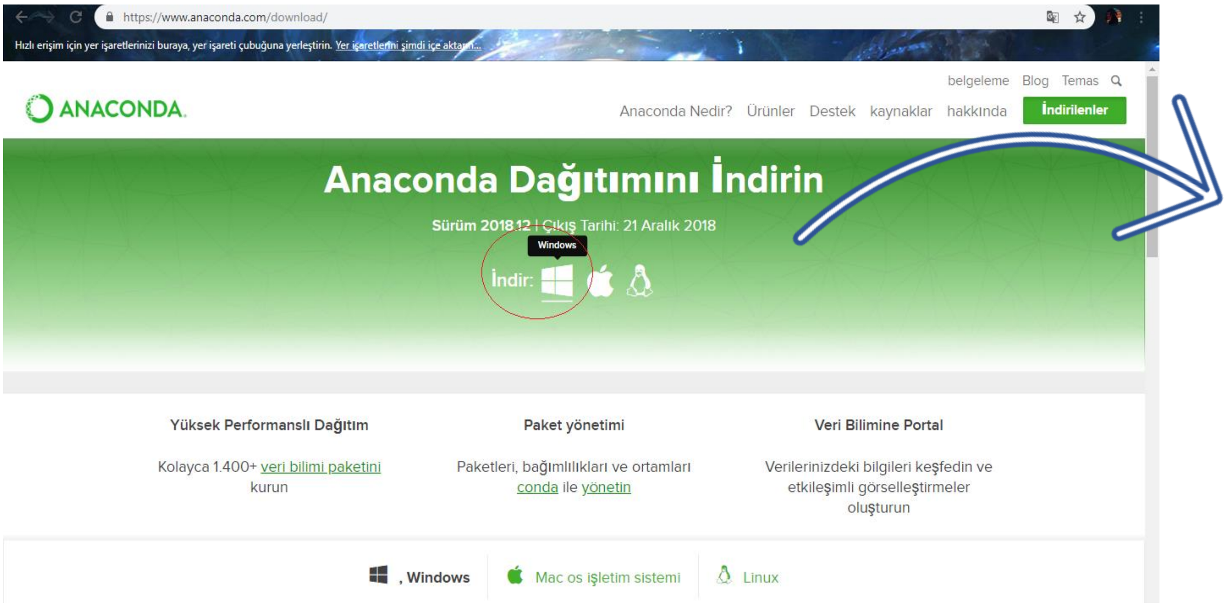
✓ As can be seen in the last figure, coding is done by using functions on NLP in the Python language.

VI. APPLICATIONS THAT NEED TO BE INSTALLED FOR THE USER

i. Anaconda Setup

✓ Click here to download the Anaconda integrated distribution and download the appropriate

✓ (Windows & MacOS & Linux) setup file for your computer.


https://www.anaconda.com/download/
Hızlı erişim için yer işaretlerinizi buraya, yer işareti çubuğuna yerleştirin. Yer işaretlerini şimdi içe aktarın...
belgeleme
Blog
Temas
ANACONDA.
Anaconda Nedir?
Ürünler
Destek
kaynaklar
hakkında
İndirilenler
Windows
İndir:
, Windows
Mac os işletim sistemi
Linux

Anaconda Dağıtımını İndirin

Sürüm 2018.12 | Çıkış Tarihi: 21 Aralık 2018

Yüksek Performanslı Dağıtım

Kolayca 1.400+ veri bilimi paketini kurun

Paket yönetimi

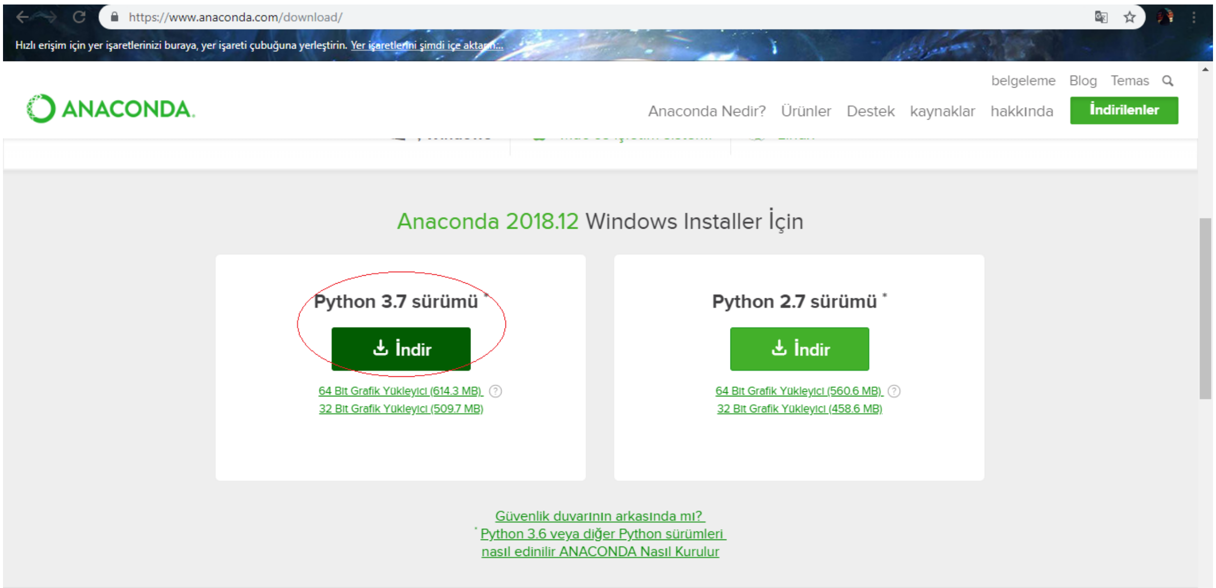
Paketleri, bağımlılıkları ve ortamları conda ile yönetin

Veri Bilimine Portal

Verilerinizdeki bilgileri keşfedin ve etkileşimli görselleştirmeler oluşturun

CLICK FOR DOWNLOAD

✓ Then, if you do not know how many bits your computer has, check the computer features, how many bits your computer has, then download the 64-bit or 32-bit one.


https://www.anaconda.com/download/
Hızlı erişim için yer işaretlerinizi buraya, yer işareti çubuğuna yerleştirin. Yer işaretlerini şimdi içe aktarın...
ANACONDA.
belgeleme
Blog
Temas
Anaconda Nedir?
Ürünler
Destek
kaynaklar
hakkında
İndirilenler
İndir
İndir

Anaconda 2018.12 Windows Installer İçin

Python 3.7 sürümü *

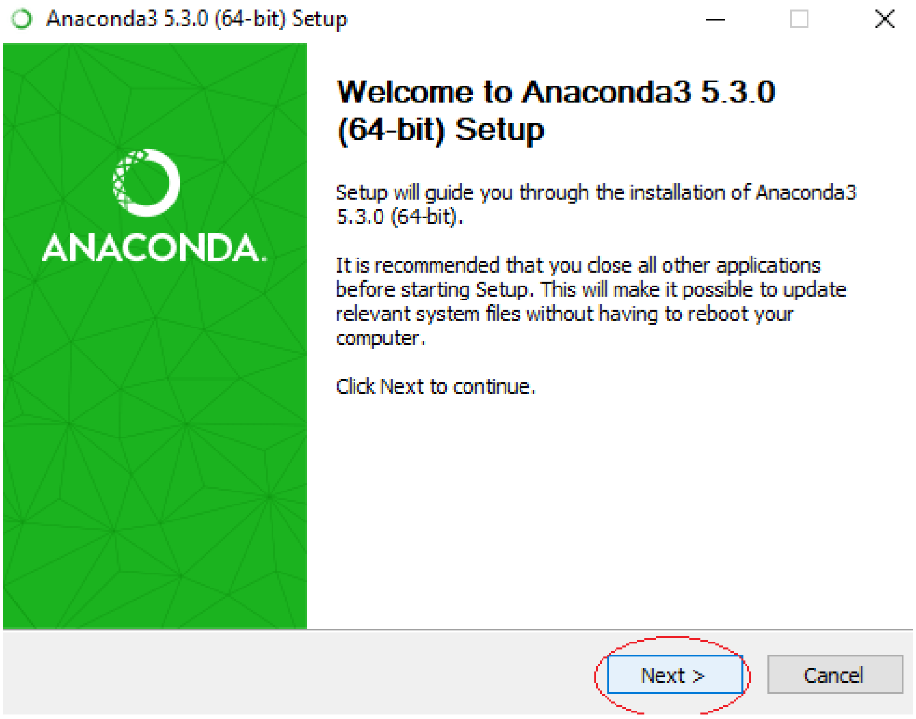
64 Bit Grafik Yükleyici (614.3 MB)
32 Bit Grafik Yükleyici (509.7 MB)

Python 2.7 sürümü *

64 Bit Grafik Yükleyici (560.6 MB)
32 Bit Grafik Yükleyici (458.6 MB)

Güvenlik duvarının arkasında mı?
* Python 3.6 veya diğer Python sürümleri nasıl edinilir ANACONDA Nasıl Kurulur

✓ After downloading Anaconda, click and open it and click "next" in the window that opens as you see it on the screen.



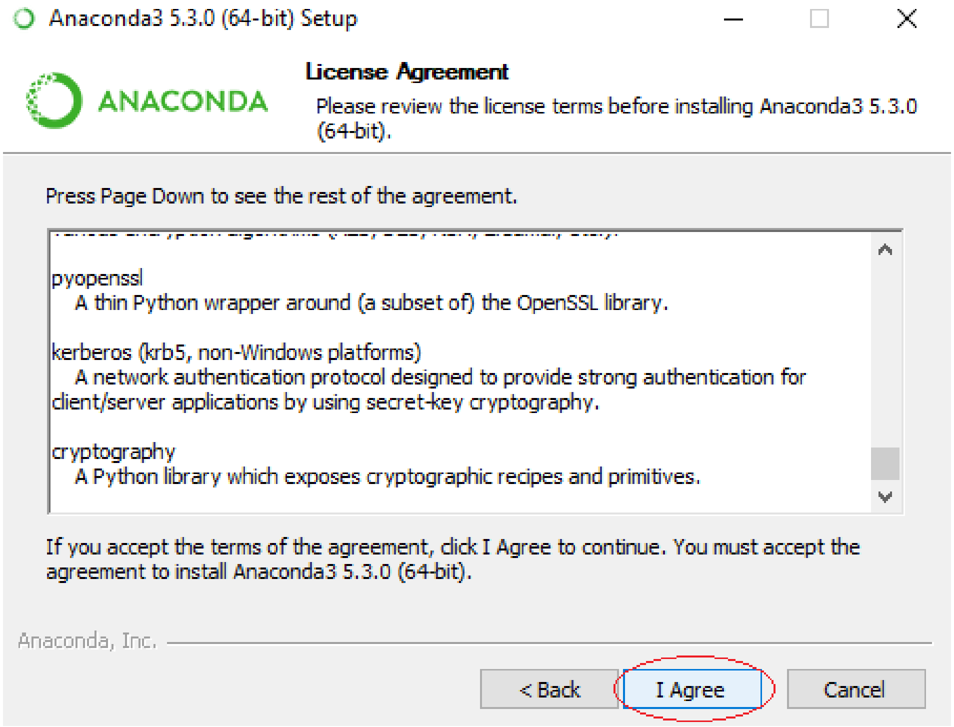
Welcome to Anaconda3 5.3.0 (64-bit) Setup

Setup will guide you through the installation of Anaconda3 5.3.0 (64-bit).

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

✓ The program will then offer you the terms of a license agreement. If you do not accept the program, the program will not be installed. We say, "I agree" and continue.



Anaconda3 5.3.0 (64-bit) Setup

License Agreement

Please review the license terms before installing Anaconda3 5.3.0 (64-bit).

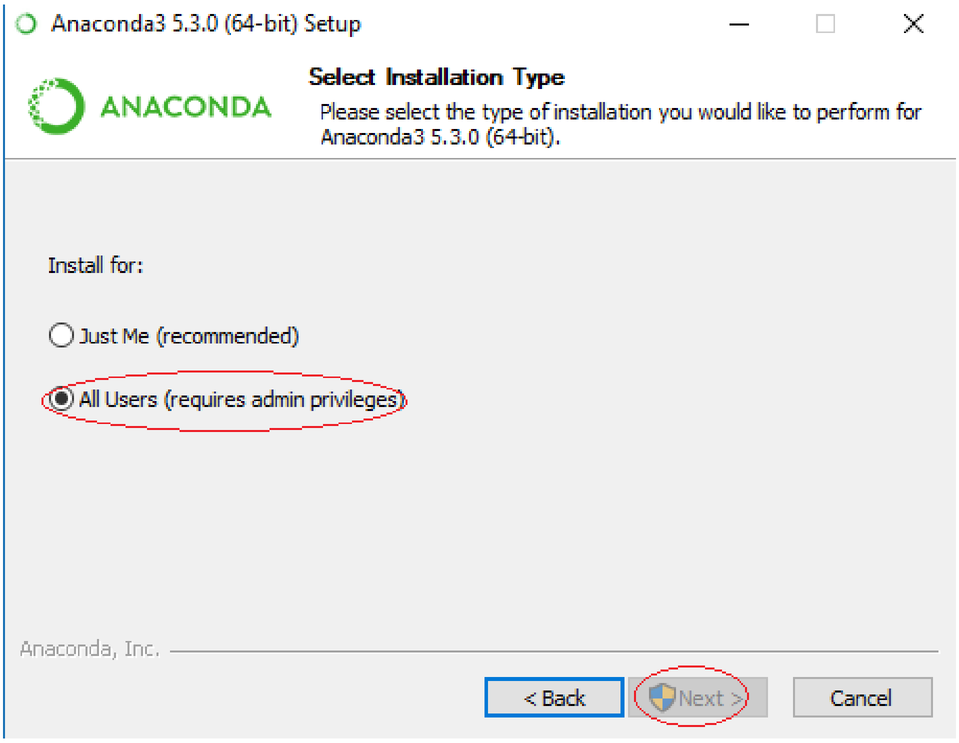
Press Page Down to see the rest of the agreement.

```
pyopenssl
    A thin Python wrapper around (a subset of) the OpenSSL library.

kerberos (krb5, non-Windows platforms)
    A network authentication protocol designed to provide strong authentication for
client/server applications by using secret-key cryptography.

cryptography
    A Python library which exposes cryptographic recipes and primitives.
```

If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install Anaconda3 5.3.0 (64-bit).

Anaconda, Inc.

✓ The program will ask you the type of installation, we say "All Users" here, we say "next" and run it as an administrator.



Anaconda3 5.3.0 (64-bit) Setup

Select Installation Type

Please select the type of installation you would like to perform for Anaconda3 5.3.0 (64-bit).

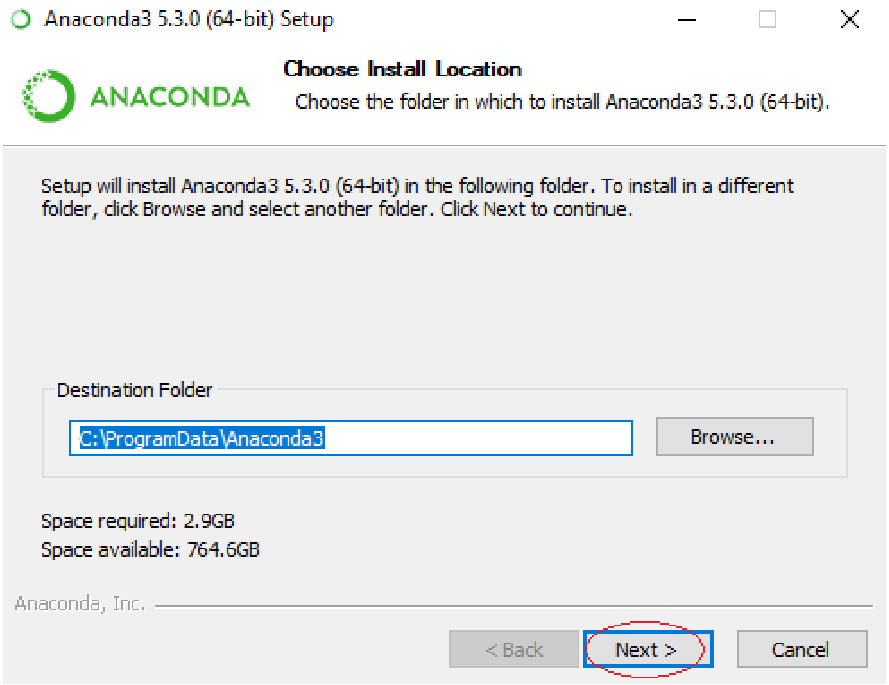
Install for:

○ Just Me (recommended)

● All Users (requires admin privileges)

Anaconda, Inc.

- ✓ Here the program shows the folder in which to install anaconda.



Anaconda3 5.3.0 (64-bit) Setup

Choose Install Location

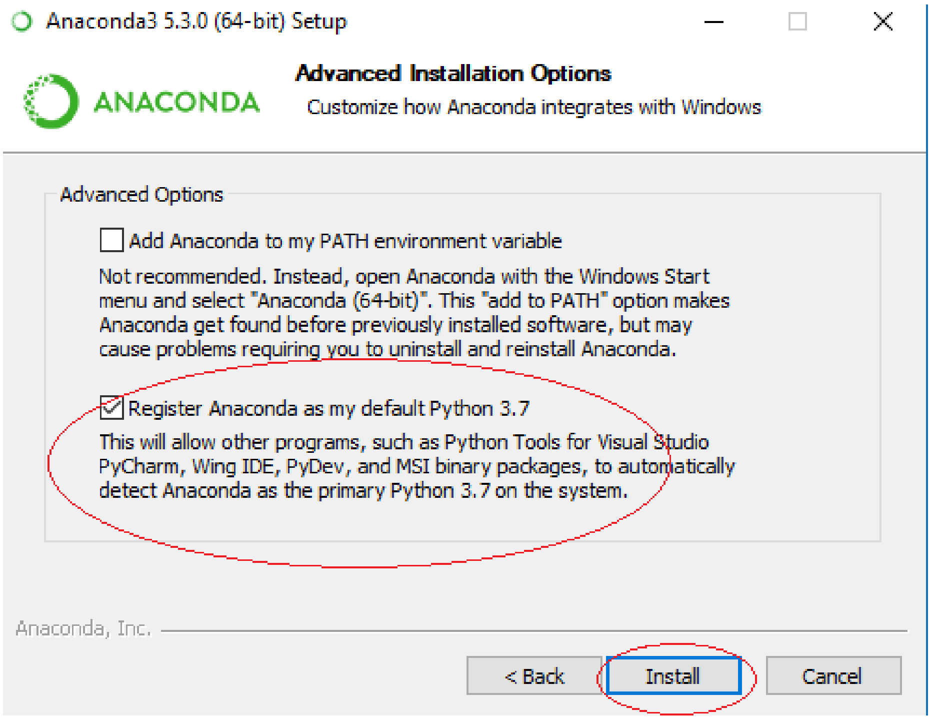
Choose the folder in which to install Anaconda3 5.3.0 (64-bit).

Setup will install Anaconda3 5.3.0 (64-bit) in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.

Destination Folder: C:\ProgramData\Anaconda3

Space required: 2.9GB
Space available: 764.6GB

- ✓ Here we download anaconda by clicking on "register as advanced Python version".



Anaconda3 5.3.0 (64-bit) Setup

Advanced Installation Options

Customize how Anaconda integrates with Windows

Advanced Options

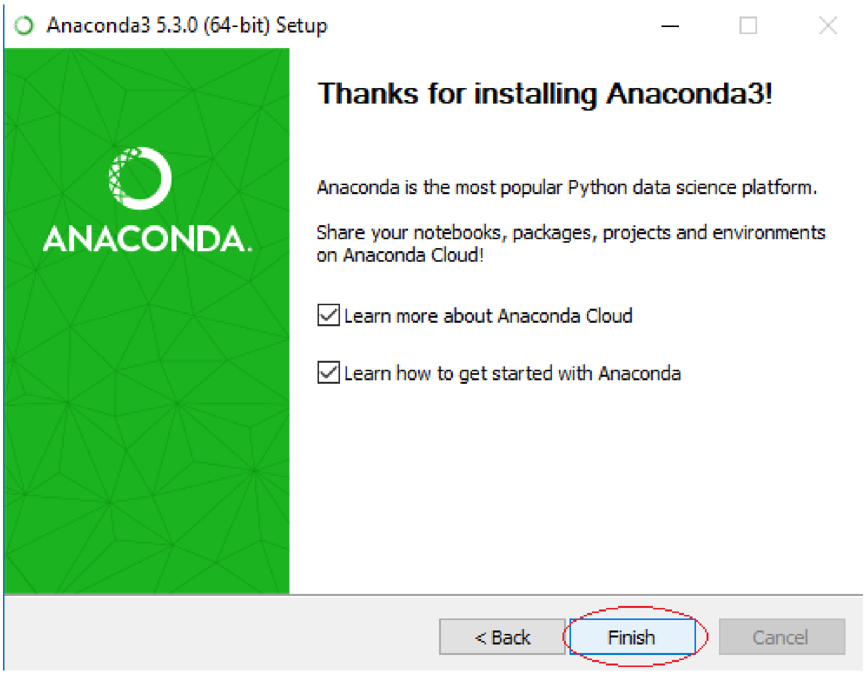
□ Add Anaconda to my PATH environment variable

Not recommended. Instead, open Anaconda with the Windows Start menu and select "Anaconda (64-bit)". This "add to PATH" option makes Anaconda get found before previously installed software, but may cause problems requiring you to uninstall and reinstall Anaconda.

☑ Register Anaconda as my default Python 3.7

This will allow other programs, such as Python Tools for Visual Studio PyCharm, Wing IDE, PyDev, and MSI binary packages, to automatically detect Anaconda as the primary Python 3.7 on the system.

- ✓ By saying "Finish" we complete the installation of the program.



Anaconda3 5.3.0 (64-bit) Setup

Thanks for installing Anaconda3!

Anaconda is the most popular Python data science platform.

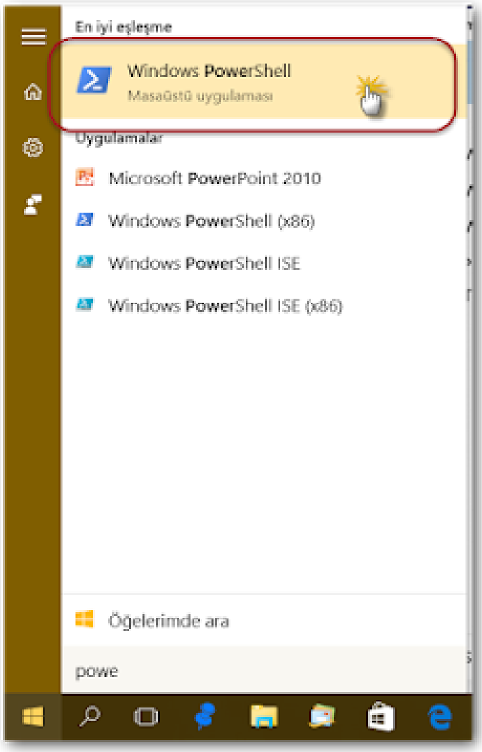
Share your notebooks, packages, projects and environments on Anaconda Cloud!

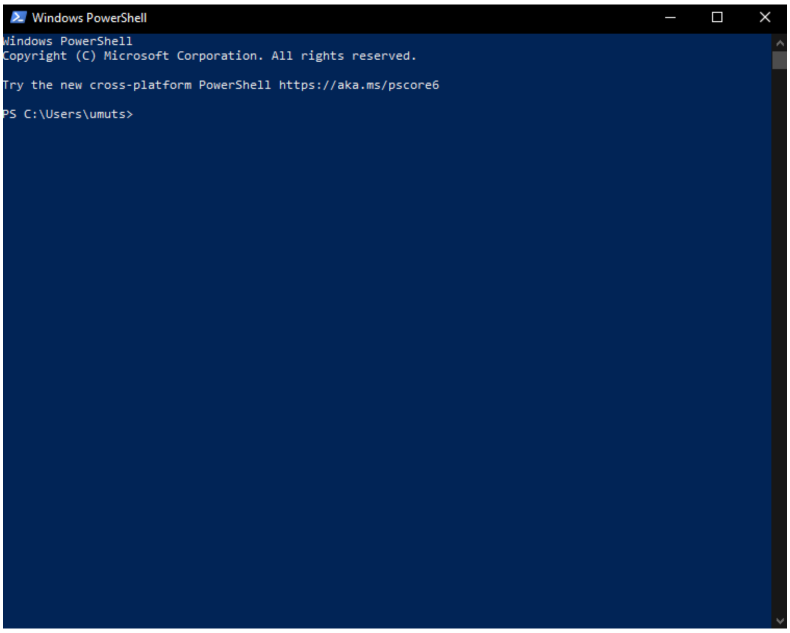
☑ Learn more about Anaconda Cloud

☑ Learn how to get started with Anaconda

ii. PowerShell

- ✓ Then, we open it by typing PowerShell in the search bar.



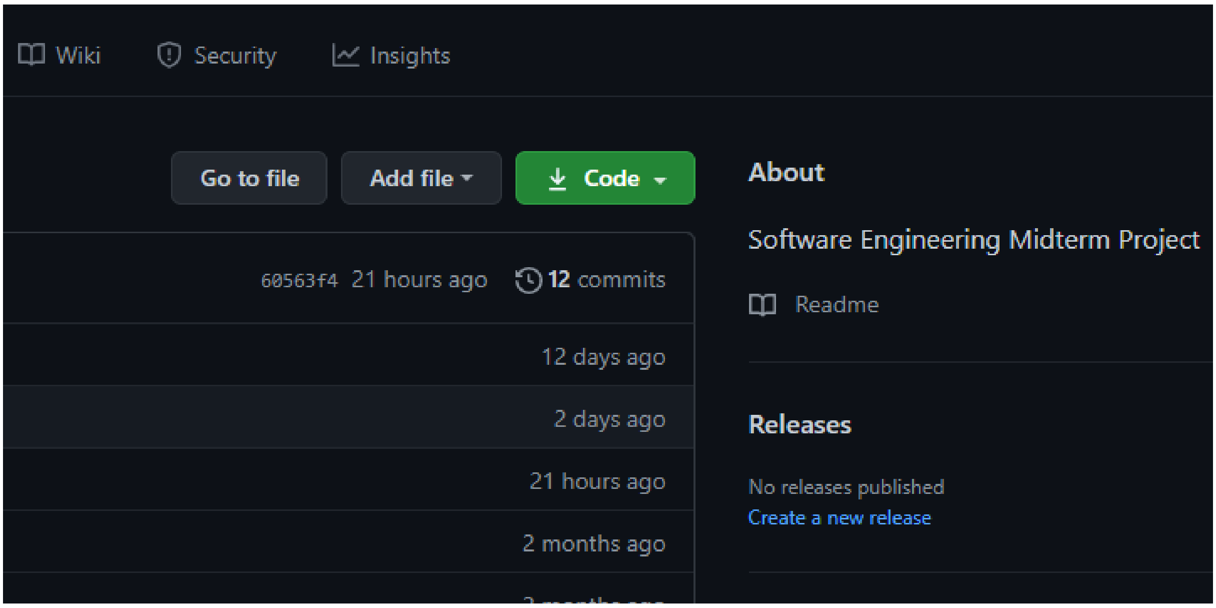


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\umuts>
```

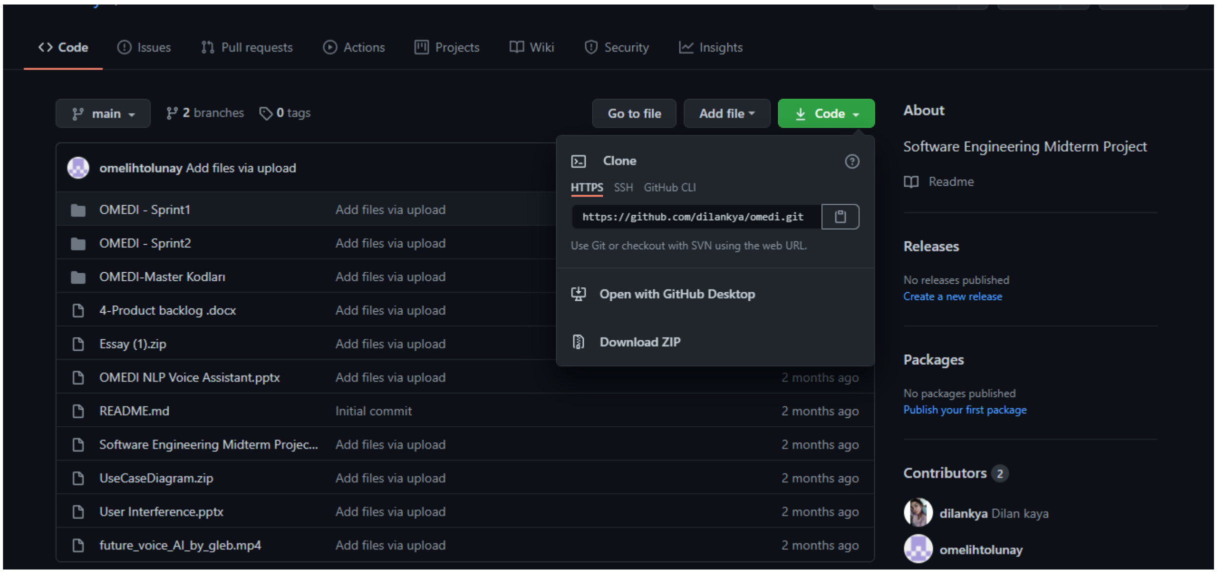
- ✓ The following codes are written in Windows PowerShell in order.
  - ➢ conda activate
  - ➢ conda config—set auto_activate_base false

iii. Cloning a repository using the command line

✓ On GitHub, navigate to the main page of the repository.Above the list of files, click **Code**.



✓ To clone the repository using HTTPS, under "Clone with HTTPS", click. To clone the repository using an SSH key, including a certificate issued by your organization's SSH certificate authority, click **Use SSH**, then click. To clone a repository using GitHub CLI, click **Use GitHub CLI**, then click.



omelihtolunay Add files via upload

OMEDI - Sprint1	Add files via upload	
OMEDI - Sprint2	Add files via upload	
OMEDI-Master Kodları	Add files via upload	
4-Product backlog .docx	Add files via upload	
Essay (1).zip	Add files via upload	
OMEDI NLP Voice Assistant.pptx	Add files via upload	2 months ago
README.md	Initial commit	2 months ago
Software Engineering Midterm Projec...	Add files via upload	2 months ago
UseCaseDiagram.zip	Add files via upload	2 months ago
User Interference.pptx	Add files via upload	2 months ago
future_voice_AI_by_gleb.mp4	Add files via upload	2 months ago

Clone

HTTPS SSH GitHub CLI

`https://github.com/dilankya/omedi.git`

Use Git or checkout with SVN using the web URL.

Open with GitHub Desktop

Download ZIP

About

Software Engineering Midterm Project

Readme

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Contributors 2

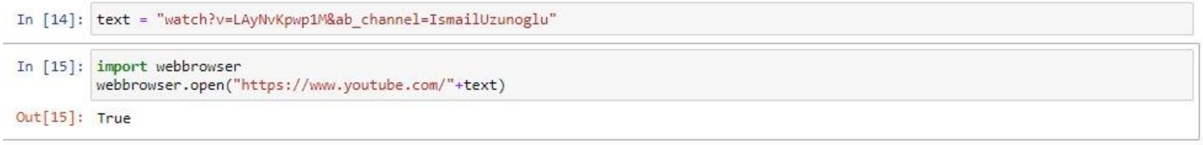
dilankya Dilan kaya
omelihtolunay

a. Open Git Bash.
b. Change the current working directory to the location where you want the cloned directory.
c. Type git clone, and then paste the URL you copied earlier.
   - $ git clone gh repo clone dilankya/omedi
d. Press **Enter** to create your local clone.

   -
     ```
     $ git clone gh repo clone dilankya/omedi
     Cloning into `Spoon-Knife`...
     > remote: Counting objects: 10, done.
     > remote: Compressing objects: 100% (8/8), done.
     > remove: Total 10 (delta 1), reused 10 (delta 1)
     > Unpacking objects: 100% (10/10), done.
     ```

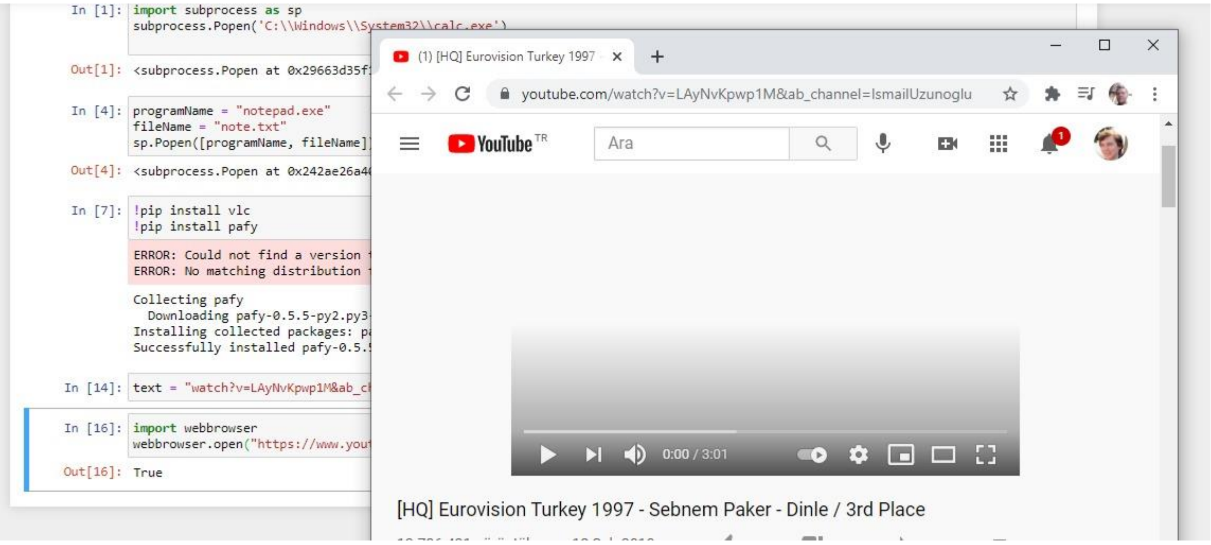
VII. PROGRAM USAGE

✓ First, we say the action we want to do. We said we wanted to open a song on YouTube.



```
In [14]: text = "watch?v=LAyNvKpwp1M&ab_channel=IsmailUzunoglu"

In [15]: import webbrowser
         webbrowser.open("https://www.youtube.com/"+text)

Out[15]: True
```

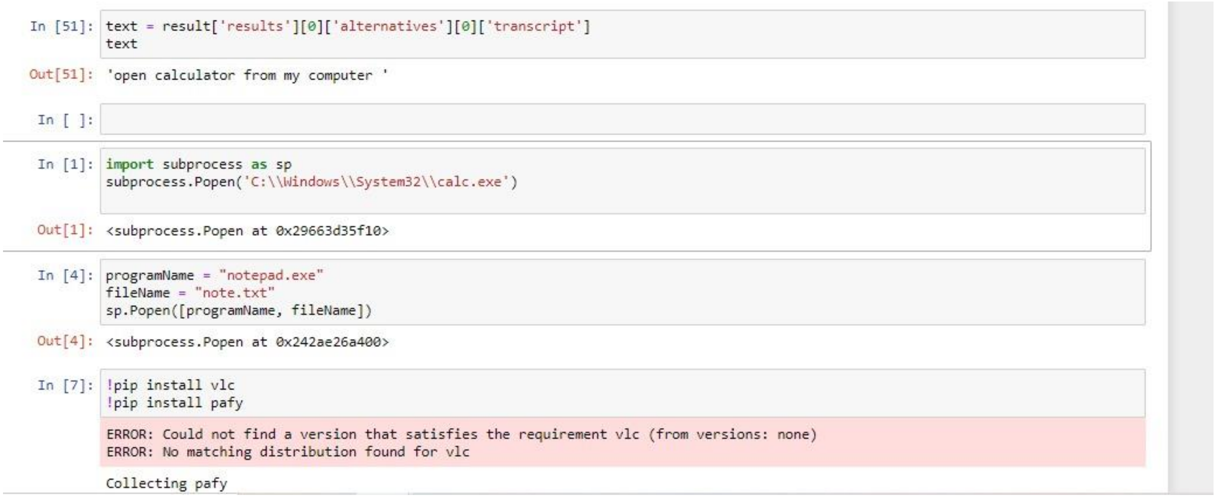
✓ The song we wanted was opened.



```
In [1]: import subprocess as sp
        subprocess.Popen('C:\\Windows\\System32\\calc.exe')

Out[1]: <subprocess.Popen at 0x29663d35f

In [4]: programName = "notepad.exe"
        fileName = "note.txt"
        sp.Popen([programName, fileName]

Out[4]: <subprocess.Popen at 0x242ae26a4

In [7]: !pip install vlc
        !pip install pafy

        ERROR: Could not find a version
        ERROR: No matching distribution

        Collecting pafy
          Downloading pafy-0.5.5-py2.py3
        Installing collected packages: p
        Successfully installed pafy-0.5.

In [14]: text = "watch?v=LAyNvKpwp1M&ab_c

In [16]: import webbrowser
         webbrowser.open("https://www.you

Out[16]: True
```

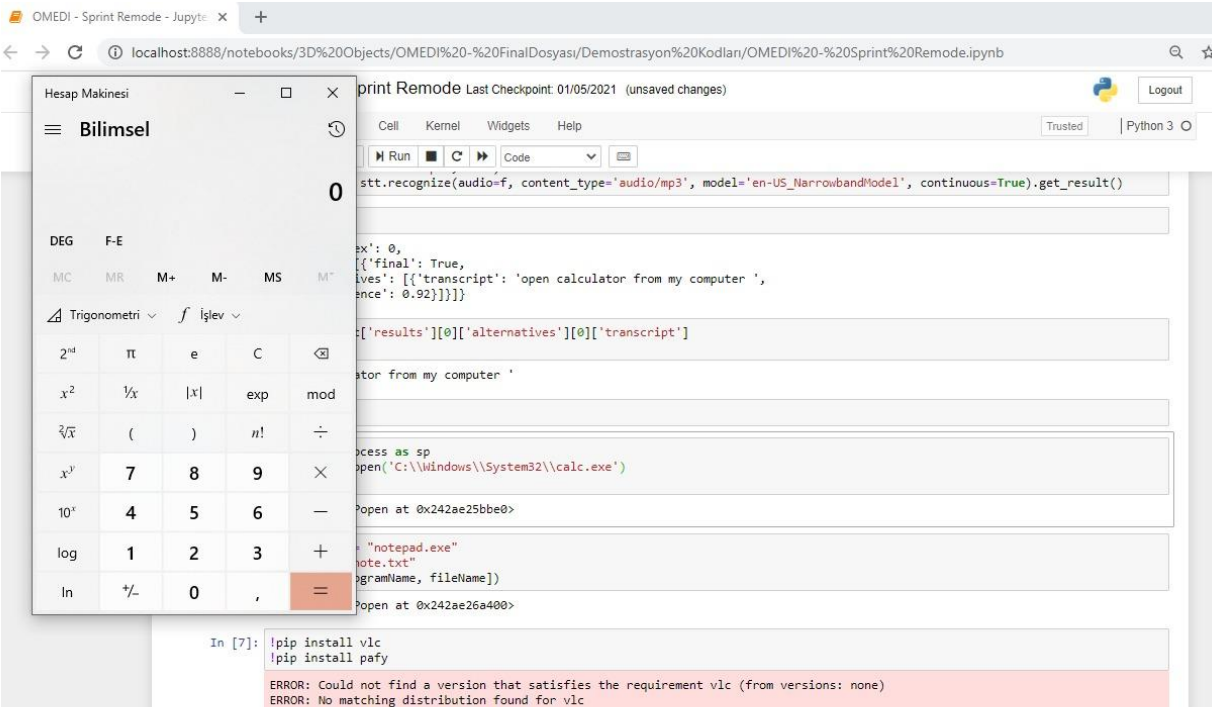
(1) [HQ] Eurovision Turkey 1997
youtube.com/watch?v=LAyNvKpwp1M&ab_channel=IsmailUzunoglu
YouTube TR
Ara
0:00 / 3:01
[HQ] Eurovision Turkey 1997 - Sebnem Paker - Dinle / 3rd Place

- ✓ Then we ask it to open the calculator.



```
In [51]: text = result['results'][0]['alternatives'][0]['transcript']
         text

Out[51]: 'open calculator from my computer '

In [ ]:

In [1]: import subprocess as sp
        subprocess.Popen('C:\\Windows\\System32\\calc.exe')

Out[1]: <subprocess.Popen at 0x29663d35f10>

In [4]: programName = "notepad.exe"
        fileName = "note.txt"
        sp.Popen([programName, fileName])

Out[4]: <subprocess.Popen at 0x242ae26a400>

In [7]: !pip install vlc
        !pip install pafy

        ERROR: Could not find a version that satisfies the requirement vlc (from versions: none)
        ERROR: No matching distribution found for vlc

        Collecting pafy
```

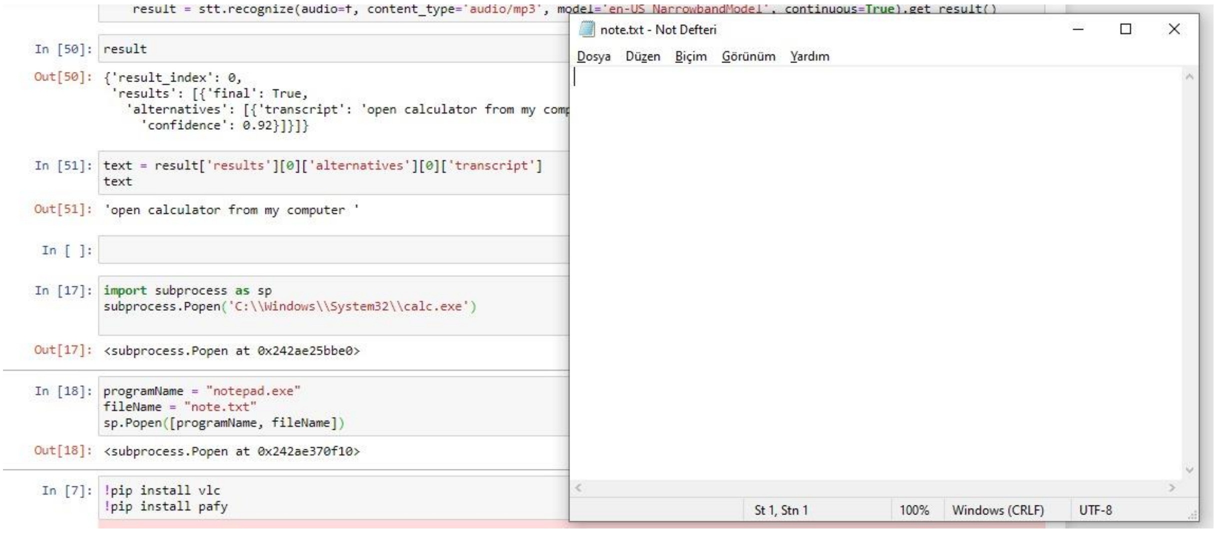
- ✓ And opens.



OMEDI - Sprint Remode - Jupyte
localhost:8888/notebooks/3D%20Objects/OMEDI%20-%20FinalDosyasi/Demostrasyon%20Kodlari/OMEDI%20-%20Sprint%20Remode.ipynb
print Remode Last Checkpoint: 01/05/2021 (unsaved changes)
Logout
Cell
Kernel
Widgets
Help
Trusted
Python 3
Run
Code
Hesap Makinesi
Bilimsel
0
DEG
F-E
MC
MR
M+
M-
MS
M˅
Trigonometri
İşlev
2nd
π
e
C
x²
¹⁄x
|x|
exp
mod
∛x
(
)
n!
÷
xʸ
7
8
9
×
10ˣ
4
5
6
—
log
1
2
3
+
ln
+/-
0
,
=

```
stt.recognize(audio=f, content_type='audio/mp3', model='en-US_NarrowbandModel', continuous=True).get_result()

ex': 0,
[{'final': True,
ives': [{'transcript': 'open calculator from my computer ',
ence': 0.92}]}]}

:['results'][0]['alternatives'][0]['transcript']

ator from my computer '

ocess as sp
pen('C:\\Windows\\System32\\calc.exe')

Popen at 0x242ae25bbe0>

 "notepad.exe"
ote.txt"
gramName, fileName])

Popen at 0x242ae26a400>

In [7]: !pip install vlc
        !pip install pafy

        ERROR: Could not find a version that satisfies the requirement vlc (from versions: none)
        ERROR: No matching distribution found for vlc
```

- ✓ Finally, let's open the notebook.



```
    result = stt.recognize(audio=f, content_type='audio/mp3', model='en-US_NarrowbandModel', continuous=True).get_result()

In [50]: result

Out[50]: {'result_index': 0,
           'results': [{'final': True,
             'alternatives': [{'transcript': 'open calculator from my comp
               'confidence': 0.92}]}]}

In [51]: text = result['results'][0]['alternatives'][0]['transcript']
         text

Out[51]: 'open calculator from my computer '

In [ ]:

In [17]: import subprocess as sp
         subprocess.Popen('C:\\Windows\\System32\\calc.exe')

Out[17]: <subprocess.Popen at 0x242ae25bbe0>

In [18]: programName = "notepad.exe"
         fileName = "note.txt"
         sp.Popen([programName, fileName])

Out[18]: <subprocess.Popen at 0x242ae370f10>

In [7]: !pip install vlc
        !pip install pafy
```