

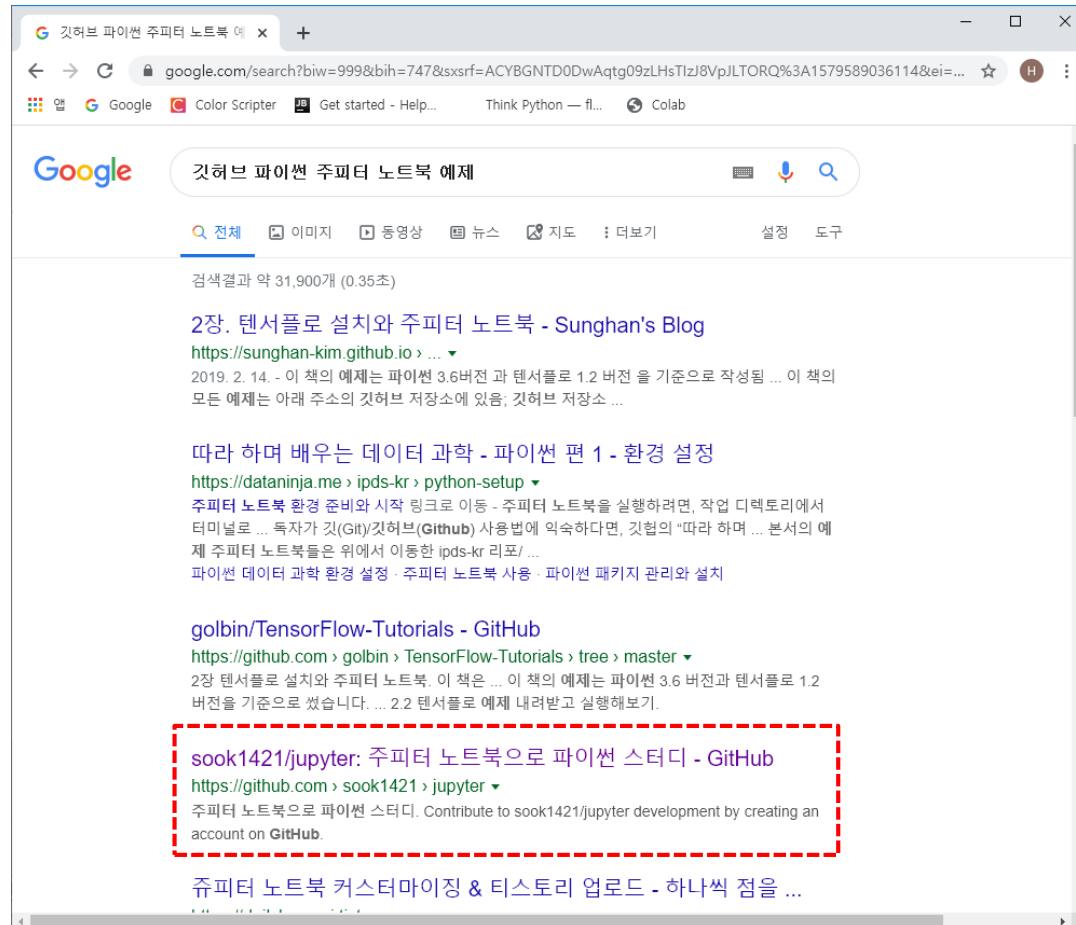
Colab 활용

인공지능소프트웨어학과
강 환수 교수

깃허브의 노트 파일을 Colab에서 활용

깃허브 사이트 검색

- <https://github.com/sook1421/jupyter>



노트 열기

- sook1421/jupyter

The screenshot shows the Google Colaboratory web interface. The browser tabs include 'test.ipynb - Colaboratory', 'pyrand.ipynb - Colaboratory', 'myColab - Google Drive', and 'Untitled0.ipynb - Colaboratory'. The address bar shows the URL 'colab.research.google.com/drive/1UZAJaglmS8IK0DavaUUz9U_UdAf1AaW5'. The 'File' menu is open, showing options like '드라이브에서 찾기', '실습 모드에서 열기', '새 Python 3 노트', '새 Python 2 노트', '노트 열기...', '노트 업로드...', '이름 바꾸기...', '휴지통으로 이동', '드라이브에 사본 저장...', 'GitHub Gist로 사본 저장...', 'GitHub에 사본 저장...', '저장', '버전 저장 및 고정', '업데이트 기록', '.ipynb 다운로드', '.py 다운로드', '드라이브 미리보기 업데이트', and '인쇄'. The 'Open' dialog is displayed, showing the search results for 'sook1421/jupyter'. The search results include 'html을 가져와서 파일로 저장.ipynb', 'matplotlib.ipynb', '기본 문법 알아보기 (파이썬).ipynb', and '문자열 자료형.ipynb'. The '기본 문법 알아보기 (파이썬).ipynb' file is highlighted, and the 'Open' button is visible.

파일 열기 후

모두 실행

The screenshot shows a Google Colab notebook interface. The browser address bar indicates the URL: `colab.research.google.com/github/sook1421/jupyter/blob/master/기본%20문법%20알아보기%20(파이썬).ipynb`. The notebook title is "기본 문법 알아보기 (파이썬).ipynb". The interface includes tabs for "코드" (Code), "텍스트" (Text), and "드래그" (Drag). A context menu is open over the code cells, displaying the following options:

- 모두 실행 (Ctrl+F9)
- 이전 셀 실행 (Ctrl+F8)
- 초점이 맞춰진 셀 실행 (Ctrl+Enter)
- 선택항목 실행 (Ctrl+Shift+Enter)
- 이후 셀 실행 (Ctrl+F10)
- 실행 중단 (Ctrl+M |)
- 런타임 다시 시작... (Ctrl+M .)
- 다시 시작 및 모두 실행...
- 런타임 초기화
- 런타임 유형 변경
- 세션 관리
- 런타임 로그 보기

The notebook content includes:

- 1 세미콜론**: A section explaining that Python uses semicolons to separate multiple statements on the same line. It shows two code cells:


```
[ ] print('Hello, world!')
```

 and


```
[ ] print('Hello, world!'); print('Hello, world')
```

 Both cells show the output "Hello, world!".
- 여러 구분을 사용할 때 세미콜론으로 구분**: A section explaining that semicolons are used to separate multiple statements on the same line. It shows a code cell:


```
[ ] print('Hello!'); print('JeongSook')
```

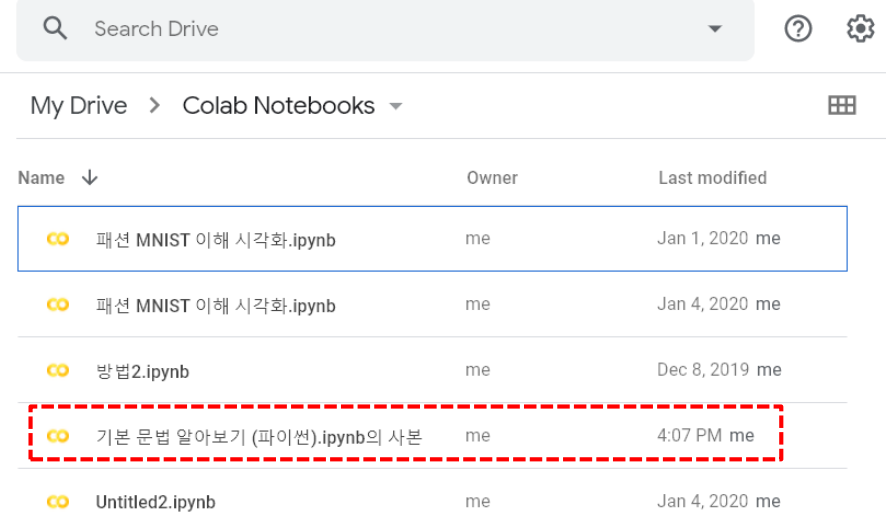
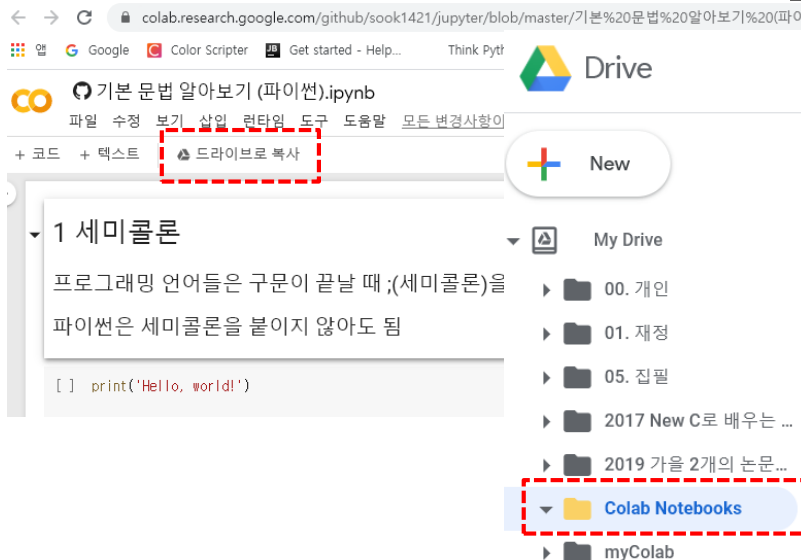
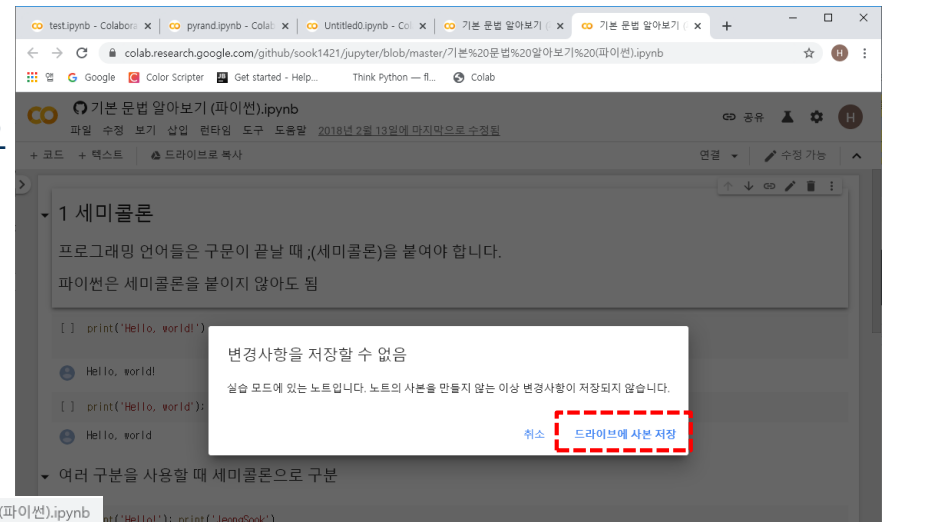
 The output is:


```
Hello!
      JeongSook
```
- 2 주석**: A section explaining that comments do not affect program execution. It mentions that there are two types of comments: line comments and block comments.

내 파일로 저장

• 저장

- '드라이브에 사본 저장...'이 먼저 필요
- 드라이브의 다음 폴더에 저장
 - Colab Notebooks
- 버튼 메뉴
 - 드라이버로 복사 사용 권장



실습

- '케라스 창시자에게 배우는 딥러닝 깃허브' 검색
 - <https://github.com/gilbutITbook/006975>

첫 파일 열기

• 2.1-a-first-look-at-a-neural-network.ipynb

예

최근 사용

Google 드라이브

GitHub

업로드

GitHub URL을 입력하거나 조직 또는 사용자로 검색하세요.

☒ 비공개 저장소 포함

gilbutITbook/006975

🔍

저장소:

gilbutITbook/006975 ▼

브랜치:

master ▼

경로

2.1-a-first-look-at-a-neural-network.ipynb

3.4-classifying-movie-reviews.ipynb

3.5-classifying-newswires.ipynb

3.6-predicting-house-prices.ipynb

취소

Python

8

자신 드라이브에 저장

2.1-a-first-look-at-a-neural-network.ipynb의 사본 ☆

파일 수정 보기 삽입 런타임 도구 도움말 오후 4:43에 마지막으로 저장됨

+ 코드 + 텍스트

```
import keras
keras.__version__
```

Using TensorFlow backend.
'2.2.2'

신경망과의 첫 만남

이 노트북은 [케라스 창시자에게 배우는 딥러닝](#) 책의 2장 1절의 코드 예제입니다. 책에는 더 많은 내용과 그림이 있습니다. 이 노트북에는 소스 코드에 관련된 설명만 포함합니다.

케라스 파이썬 라이브러리를 사용하여 손글씨 숫자 분류를 학습하는 구체적인 신경망 예제를 살펴보겠습니다. 케라스나 비슷한 라이브러리를 사용한 경험이 없다면 당장은 이 첫 번째 예제를 모두 이해하지 못할 것입니다. 아직 케라스를 설치하지 않았을지도 모릅니다. 괜찮습니다. 다음 장에서 이 예제를 하나하나 자세히 설명합니다. 코드가 좀 이상하거나 요술처럼 보더라도 너무 걱정하지 마세요. 일단 시작해 보겠습니다.

여기에서 돌리고 하는 문제는 흑백 손글씨 숫자 이미지(28x28 픽셀)를 10개의 범주(0에서 9까지)로 분류하는 것입니다. 머신 러닝 커뮤니티에서 고전으로 취급받는 데이터셋인 MNIST를 사용하겠습니다. 이 데이터셋은 머신 러닝의 역사만큼 오래되었고 많은 연구에 사용되었습니다. 이 데이터셋은 1980년대에 미국 국립표준기술연구소에서 수집한 6만 개의 훈련 이미지와 1만 개의 테스트 이미지로 구성되어 있습니다. MNIST 문제를 알고리즘이 제대로 작동하는지 확인하기 위한 딥러닝계의 'Hello World'라고 생각해도 됩니다. 머신 러닝 기술자가 되기까지 연구 논문이나 블로그 포스트 등에서 MNIST를 보고 또 보게 될 것입니다.

MNIST 데이터셋은 넘파이 배열 형태로 케라스에 이미 포함되어 있습니다:

```
[ ] from keras.datasets import mnist

(train_images, train_labels), (test_images, test_labels) = mnist.load_data()
```


train_images와 train_labels가 모델이 학습해야 할 훈련 세트를 구성합니다. 모델은 test_images와 test_labels를 사용하여 테스트합니다.

My Drive > Colab Notebooks

Name	Owner	Last modified	
2.1-a-first-look-at-a-neural-network.ipynb의 사본	me	4:43 PM	
기본 문법 알아보기 (파이썬).ipynb의 사본	me	4:07 PM	

노트 파일의 깃허브 URL을
Colab으로 열기

노트 파일을 읽는 방법

- 단일 .ipynb 파일을 clone 하는 방법
 - <https://github.com/~ ~ ~> 에서 ~ ~ ~ 부분을
 - <https://colab.research.google.com/github/~ ~ ~> 로 교체
- 
- The diagram illustrates the process of replacing the user part of a GitHub URL. An orange arrow points from the '~ ~ ~' placeholder in the first URL to the '~ ~ ~' placeholder in the second URL, indicating that the user part of the GitHub URL should be replaced with the Colab GitHub path.

깃허브 노트 바로 코랩에서 열기

- 내가 알고 있는 노트 파일 url
 - <https://github.com/zzsza/TIL/blob/master/python/tensorflow-1.ipynb>
 - 노트저장소
 - 깃허브 서버 정보 빼고 다음 정보 확인
 - </zzsza/TIL/blob/master/python/tensorflow-1.ipynb>
- 코랩 url 이후 /github 를 붙이고 계속 붙이기
 - <https://colab.research.google.com>
 - /github 추가
 - <https://colab.research.google.com/github>
- 다음 주소로 바로 접근
 - https://Colab주소/github/id/노트저장소/*.ipynb
 - <https://colab.research.google.com/github/zzsza/TIL/blob/master/python/tensorflow-1.ipynb>

코랩 접속 화면

The screenshot displays the Google Colaboratory web interface in a browser. The address bar shows the URL: `colab.research.google.com/github/zzsza/TIL/blob/master/python/tensorflow-1.ipynb`. The notebook title is `tensorflow-1.ipynb`. The interface includes a toolbar with icons for file operations, a code editor, and a console output area.

```

import tensorflow as tf

[ ] hello = tf.constant("Hi")

[ ] print(hello)

Tensor("Const:0", shape=(), dtype=string)

[ ] # Tensor : 자료형

[ ] a = tf.constant(10)
    b = tf.constant(2)
    c = a + b

[ ] print(c)

Tensor("add:0", shape=(), dtype=int32)

[ ] # 텐서플로우는 그래프 생성 후, 그래프 실행하는 과정을 거쳐야 함. 지연 실행 방법을 사용

[ ] sess = tf.Session()
    print(sess.run(c))

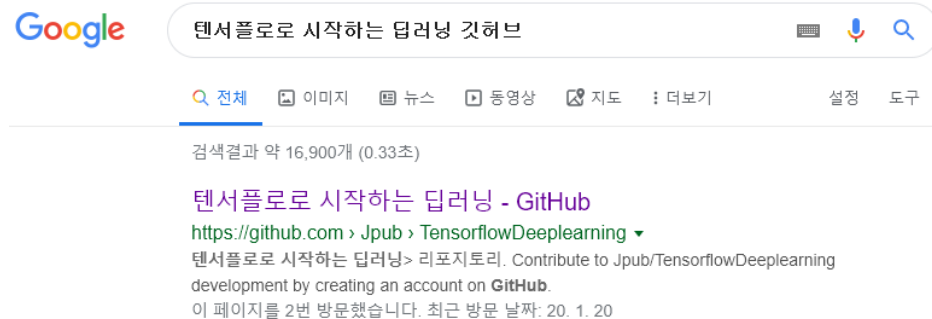
12

[ ] print(sess.run(hello))

b'Hi'
  
```

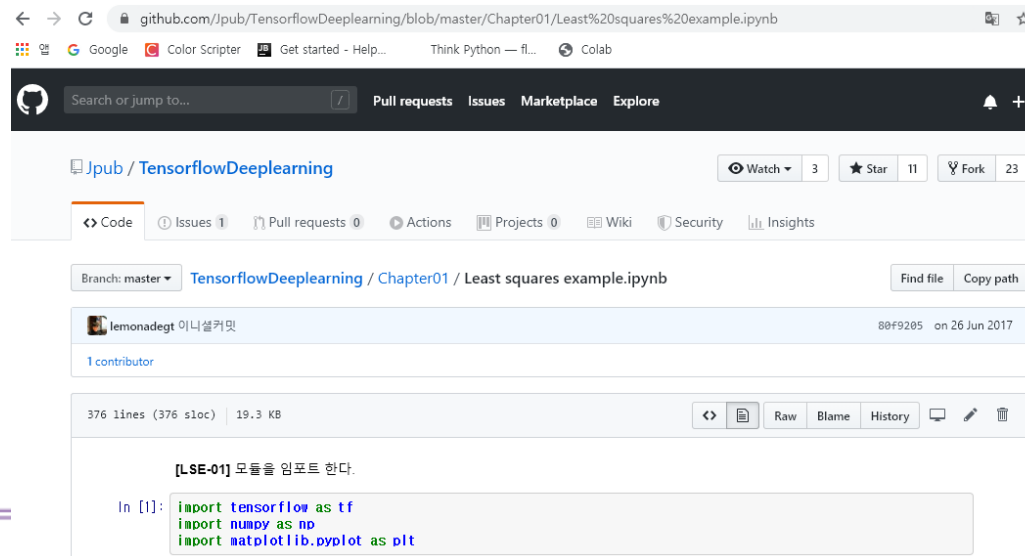
실습

• 텐서플로로 시작하는 딥러닝 깃허브



• 깃허브 페이지

- <https://github.com/Jpub/TensorflowDeeplearning/blob/master/Chapter01/Least%20squares%20example.ipynb>



접속 url

- 코랩 url

- <https://colab.research.google.com>

- 노트 url

- [/Jpub/TensorflowDeeplearning/blob/master/Chapter01/Least%20squares%20example.ipynb](https://colab.research.google.com/Jpub/TensorflowDeeplearning/blob/master/Chapter01/Least%20squares%20example.ipynb)

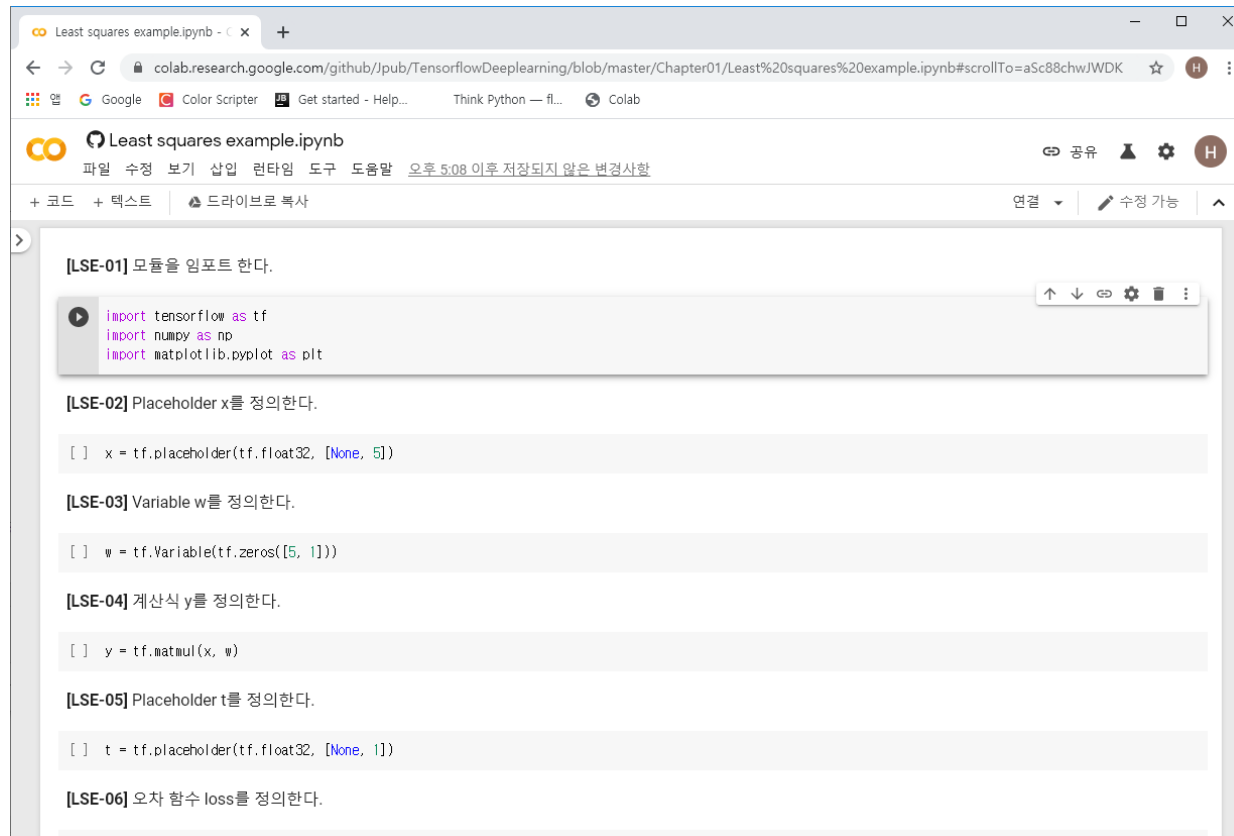
- 결과

- [/github 추가](#)
- <https://colab.research.google.com/github/Jpub/TensorflowDeeplearning/blob/master/Chapter01/Least%20squares%20example.ipynb#scrollTo=2ULb6gM5JWDO>

열기 성공

- 주소

- <https://colab.research.google.com/github/Jpub/TensorflowDeeplearning/blob/master/Chapter01/Least%20squares%20example.ipynb#scrollTo=2ULb6gM5JWDO>



```
[LSE-01] 모듈을 임포트 한다.
import tensorflow as tf
import numpy as np
import matplotlib.pyplot as plt

[LSE-02] Placeholder x를 정의한다.
[ ] x = tf.placeholder(tf.float32, [None, 5])

[LSE-03] Variable w를 정의한다.
[ ] w = tf.Variable(tf.zeros([5, 1]))

[LSE-04] 계산식 y를 정의한다.
[ ] y = tf.matmul(x, w)

[LSE-05] Placeholder t를 정의한다.
[ ] t = tf.placeholder(tf.float32, [None, 1])

[LSE-06] 오차 함수 loss를 정의한다.
```

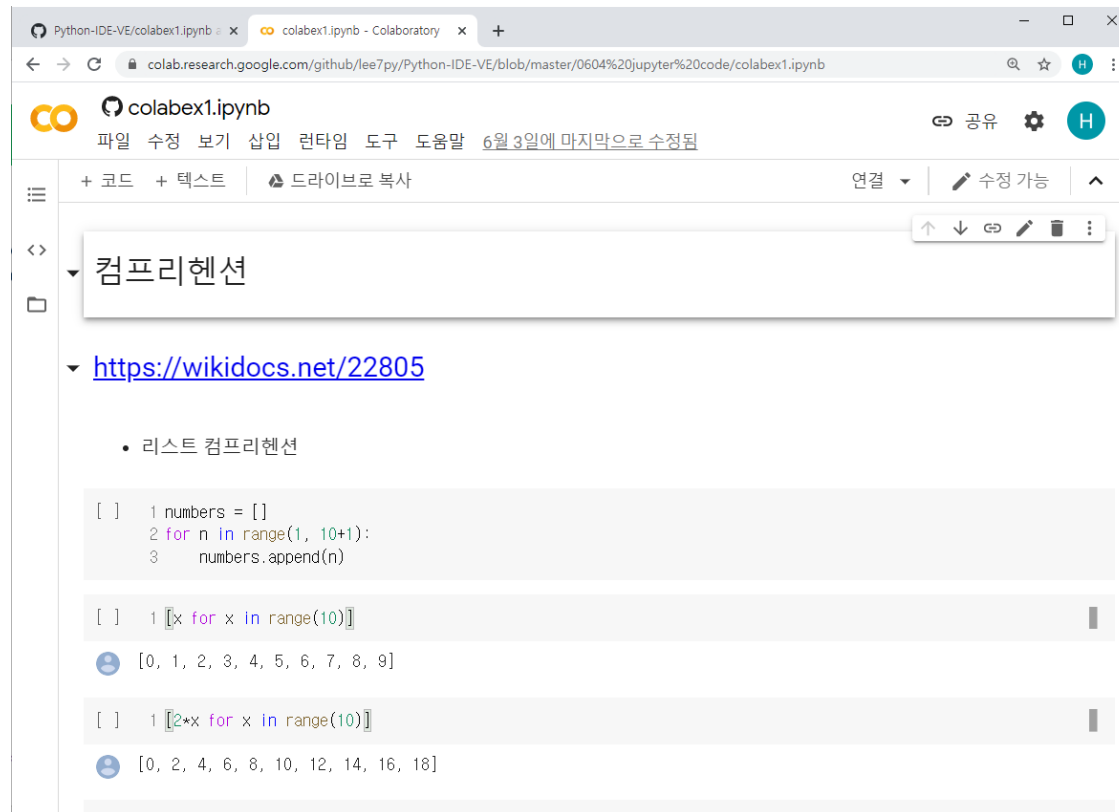

실습

- 우리 수업 사이트
 - <https://github.com/lee7py/2021-py-IDE-VE/>
 - 파일
 - <https://github.com/lee7py/2021-py-IDE-VE/blob/master/>

우리 자료 코랩으로 열기

- URL

- <https://colab.research.google.com/github/lee7py/Python-IDE-VE/blob/master/0604%20jupyter%20code/colabex1.ipynb>



Quiz

- 다음 설명 중 잘못된 것은?

- 인터넷 상에서 오픈한 주피터 노트북 파일은 코랩에서 사용할 수 없다.
- 코랩에서 바로 깃허브의 주피터 노트북 파일을 열 수 있다.
- 코랩에서 바로 깃허브 id만으로 주피터 노트북 파일을 찾아 열 수 있다.
- 코랩에서 작성한 파일도 주피터 노트북에서 사용할 수 있다.

구글 Colab 노트
Github에 저장

자신의 깃허브에 저장소를 하나 생성

- Git7py에 하나 생성
 - Python ide ve

Create a New Repository

github.com/new

Search or jump to...

Pull requests Issues Marketplace Explore

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner: git7py / Repository name: python ide ve ✓

Great repository name: Your new repository will be created as python-ide-ve. about super-duper-potato?

Description (optional)

☒ Public
Anyone can see this repository. You choose who can commit.

☐ Private
You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

☐ Initialize this repository with a README
This will let you immediately clone the repository to your computer.

Add .gitignore: None Add a license: None ⓘ

Create repository

Welcome에서 Pandas 소개로 이동

- 다양한 자습 리소스 지원

머신러닝 단기집중과정

다음은 Google 온라인 머신러닝 과정에서 가져온 일부 노트입니다. [전체 과정 웹사이트](#)에서 자세한 내용을 확인하세요.

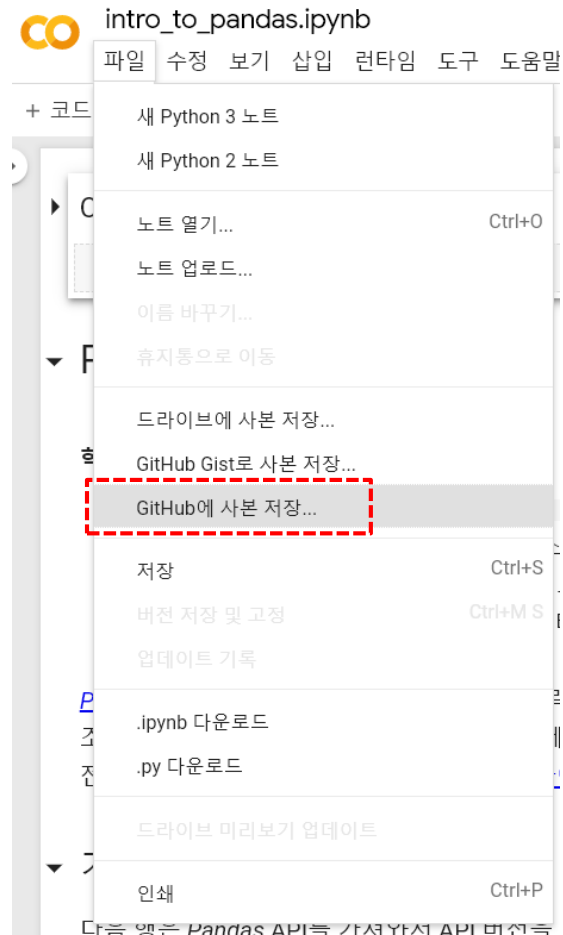
- [Pandas 소개](#)
- [TensorFlow 개념](#)
- [TensorFlow 첫걸음](#)
- [신경망 소개](#)
- [희소 데이터 및 임베딩 소개](#)

가속 하드웨어 사용하기

- [GPU를 사용한 TensorFlow](#)
- [TPU를 사용한 TensorFlow](#)

노트 파일에서 메뉴 선택


- 파일 하부
 - Github에 사본 저장




깃허브의 저장소로 저장

- 깃허브 저장소 입력
 - 인증(로그인) 필요

GitHub으로 복사

저장소:  hskang7/test ▼

브랜치:  master ▼

파일 경로
intro_to_pandas.ipynb

변경사항 설명 메시지
Colaboratory를 통해 생성됨


☒ Colaboratory 링크 추가

취소

확인

GitHub으로 복사

저장소:  git7py/python-ide-ve ▼

브랜치:  master ▼

파일 경로
intro_to_pandas.ipynb

변경사항 설명 메시지
Colaboratory를 통해 생성됨

☒ Colaboratory 링크 추가

취소

확인

하부 폴더에 저장

- 파일 이름
 - 폴더명/파일명.ipynb
 - 폴더명이 없는 경우
 - 새로운 폴더를 만들고 파일이 저장됨
- 주의 사항
 - 맨 앞에 / 넣으면 오류발생

자신의 저장소 확인, 예 1

- hskang7/test

hskang7 / test

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security Insights Settings

Branch: master test / intro_to_pandas.ipynb Find file Copy path

hskang7 Colaboratory를 통해 생성됨 f57e526 5 minutes ago

1 contributor

656 lines (656 sloc) 19.6 KB

Open in Colab

Copyright 2017 Google LLC.

In [0]:

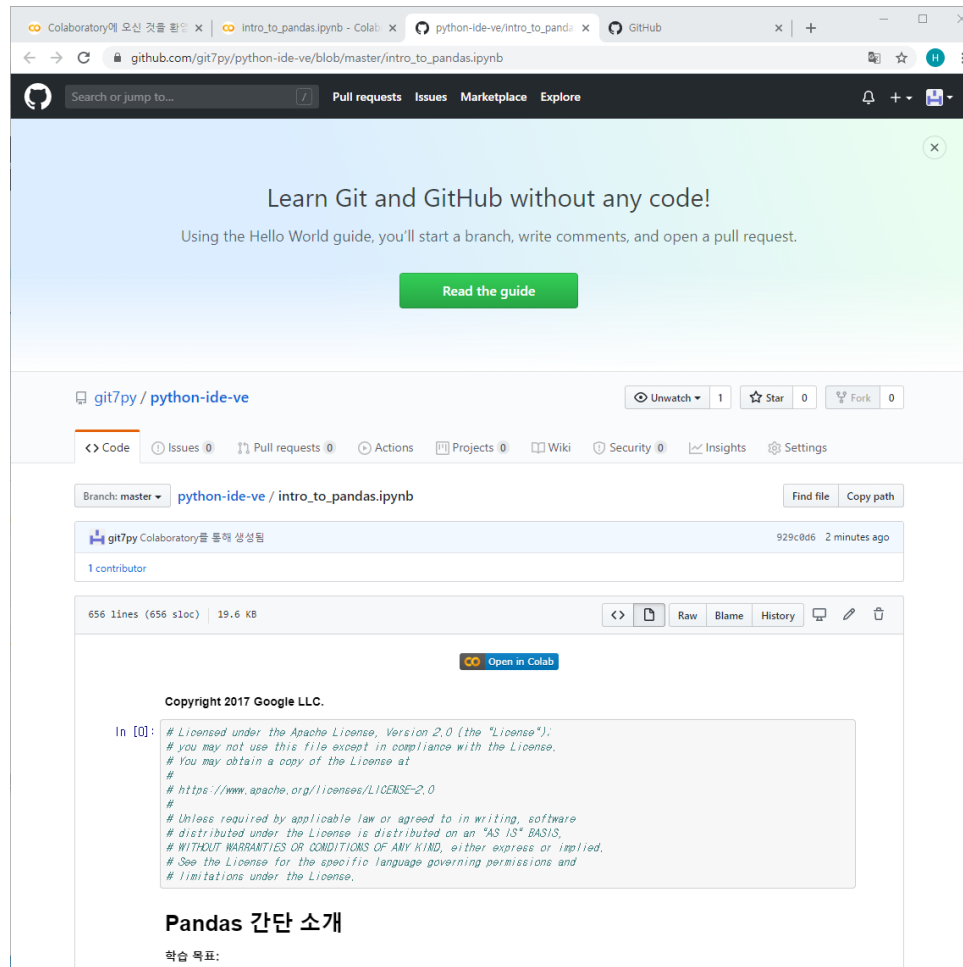
```
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
# https://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
```

Pandas 간단 소개

학습 목표:

자신의 저장소 확인, 예 2

- 복사가 성공하면 자동으로 브라우저 활성화



구글 Colab 노트
깃허브 Gist에 저장

Welcome에서 Tensorflow 개념으로 이동

- 다양한 자습 리소스 지원

머신러닝 단기집중과정

다음은 Google 온라인 머신러닝 과정에서 가져온 일부 노트입니다. [전체 과정 웹사이트](#)에서 자세한 내용을 확인하세요.

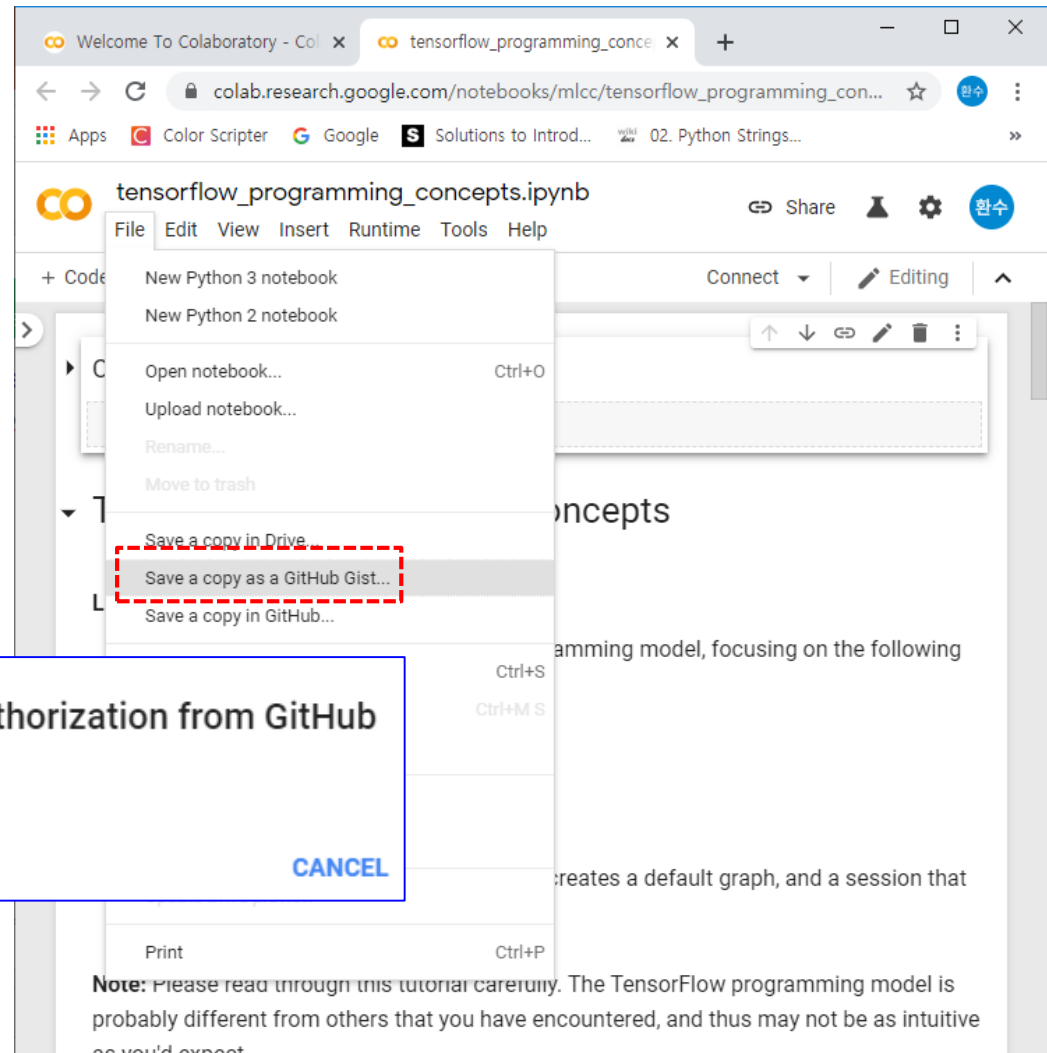
- [Pandas 소개](#)
- [TensorFlow 개념](#)
- [TensorFlow 첫걸음](#)
- [신경망 소개](#)
- [희소 데이터 및 임베딩 소개](#)

가속 하드웨어 사용하기

- [GPU를 사용한 TensorFlow](#)
- [TPU를 사용한 TensorFlow](#)

노트 파일에서 메뉴 선택

- 파일 하부
 - Github Gist로 사본 저장
- 깃허브 인증 필요





깃허브 인증

- 로그인

Sign in to GitHub · GitHub - Google Chrome

github.com/login?client_id=5036cf6d81e65aaa6340&...





Sign in to **GitHub**
to continue to **Colaboratory**

Username or email address

Password [Forgot password?](#)

Sign in

New to GitHub? [Create an account.](#)

자신의 깃허브 Gist에서 확인

- Gist 주소: gist.github.com/사용자id
 - gist.github.com/hskang7

비밀이라 공개는 안됨

hskang7's gists

gist.github.com/hskang7

Apps Color Scripter Google Solutions to Introd... 02. Python Strings... DeepLearningZero... Tigercow.Door :: 텐...

GitHub Gist Search... All gists Back to GitHub

All gists 7

Type: All Sort: Recently created

hskang7 / tensorflow_programming_concepts.ipynb Secret 1 file 0 forks 0 comments 0 stars

Created 6 minutes ago

tensorflow_programming_concepts.ipynb

hskang7

View GitHub Profile

Open in Colab

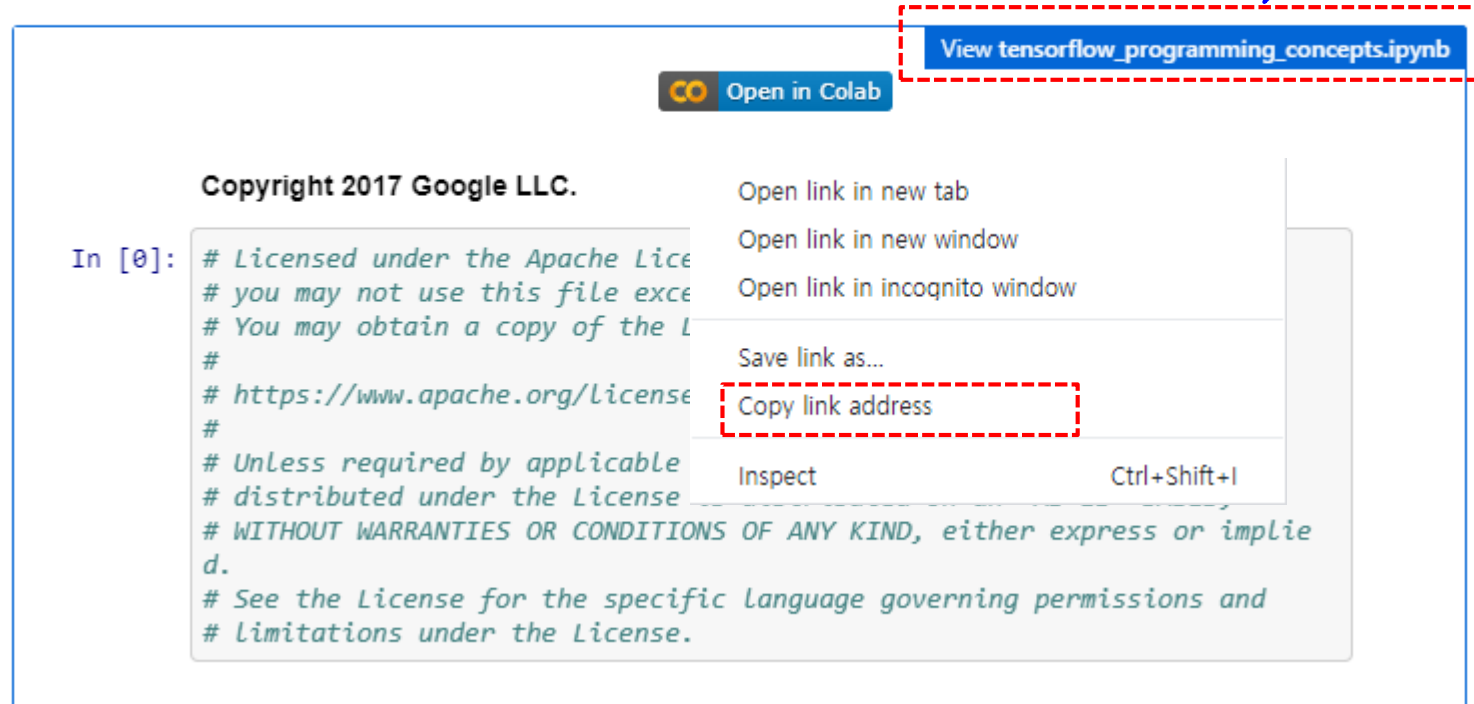
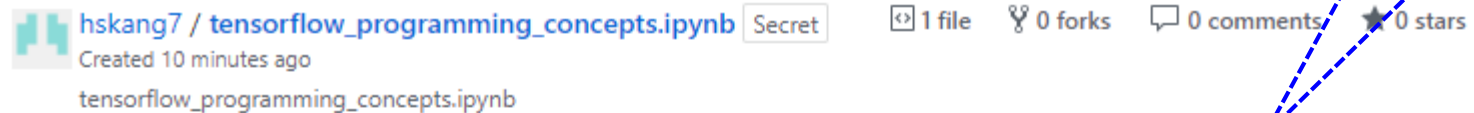
Copyright 2017 Google LLC.

```
In [0]: # Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
# https://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
```

Colab에서
소스 열기

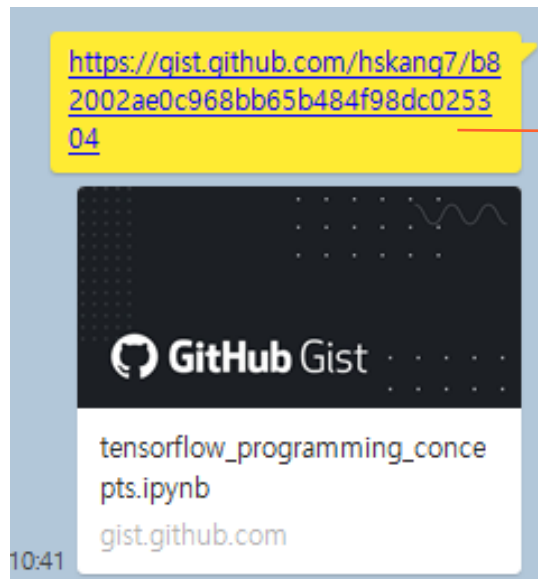
Gist 활용 방법

- 소스의 'Open in Colab'에서
 - 오른쪽 팝업 메뉴에서 '링크 주소 복사' 선택



카톡 등에서 복사하면

- 바로 소스로 이동할 수 있는 링크 제공



Colab에서 열기 가능

GitHub Gist Search... All gists Back to GitHub Sign in

Instantly share code, notes, and snippets.

hskang7 / tensorflow_programming_concepts.ipynb Secret

Created 24 minutes ago

Code Revisions 2 Embed <script src="https://gist." >

tensorflow_programming_concepts.ipynb

tensorflow_programming_concepts.ipynb

Open in Colab

Copyright 2017 Google LLC.

```
In [0]: # Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
# https://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
```

TensorFlow Programming Concepts

Learning Objectives:

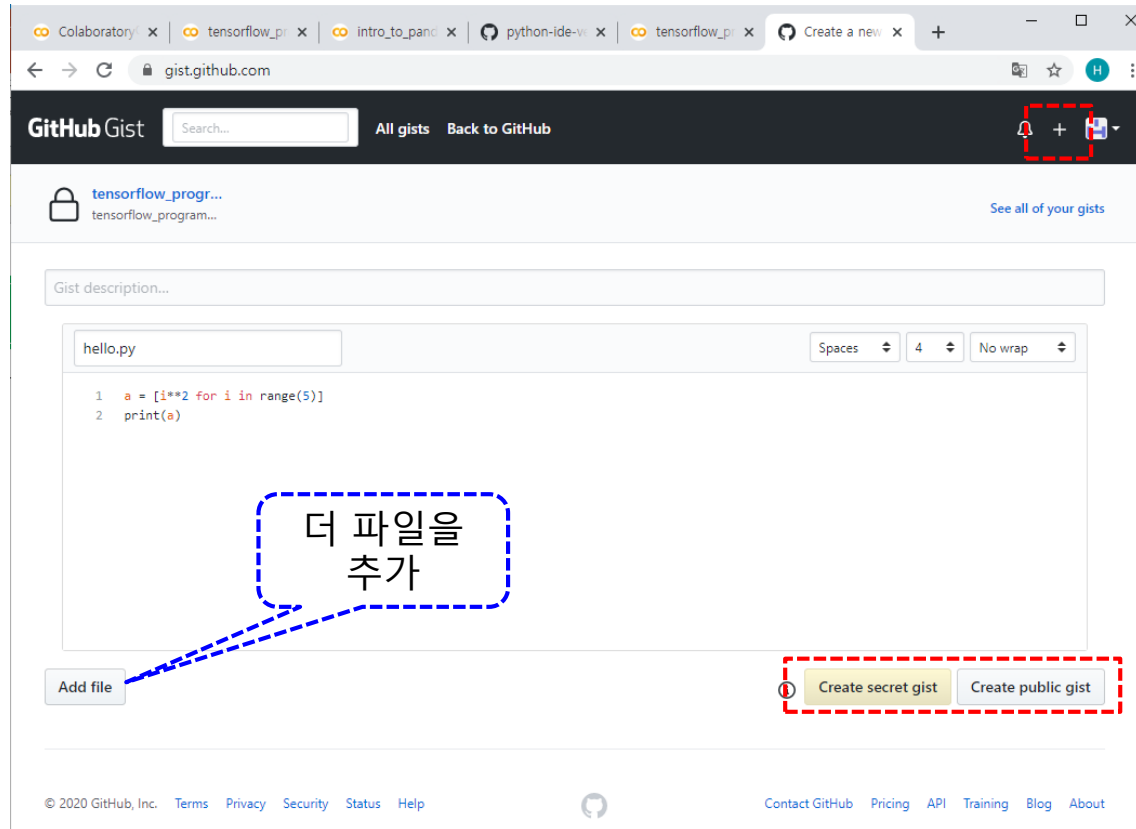
- Learn the basics of the TensorFlow programming model, focusing on the following concepts:
 - tensors
 - operations
 - graphs
 - sessions
- Build a simple TensorFlow program that creates a default graph, and a session that runs the graph

Github Gist 개념

- 깃허브에서 제공하는 링크 공유 서비스
 - 블로그 등에 부분 소스 코드(snippets)로 보기 좋게 붙여 놓는 방법
- Secret과 public
 - Secret
 - 검색 엔진에는 노출이 안되지만 당신이 URL을 알려준 사람에게는 보여주는 수준

간단한 gist 생성

- 파일 이름, 확장자 붙이기



Html 파일에 삽입

- 태그 복사하여 붙여넣기, 브라우저로 확인

The image is a composite of three screenshots illustrating the steps to embed a GitHub Gist into an HTML file.

Top Screenshot: A web browser window showing a GitHub Gist page for a file named `hello.py`. The URL is `gist.github.com/git7py/3f313c23f7e9899cdd9f223bb60676de`. The page includes options to `Edit`, `Delete`, `Unsubscribe`, and `Star`. The `Embed` dropdown menu is open, and the `<script>` option is selected, which is highlighted with a red dashed box. The generated script tag is visible: `<script src="https://gist.github.com/git7py/3f313c23f7e9899cdd9f223bb60676de"></script>`.

Bottom-Left Screenshot: A Windows Notepad window titled `hello.html - Windows 메모장`. The menu bar includes `파일(F)`, `편집(E)`, `서식(O)`, `보기(V)`, and `도움말(H)`. The text content of the file is the script tag copied from the Gist page: `<script src="https://gist.github.com/git7py/3f313c23f7e9899cdd9f223bb60676de"></script>`. The status bar at the bottom indicates `Ln 1, Col 57` and `100%` zoom.

Bottom-Right Screenshot: A web browser window showing the result of the embedding. The address bar displays `file:///D:/0 나스`. A message in Korean states: `북마크바로 지정된 폴더에 북마크가 없습니다. 북마크를 추가해 보세요. 북마크...`. The main content area shows the Python code from the `hello.py` file:

```
1 a = [i**2 for i in range(5)]
2 print(a)
```

 Below the code, it says `hello.py hosted with ❤ by GitHub` with a `view raw` link. The right sidebar contains icons for GitHub, a notification bell, and a plus sign for additional actions.

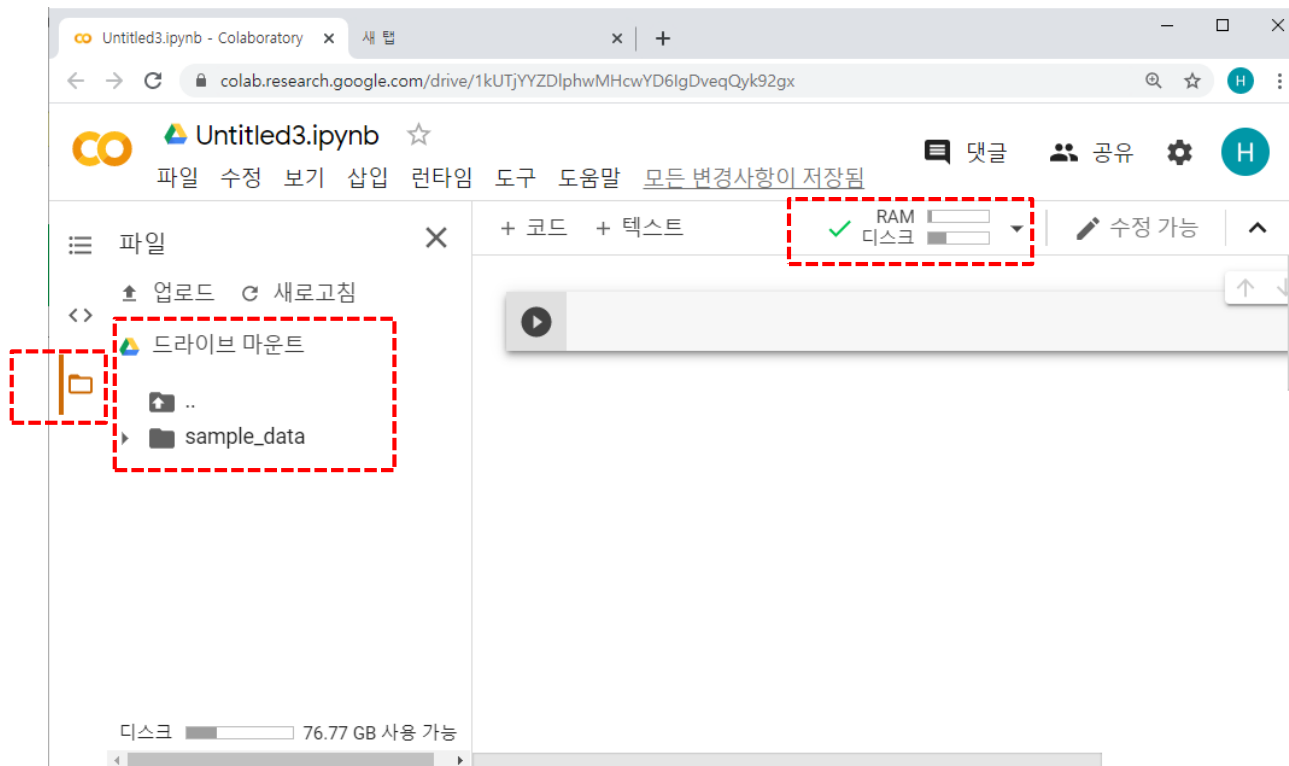
Quiz

- 다음 설명 중 잘못된 것은?
- (3)
 - 코랩에서 편집하던 주피터 노트북 파일은 자신의 깃허브에 저장할 수 있다.
 - 공개된 깃허브의 주피터 노트북 파일은 주소를 수정해서 바로 코랩에서 열 수 있다.
 - 깃허브의 gist에 있는 코드는 모두 공개되어 있다.
 - 코랩에서 편집하던 주피터 노트북 파일은 자신의 깃허브 gist에 저장할 수 있다.

Colab 서버의 폴더에
구글 드라이브를
연결(마운트) 하기

시작

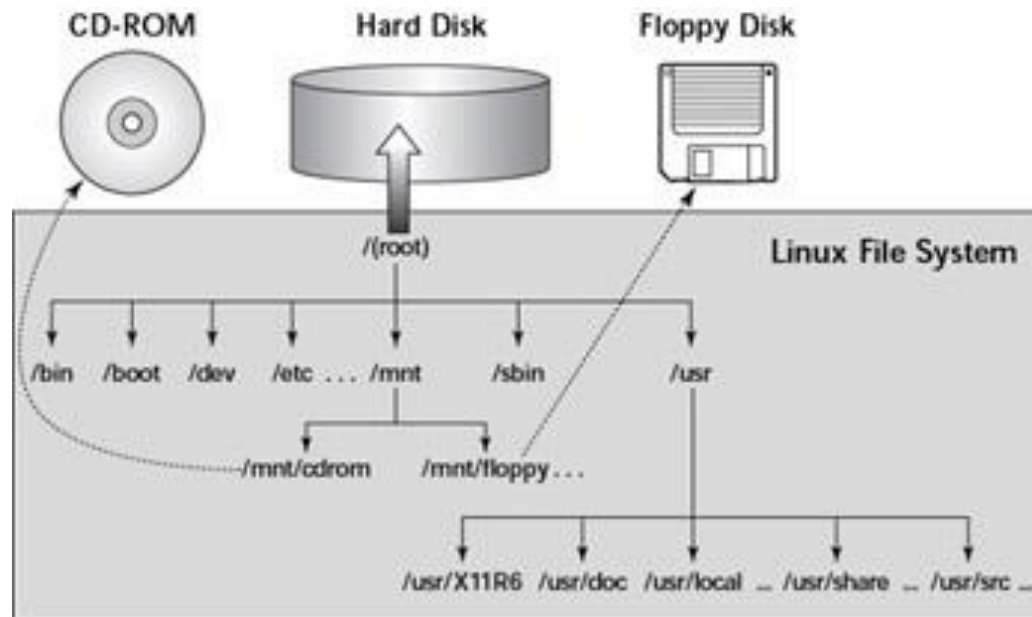
- 메뉴 파일 | 새 노트
 - 연결
 - 왼쪽 폴더 모양 클릭



마운트란?

- **마운트(mount) 의미**

- 물리적인 장치(예로 구글 드라이브)를 특정 위치 즉 디렉터리에 연결시켜주는 것



드라이브 마운트

- 자신의 구글 드라이브에 접속 후
 - 왼쪽 드라이브 마운트 클릭

자신의 구글 드라이브를 colab 시스템의 drive 폴더와 연결

The image illustrates the process of mounting Google Drive in Google Colab. It consists of several parts:

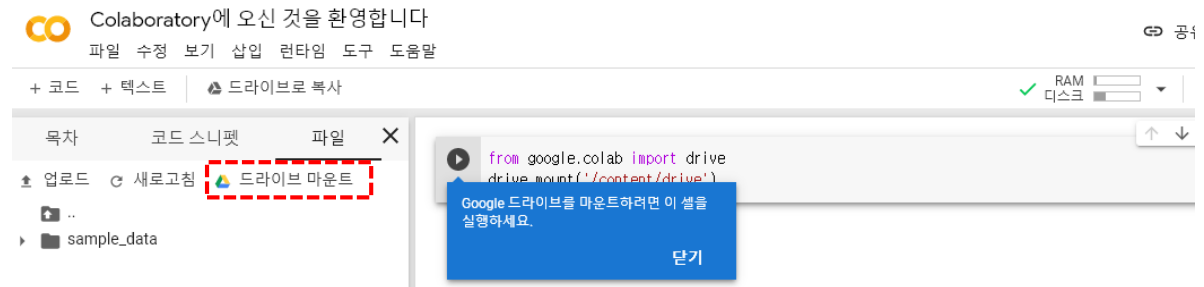
- File Explorer (Left):** Shows the '파일' (File) menu with options like '업로드' (Upload) and '새로고침' (Refresh). The '드라이브 마운트' (Mount Drive) option is highlighted with a red dashed box. Below it, the 'sample_data' folder is visible.
- Code Block:** Contains the following Python code:


```
from google.colab import drive
drive.mount('/content/drive')
```
- Colab Interface (Right):** A screenshot of the Google Colab web interface. A dialog box asks for permission to access Google Drive files. The 'GOOGLE 드라이브에 연결' (Connect to Google Drive) button is highlighted with a red dashed box. Below the dialog, a status bar indicates 'Google 드라이브 마운트하는 중...' (Mounting Google Drive...).
- Instructions (Bottom):** A red-bordered box contains the text: '자동으로 연결되나 [아니오]를 누르면 소스로 마운트 할 수 있도록 코드를 삽입해 줌' (If it connects automatically, pressing [No] will allow you to mount the source by inserting the code).

드라이브 마운트

구글 드라이브와 서버 연결

- 버튼 '드라이브 마운트' 선택
- 마운트 기능의 스니펫 코드가 삽입

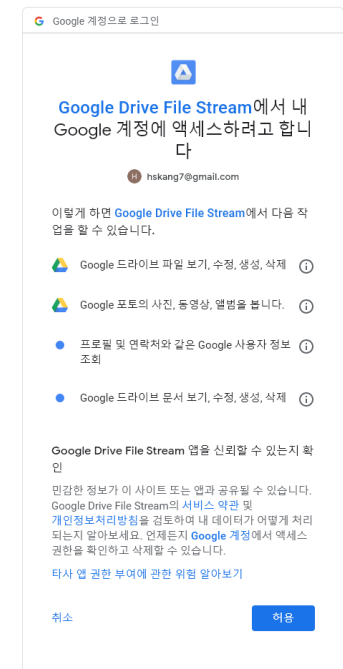
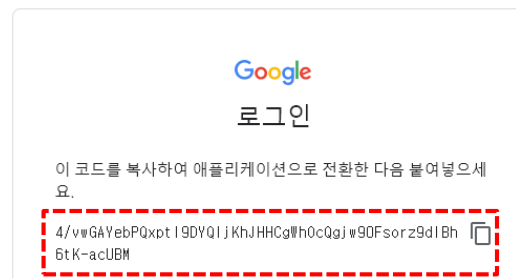
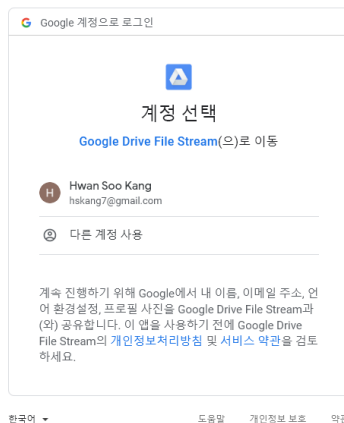


셀 실행 결과

- 인증 코드 복사

```
from google.colab import drive
drive.mount('/content/drive')
```

원하는 이름으로
수정 가능



연결된 드라이브 확인

- 셀 실행 이후
 - 복사된 인증 코드 붙여 넣고 Enter
- 메뉴 새로 고침 확인
 - 폴더 drive 자동 생성
 - 하부 구글의 My Drive 확인 가능



마운트된 구글 드라이브 활용

- 구글 드라이브의 파일을 서버에서 활용 가능
 - 구글 드라이브의 루트인 My Drive에 저장된 파이썬 소스 *.py 파일 실행 가능

The screenshot shows a Jupyter Notebook interface with a file explorer on the left and a code editor on the right.

File Explorer (Left Panel):

- Directories: .., drive, My Drive, sample_data

Code Editor (Right Panel):

```
[1] from google.colab import drive
drive.mount('/content/drive')

Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id=947318989...
Enter your authorization code:
Mounted at /content/drive

[2] pwd

'/content'

[5] cd drive/My Drive

/content/drive/My Drive

[7] !dir *.py

stringop.py

[8] cat stringop.py

print("원의 원주율 " + '3.141592')
print("python " + 'programming' + ' language')
print('파이썬 언어는' + " 강력하다")
print('파이썬 ' + "언어! " + 3 * "방가 ")

[10] !python stringop.py

원의 원주율 3.141592
python programming language
파이썬 언어는 강력하다
파이썬 언어! 방가 방가 방가
```

Colab 서버의 폴더에
깃허브를 복사 하기

서버에 깃허브 복사

• 명령어

- !git clone https://github.com/neowizard2018/neowizard.git

Colaboratory에 오신 것을 환영합니다
파일 수정 보기 삽입 런타임 도구 도움말 오후 4:58 이후 저장되지 않은 변경사항

+ 코드 + 텍스트 + 드라이브로 복사

RAM 디스크

수정 가능

목차 코드 스니펫 파일 X

업로드 새로고침 드라이브 마운트

..

drive

neowizard

DataAnalysis

data

DA_LEC_01_Example.ipynb

DA_LEC_02_Example.ipynb

DA_LEC_03_Example.ipynb

DA_LEC_04_Example.ipynb

DA_LEC_05_Example.ipynb

DA_LEC_06_Example.ipynb

MachineLearning

README.md

sample_data

[11] cd ..

/content/drive

[12] cd ..

/content

[14] !git clone https://github.com/neowizard2018/neowizard.git

Cloning into 'neowizard'...

remote: Enumerating objects: 12, done.

remote: Counting objects: 100% (12/12), done.

remote: Compressing objects: 100% (12/12), done.

remote: Total 194 (delta 2), reused 0 (delta 0), pack-reused 182

Receiving objects: 100% (194/194), 269.92 MiB | 35.97 MiB/s, done.

Resolving deltas: 100% (62/62), done.

[15] ls

drive/ neowizard/ sample_data/

[16] cd neowizard

/content/neowizard

%ls

DataAnalysis/ MachineLearning/ README.md

디스크 79.24 GB 사용 가능

마운트된 구글 드라이브 하부 깃허브 복사

• 마운트된 My Drive로 이동

– 복사 명령

- `!git clone https://github.com/hunkim/DeepLearningZeroToAll`

Colaboratory에 오신 것을 환영합니다
파일 수정 보기 삽입 런타임 도구 도움말 오후 4:58 이후 저장되지 않은 변경사항

+ 코드 + 텍스트 + 드라이브로 복사

RAM 디스크

수정 가능

목차 코드 스니펫 파일 X

업로드 새로고침 드라이브 마운트

drive

My Drive

00. 개인

01. 재정

05. 집필

2017 New C로 배우는 프로그래밍 기초

2019 가을 2개의 논문을 위한 설문

Colab Notebooks

DeepLearningZeroToAll

chainer

ipynb

lab-01-basics.ipynb

lab-02-1-linear_regression.ipynb

lab-02-2-linear_regression_fee...

lab-02-3-linear_regression_tens...

lab-03-1-minimizing_cost_sho...

lab-03-2-minimizing_cost_gradi...

lab-03-3-minimizing_cost_tf_o...

lab-03-X-minimizing_cost_tf_gr...

lab-05-1-logistic_regression.ipy...

lab-08-tensor_manipulation.ipy...

디스크 79.23 GB 사용 가능

```
[14] remote: Counting objects: 100% (12/12), done.
remote: Compressing objects: 100% (12/12), done.
remote: Total 194 (delta 2), reused 0 (delta 0), pack-reused 182
Receiving objects: 100% (194/194), 269.92 MiB | 35.97 MiB/s, done.
Resolving deltas: 100% (62/62), done.

[15] ls

drive/ neowizard/ sample_data/

[16] cd neowizard

/content/neowizard

[17] !ls

DataAnalysis/ MachineLearning/ README.md

[18] cd ../drive/My Drive

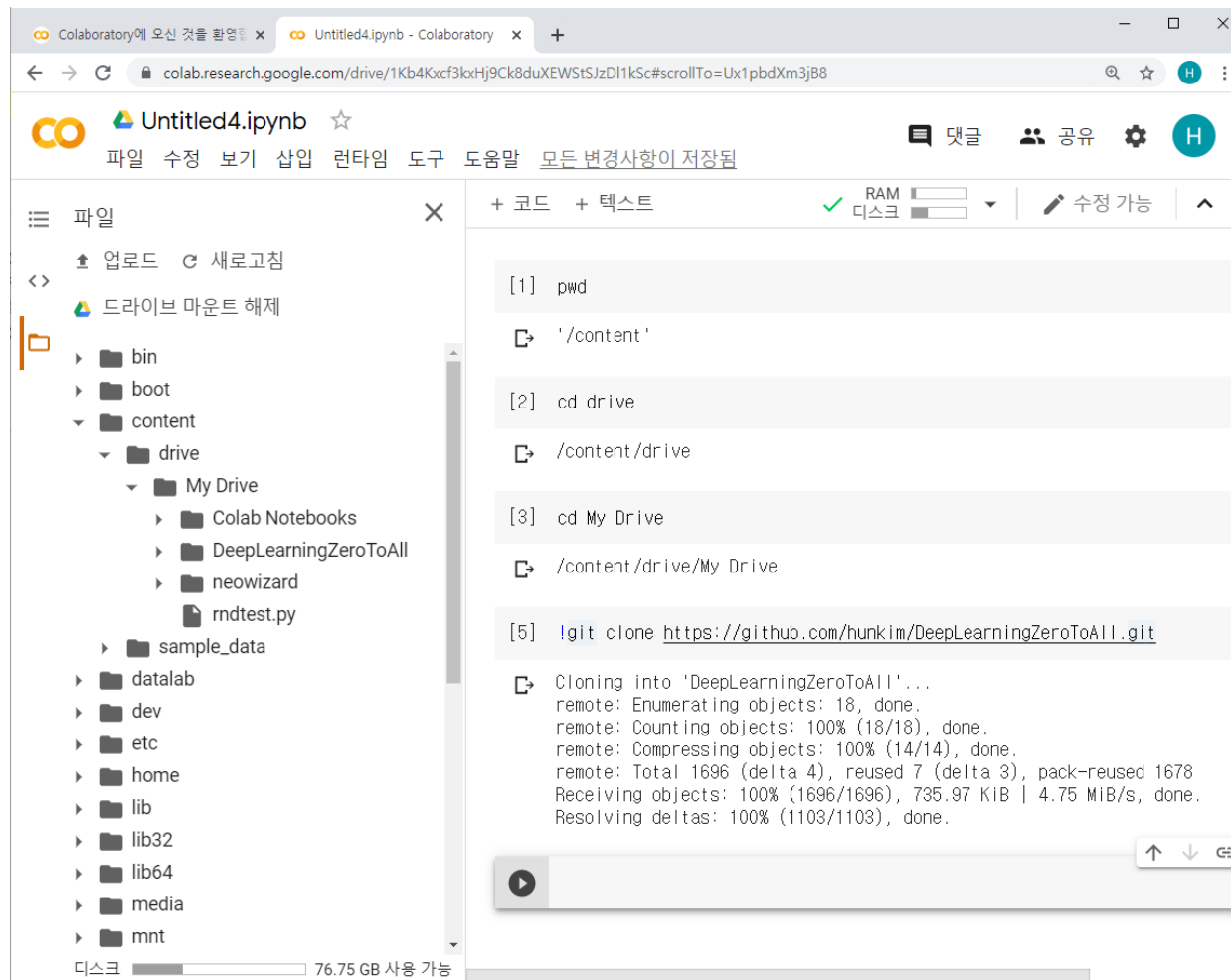
/content/drive/My Drive

[19] !git clone https://github.com/hunkim/DeepLearningZeroToAll

Cloning into 'DeepLearningZeroToAll'...
remote: Enumerating objects: 10, done.
remote: Counting objects: 100% (10/10), done.
remote: Compressing objects: 100% (8/8), done.
remote: Total 1688 (delta 2), reused 6 (delta 2), pack-reused 1678
Receiving objects: 100% (1688/1688), 733.45 KiB | 1.74 MiB/s, done.
Resolving deltas: 100% (1101/1101), done.
```

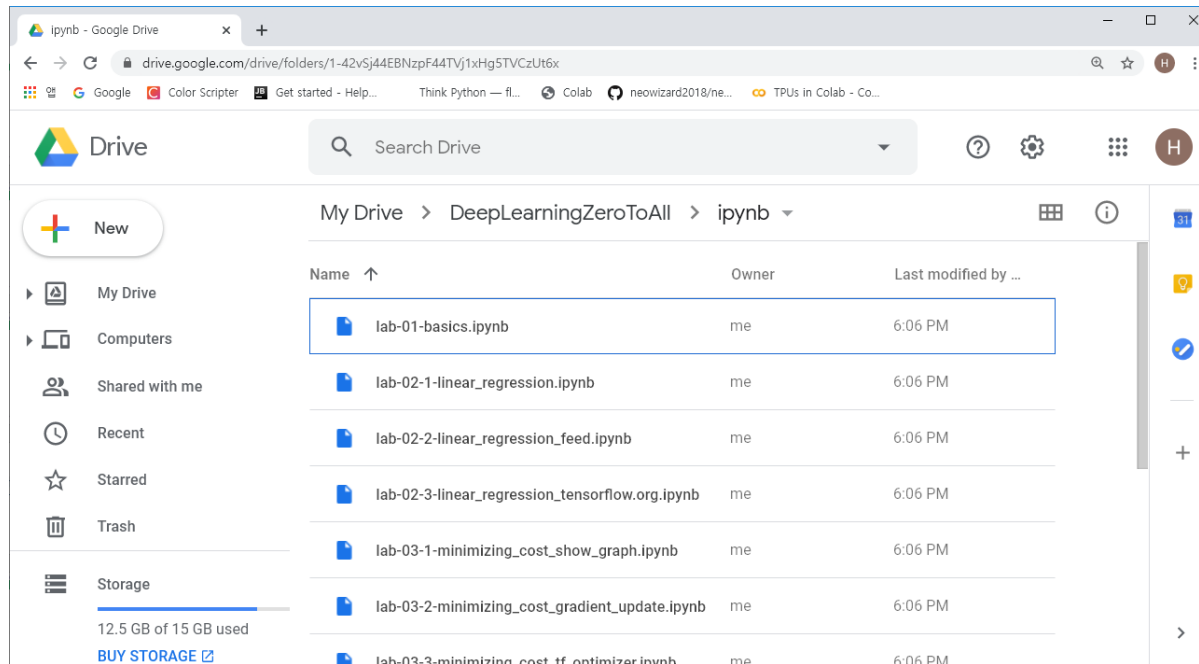

다른 샘플

- My Drive 하부에 바로 복사하는 예



구글 드라이브에서 확인

- 폴더 My Drive/DeepLearningZeroToAll/ipynb
 - 여러 파일 확인
- 파일 lab-01-basic.ipynb
 - Colab으로 열기



Colab으로 연 노트 파일

The screenshot shows a Google Colaboratory notebook interface. The browser address bar indicates the URL: `colab.research.google.com/drive/19U7Z-9irukVD-_Gh6N9VU-OHxEsKLNz`. The notebook title is "lab-01-basics.ipynb". The interface includes a file manager, a toolbar with icons for connecting, editing, and sharing, and a main content area with code cells and their outputs.

The notebook content is as follows:

Getting Started With TensorFlow

https://www.tensorflow.org/get_started/get_started

Check TF version

```
[ ] import tensorflow as tf
    tf.__version__
```

Output: `'1.0.0'`

Hello TensorFlow!

```
[ ] # Create a constant op
    # This op is added as a node to the default graph
    hello = tf.constant("Hello, TensorFlow!")

    # start a TF session
    sess = tf.Session()

    # run the op and get result
    print(sess.run(hello))
```

Output: `b'Hello, TensorFlow!'`

Tensors

```
[ ] 3 # a rank 0 tensor: this is a scalar with shape []
    [1., 2., 3.] # a rank 1 tensor: this is a vector with shape [3]
    [[1., 2., 3.], [4., 5., 6.]] # a rank 2 tensor: a matrix with shape [2, 3]
    [[[1., 2., 3.], [7., 8., 9.]]] # a rank 3 tensor with shape [2, 1, 3]
```

필요 패키지 설치

- !pip install tensorflow==1.15.0
 - Start runtime 클릭

```
colab.research.google.com/drive/15N9Kht4gizy6z-ElQPRJnZV-oLLP-#scrollTo=S6lgBwgs4aP8
lab-01-basics.ipynb
파일 수정 보기 삽입 런타임 도구 도움말 모든 변경사항이 저장됨
+ 코드 + 텍스트
RAM 디스크 수정 가능
Check TF version
[4] !import tensorflow as tf
tf.__version__
'2.2.0'
[5] !pip install tensorflow==1.15.0
Collecting tensorflow==1.15.0
  Downloading https://files.pythonhosted.org/packages/3f/98/5a99af92fb911d7a8aa0005ad5005135b4c1ba8d75fba0d726c936e6/tensorflow-1.15.0-cp36-cp36i-macosx_10_9_universal2.whl (412.3MB)
Requirement already satisfied: wrapt>=1.11.1 in /usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (1.12.1)
Requirement already satisfied: absl-py>=0.7.0 in /usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (0.9.0)
Requirement already satisfied: six>=1.10.0 in /usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (1.12.0)
Requirement already satisfied: opt-einsum>=2.3.2 in /usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (3.2.1)
Requirement already satisfied: termcolor>=1.1.0 in /usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (1.1.0)
Requirement already satisfied: wheel>=0.26 in /usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (0.34.2)
Requirement already satisfied: numpy>=2.0.1 in /usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (1.18.5)
Requirement already satisfied: grpcio>=1.8.6 in /usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (1.29.0)
Requirement already satisfied: protobuf>=3.6.1 in /usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (3.10.0)
Collecting tensorboard<1.16.0,>=1.15.0
  Downloading https://files.pythonhosted.org/packages/1e/e9/d3d747a97f7188f48aa5eda486907f3b345cd409f0a0850468ba657db246/tensorboard-1.15.0-py3-none-any.whl (3.8MB)
Requirement already satisfied: google-pasta>=0.1.6 in /usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (0.2.0)
Requirement already satisfied: keras-applications>=1.0.8 in /usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (1.0.8)
Collecting tensorflow-estimator==1.15.1
  Downloading https://files.pythonhosted.org/packages/d6/2ee9cd74c9fa21a450877847ba560b260f50fb70ee0595203082dafcc9d/tensorflow_estimator-1.15.1-py3-none-any.whl (512kB)
Requirement already satisfied: astor>=0.6.0 in /usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (0.8.1)
Collecting gast==0.2.2
  Downloading https://files.pythonhosted.org/packages/4e/35/11749b199c2d4e3cceb4d55ca22590b0d7c2c62b9de38ac4a4714687421/gast-0.2.2.tar.gz
Requirement already satisfied: keras-preprocessing>=1.0.5 in /usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (1.1.2)
Requirement already satisfied: setuptools in /usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (47.1.1)
Requirement already satisfied: werkzeug>=0.11.15 in /usr/local/lib/python3.6/dist-packages (from tensorboard<1.16.0,>=1.15.0->tensorflow==1.15.0) (3.0.1)
Requirement already satisfied: markdown>=2.6.8 in /usr/local/lib/python3.6/dist-packages (from tensorboard<1.16.0,>=1.15.0->tensorflow==1.15.0) (3.3.4)
Requirement already satisfied: h5py in /usr/local/lib/python3.6/dist-packages (from keras-applications>=1.0.8->tensorflow==1.15.0) (2.10.0)
Requirement already satisfied: importlib-metadata: python_version < "3.8" in /usr/local/lib/python3.6/dist-packages (from importlib-metadata: python_version < "3.8"->tensorflow==1.15.0) (3.10.0)
Building wheels for collected packages: gast
  Building wheel for gast (setup.py) ... done
  Created wheel for gast: filename=gast-0.2.2-cp36-none-any.whl size=7540 sha256=ea190f2b454adb0770effde16bfdbadb4b282a66a378190245242dd1c37ed587
  Stored in directory: /root/.cache/pip/wheels/5c/2e/7e/atd4d4f6cebe5c381f378ce7743a3ced3699f8eb89c6fbdadadd
Successfully built gast
ERROR: tensorflow-probability 0.10.0 has requirement gast>=0.3.2, but you'll have gast 0.2.2 which is incompatible.
Installing collected packages: tensorboard, tensorflow-estimator, gast, tensorflow
  Found existing installation: tensorboard 2.2.2
  Uninstalling tensorboard-2.2.2:
    Successfully uninstalled tensorboard-2.2.2
  Found existing installation: tensorflow-estimator 2.2.0
  Uninstalling tensorflow-estimator-2.2.0:
    Successfully uninstalled tensorflow-estimator-2.2.0
  Found existing installation: gast 0.3.3
  Uninstalling gast-0.3.3:
    Successfully uninstalled gast-0.3.3
  Found existing installation: tensorflow 2.2.0
  Uninstalling tensorflow-2.2.0:
    Successfully uninstalled tensorflow-2.2.0
Successfully installed gast-0.2.2 tensorboard-1.15.0 tensorflow-1.15.0 tensorflow-estimator-1.15.1
WARNING: The following packages were previously imported in this runtime:
[gast, tensorboard, tensorflow]
You must restart the runtime in order to use newly installed versions.
RESTART RUNTIME
```

Tensorflow 버전을 쉽게 바꾸기

- %의 파워 명령어
- 메뉴 런타임 / 런타임 다시 시작 후 다음 명령어 실행 후 ...
- %tensorflow_version 1.x
- %tensorflow_version 2.x

오류나던 소스가 실행

```

# Create a constant op
# This op is added as a node to the default graph
hello = tf.constant("Hello, TensorFlow!")

# start a TF session
sess = tf.Session()

# run the op and get result
print(sess.run(hello))

```

```

AttributeError                                Traceback (most recent call last)
<ipython-input-2-bda83f88a53e> in <module>()
      4
      5 # start a TF session
----> 6 sess = tf.Session()
      7
      8 # run the op and get result

AttributeError: module 'tensorflow' has no attribute 'Session'

```

```

# Create a constant op
# This op is added as a node to the default graph
hello = tf.constant("Hello, TensorFlow!")

# start a TF session
sess = tf.Session()

# run the op and get result
print(sess.run(hello))

```

```

b'Hello, TensorFlow!'

```

▼ Tensors

```

[] 3 # a rank 0 tensor; this is a scalar with shape []
[1., 2., 3.] # a rank 1 tensor; this is a vector with shape [3]
[[1., 2., 3.], [4., 5., 6.]] # a rank 2 tensor; a matrix with shape [2, 3]
[[[1., 2., 3.], [7., 8., 9.]]] # a rank 3 tensor with shape [2, 1, 3]
[[[1.0, 2.0, 3.0], [7.0, 8.0, 9.0]]]

```

자신의 PC의 파일을 Colab에 업로드

- 다음 스니펫 코드로 하나의 파일 업로드 가능
 - `from google.colab import files`
 - `file_uploaded = files.upload()`
- 파일 선택 대화 상자 뜸
 - 선택

The screenshot shows the Google Colab interface. At the top, a code cell contains the following Python code:

```
1 from google.colab import files
2 files.upload()
```

Below the code cell, a file selection dialog box is displayed. It has a "파일 선택" (File Selection) button, a status "선택된 파일 없음" (No file selected), and a "Cancel upload" button.

Below the dialog box, a code cell is shown with the output of the previous cell:

```
[20] 1 from google.colab import files
      2 files.upload()
```

The output shows that a file named "ch02-study.ipynb" has been uploaded. The output text includes:

- **ch02-study.ipynb**(n/a) - 105497 bytes, last modified: 2021. 5. 29. - 100% done
- Saving ch02-study.ipynb to ch02-study.ipynb
- {'ch02-study.ipynb': b'{\n "cells": [\n {\n "cell_type": "markdown",\n "metadata":

Below this, a code cell is shown with the output of the previous cell:

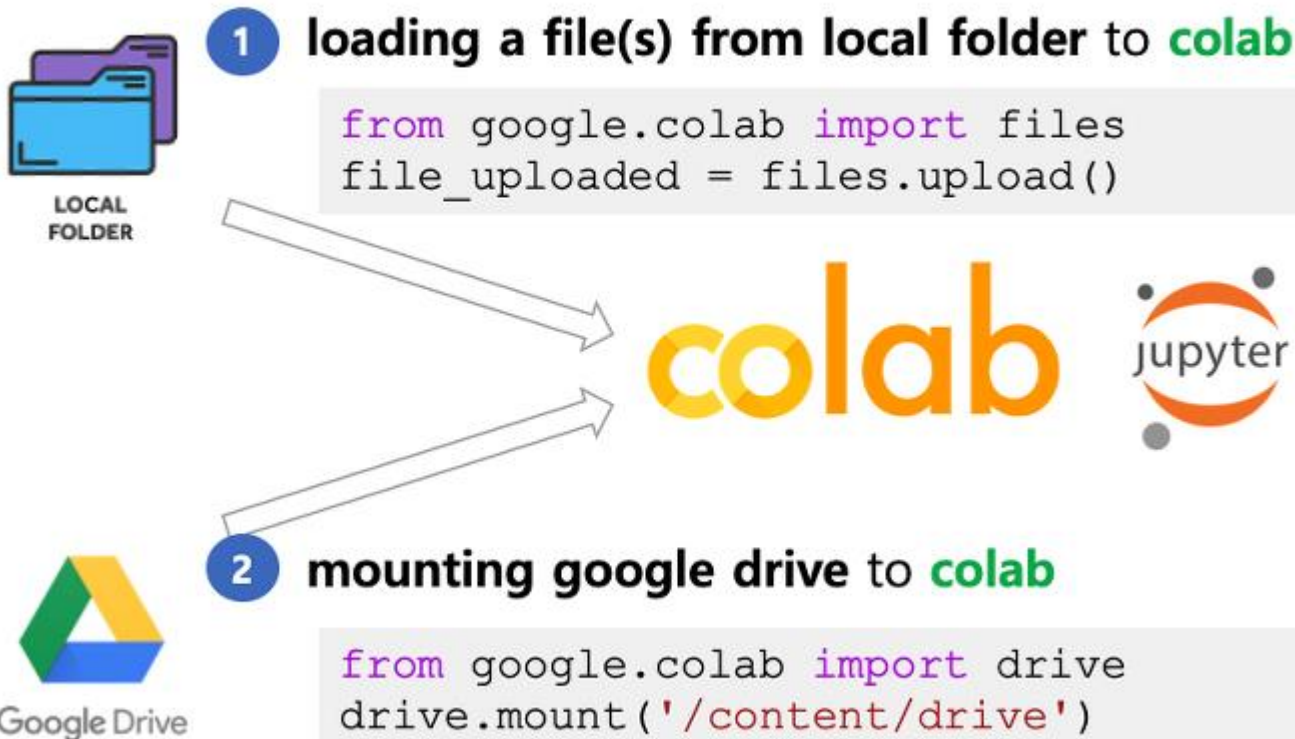
```
[21] 1 ls
```

The output shows a list of files in the current directory:

```
'텐서플로우 2.0 첫걸음'@ '2021 pandas 집필' / '제목 없는 설문지.gform'
'2020 07 tf teaching' / ch02-study.ipynb 'tensorflow 1.15' /
'2020 summer python' / Chapter8.ipynb의 사본 'tst' /
'2020 TF DNN' / 'Colab Notebooks' /
'2021 IDE & VE' / DeepLearningZeroToAll /
```

Colab으로 파일 업로드 2가지 방법

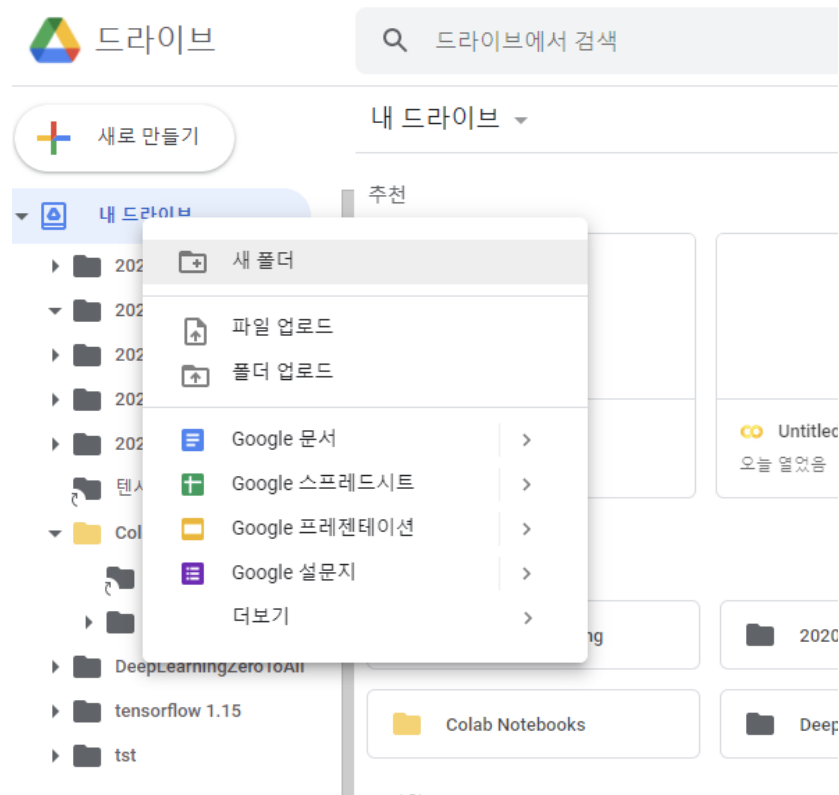
Google Colab에 데이터를 로딩하는 2가지 방법



R, Python 데이터 분석과 프로그래밍의 친구 <http://rfriend.tistory.com>

구글 드라이브에서 PC 업로드

- 오른쪽 화면에서 메뉴
 - 파일 업로드
 - 폴더 업로드



구글 Colab 서버 정보 확인

서버 정보 확인 명령

[1] `!cat /etc/issue.net`

↳ Ubuntu 18.04.3 LTS

[2] `cat /etc/issue.net`

↳ Ubuntu 18.04.3 LTS

[4] `!head /proc/cpuinfo`

```
↳ processor      : 0
   vendor_id     : GenuineIntel
   cpu family    : 6
   model         : 79
   model name    : Intel(R) Xeon(R) CPU @ 2.20GHz
   stepping      : 0
   microcode     : 0x1
   cpu MHz       : 2200.000
   cache size    : 56320 KB
   physical id   : 0
```

[6] `!head /proc/meminfo`

```
↳ MemTotal:      13335188 kB
   MemFree:      10804652 kB
   MemAvailable: 12509160 kB
   Buffers:      70536 kB
   Cached:       1790100 kB
   SwapCached:   0 kB
   Active:       692488 kB
   Inactive:     1593188 kB
   Active(anon): 402272 kB
   Inactive(anon): 312 kB
```

[8] `!df -h`

```
↳ Filesystem      Size  Used Avail Use% Mounted on
   overlay         108G   29G   75G   28% /
   tmpfs           64M    0   64M    0% /dev
   tmpfs           6.4G    0   6.4G    0% /sys/fs/cgroup
   tmpfs           6.4G   8.0K   6.4G    1% /var/colab
   /dev/sda1       114G   30G   85G   26% /etc/hosts
   shm             6.0G   4.0K   6.0G    1% /dev/shm
   tmpfs           6.4G    0   6.4G    0% /proc/acpi
   tmpfs           6.4G    0   6.4G    0% /proc/scsi
   tmpfs           6.4G    0   6.4G    0% /sys/firmware
```

[9] `pwd`

↳ `'/content'`

[10] `ls -al`

```
↳ total 16
   drwxr-xr-x 1 root root 4096 Jan 13 16:38 ./
   drwxr-xr-x 1 root root 4096 Jan 21 08:43 ../
   drwxr-xr-x 1 root root 4096 Jan 13 16:38 .config/
   drwxr-xr-x 1 root root 4096 Jan 13 16:38 sample_data/
```

[12] `!python --version`

↳ Python 3.6.9

[13] `import tensorflow as tf`
`print(tf.__version__)`

```
↳ The default version of TensorFlow in Colab will soon switch to TensorFlow 2.x.
   We recommend you upgrade now or ensure your notebook will continue to use TensorFlow 1.x via
   the %tensorflow_version 1.x magic: more info.
   1.15.0
```

Quiz

- 다음 설명 중 잘못된 것은?
- (2)
 - 코랩 서버에서 자신의 구글 드라이브를 연결할 수 있다.
 - 코랩 서버 어디든 저장된 파일이나 폴더는 나중에 접속해도 사용할 수 있다.
 - 코랩에서 프로그램 코드로 파일을 업로드할 수 있다.
 - 코랩에서 명령어로 깃허브의 저장소를 복사할 수 있다.