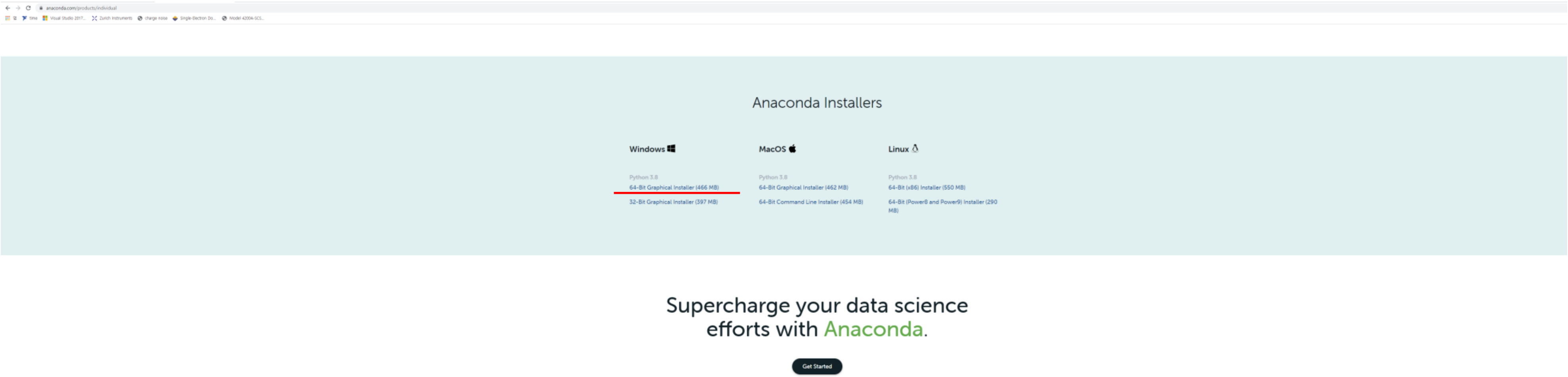
완전 처음하는 사람들을 위한 quTip 설치 및 설치 확인법

2020-8-30 서울대학교 물리천문학부 김도헌

64 bit, Window – 10 기준 quTip 설치 및 테스트 walkthrough
quTip 은 python language 기반입니다. Python 도 설치되지 않은 경우를 기준으로 설명합니다.

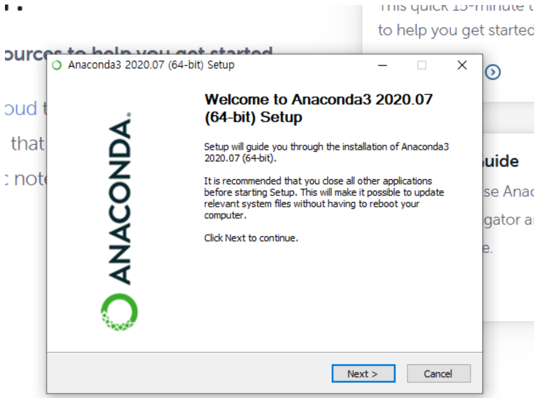
https://www.anaconda.com/products/individual
1. 위 주소에 들어가서 installer 를 다운받아 anaconda (python 을 포함하여 프로그래밍 환경 사용을 도와주는 번들-무료) 를 설치한다.



Anaconda Installers

Windows	MacOS	Linux
Python 3.8	Python 3.8	Python 3.8
64-Bit Graphical Installer (466 MB)	64-Bit Graphical Installer (462 MB)	64-Bit (x86) Installer (550 MB)
32-Bit Graphical Installer (397 MB)	64-Bit Command Line Installer (454 MB)	64-Bit (Power8 and Power9) Installer (290 MB)

Supercharge your data science efforts with Anaconda.



Anaconda3 2020.07 (64-bit) Setup

Welcome to Anaconda3 2020.07 (64-bit) Setup

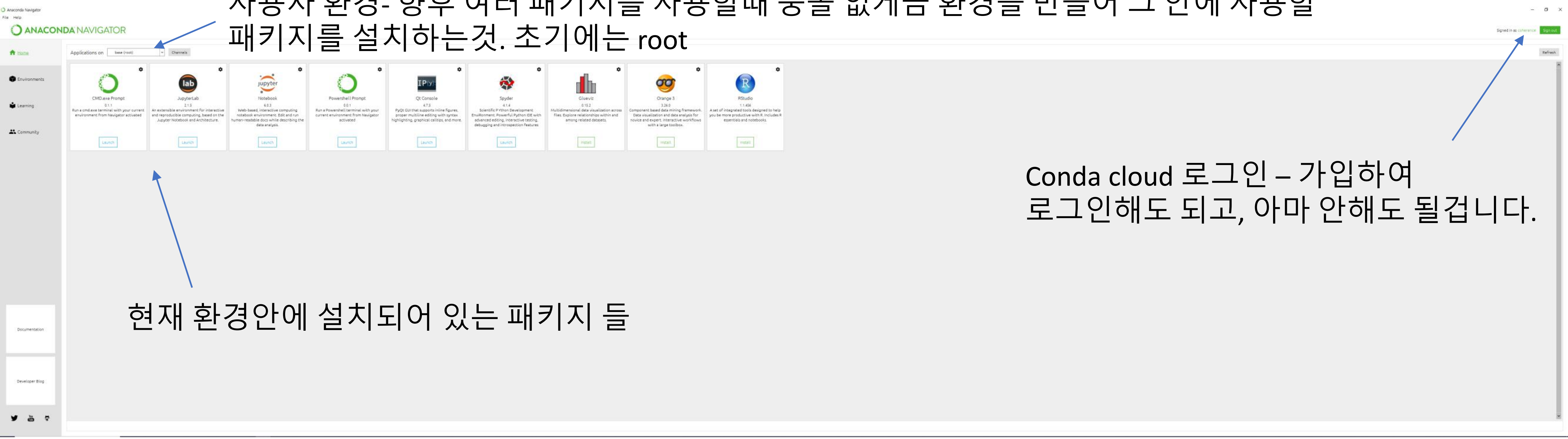
Setup will guide you through the installation of Anaconda3 2020.07 (64-bit).

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.

설치시, 추천하는 항목대로 그대로 설치.

2. Anaconda Navigator 를 실행한다. (window 검색창에서 anaconda navigator 를 검색해서 관리자 권한으로 실행추천)



Applications on: base (root)

CMD.exe Prompt 0.1.1 — Run a cmd.exe terminal with your current environment from Navigator activated

JupyterLab 2.1.5 — An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Notebook 6.0.3 — Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

Powershell Prompt 0.0.1 — Run a Powershell terminal with your current environment from Navigator activated

Qt Console 4.7.5 — PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

Spyder 4.1.4 — Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

Glueviz 0.15.2 — Multidimensional data visualization across files. Explore relationships within and among related datasets.

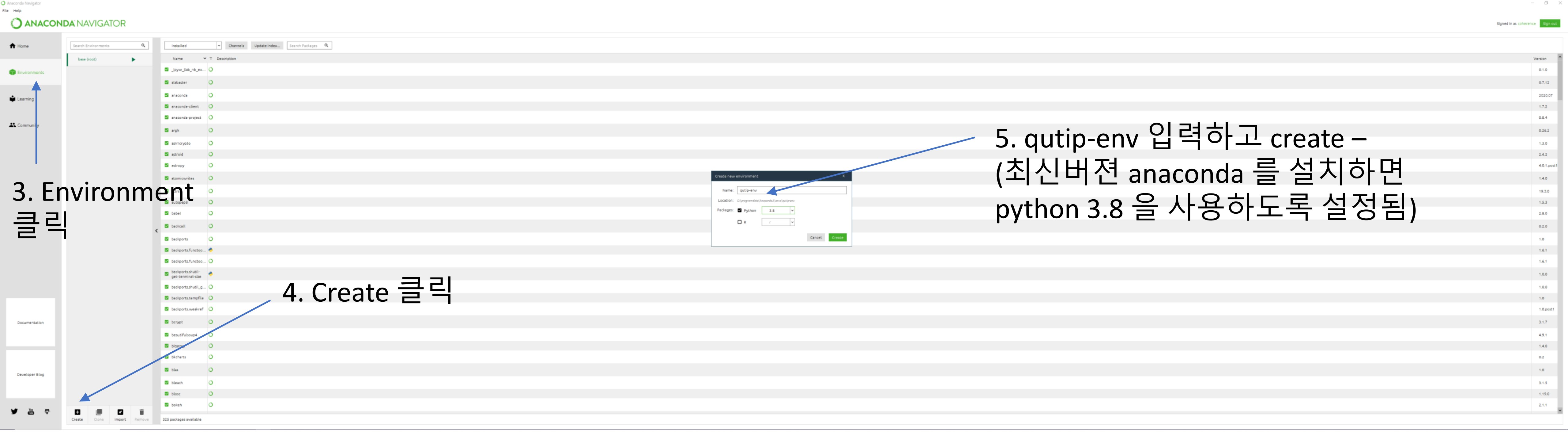
Orange 3 3.26.0 — Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.

RStudio 1.1.456 — A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.

사용자 환경- 향후 여러 패키지를 사용할때 충돌 없게끔 환경을 만들어 그 안에 사용할
패키지를 설치하는것. 초기에는 root

Conda cloud 로그인 – 가입하여
로그인해도 되고, 아마 안해도 될겁니다.

현재 환경안에 설치되어 있는 패키지 들



3. Environment 클릭

4. Create 클릭

5. qutip-env 입력하고 create – (최신버젼 anaconda 를 설치하면 python 3.8 을 사용하도록 설정됨)

Create new environment

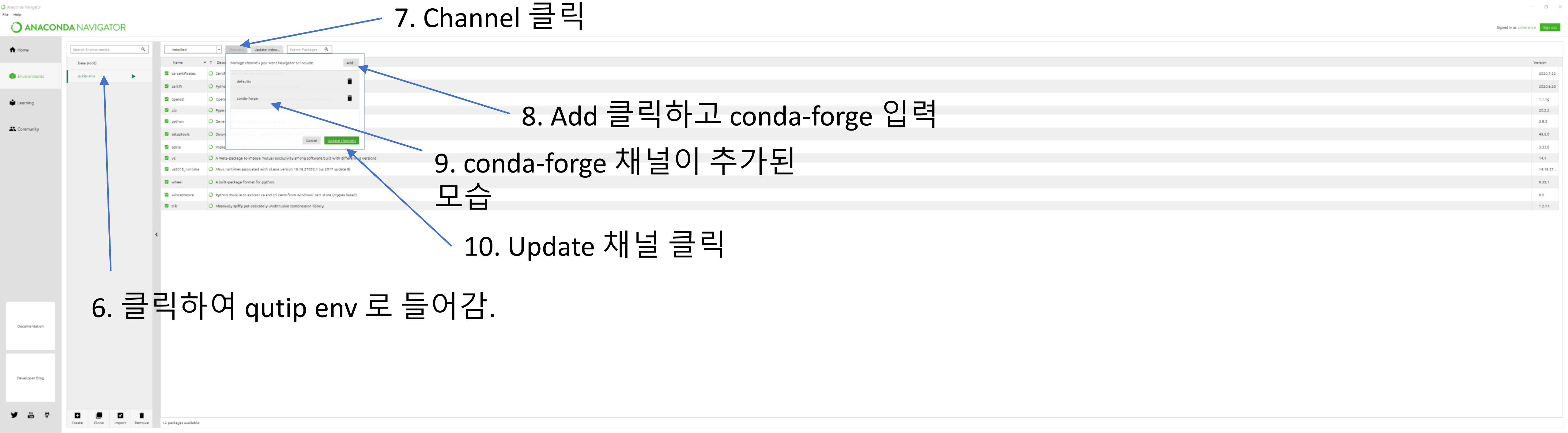
Name: qutip-env

Location: D:\programdata\Anaconda3\envs\qutip-env

Packages: ☑ Python 3.8

□ R r

Cancel Create



6. 클릭하여 qutip env 로 들어감.

7. Channel 클릭

8. Add 클릭하고 conda-forge 입력

9. conda-forge 채널이 추가된 모습

10. Update 채널 클릭

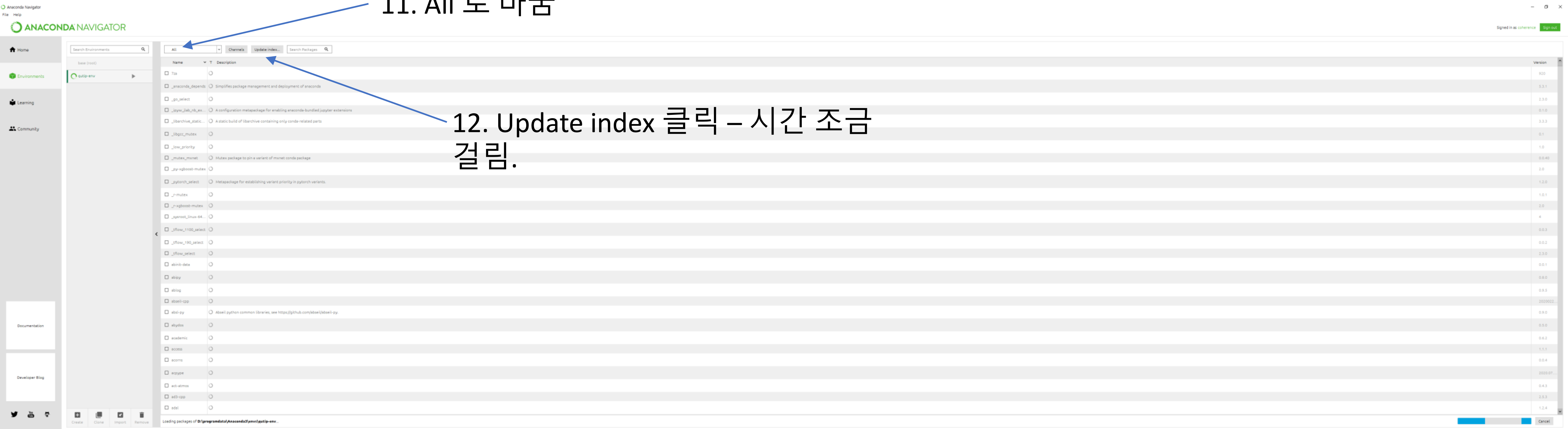
Manage channels you want Navigator to include. Add...

defaults

conda-forge

Cancel Update channels

11. All 로 바꿈

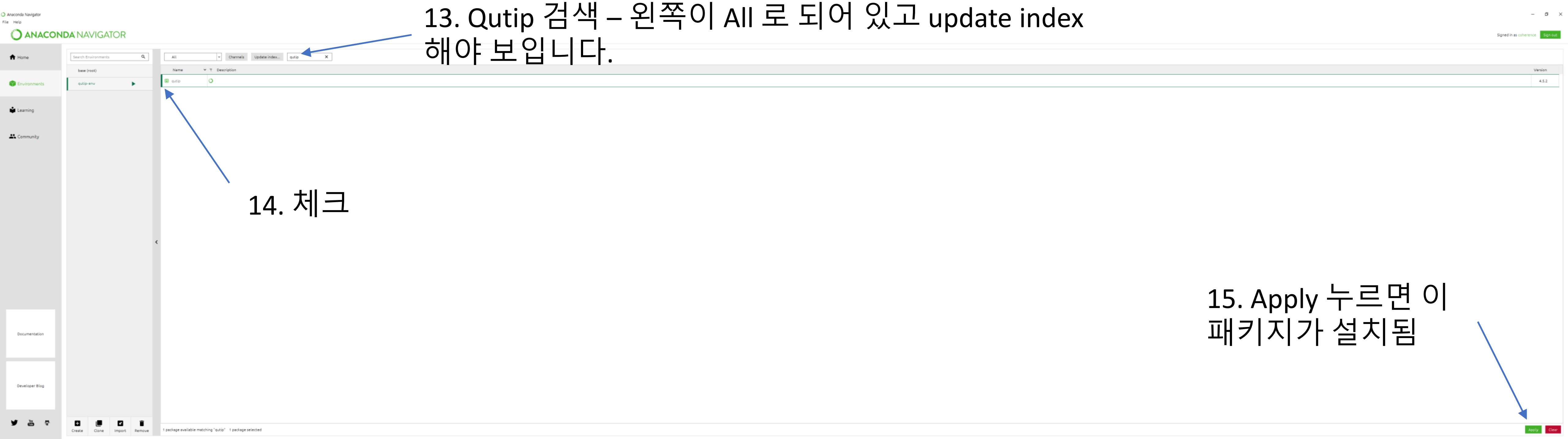


Name	T	Description	Version
□ 7za	○		920
□ _anaconda_depends	○	Simplifies package management and deployment of anaconda	5.3.1
□ _go_select	○		2.3.0
□ _ipyw_jlab_nb_ex...	○	A configuration metapackage for enabling anaconda-bundled jupyter extensions	0.1.0
□ _libarchive_static...	○	A static build of libarchive containing only conda-related parts	3.3.3
□ _libgcc_mutex	○		0.1
□ _low_priority	○		1.0
□ _mutex_mxnet	○	Mutex package to pin a variant of mxnet conda package	0.0.40
□ _py-xgboost-mutex	○		2.0
□ _pytorch_select	○	Metapackage for establishing variant priority in pytorch variants.	1.2.0
□ _r-mutex	○		1.0.1
□ _r-xgboost-mutex	○		2.0
□ _sysroot_linux-64...	○		4
□ _tflow_1100_select	○		0.0.3
□ _tflow_190_select	○		0.0.2
□ _tflow_select	○		2.3.0
□ abinit-data	○		0.0.1
□ abipy	○		0.8.0
□ ablog	○		0.9.5
□ abseil-cpp	○		2020022...
□ absl-py	○	Abseil python common libraries, see https://github.com/abseil/abseil-py.	0.9.0
□ abydos	○		0.5.0
□ academic	○		0.6.2
□ access	○		1.1.1
□ acorns	○		0.0.4
□ acpype	○		2020.07....
□ act-atmos	○		0.4.3
□ ad3-cpp	○		2.5.3
□ adal	○		1.2.4

Loading packages of **D:\programdata\Anaconda3\envs\qutip-env**...

12. Update index 클릭 – 시간 조금 걸림.

13. Qutip 검색 – 왼쪽이 All 로 되어 있고 update index 해야 보입니다.



qutip

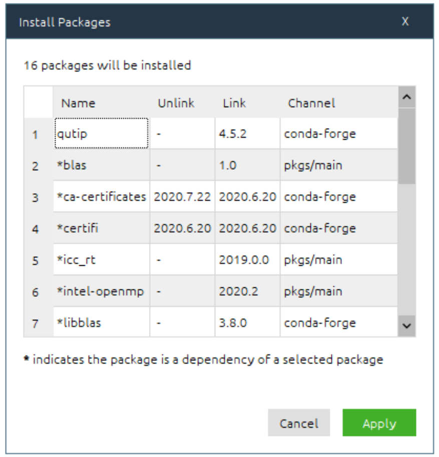
Name	T	Description	Version
☑ qutip	○		4.5.2

1 package available matching "qutip"   1 package selected

14. 체크

15. Apply 누르면 이 패키지가 설치됨

16. 설치시 나오는 화면 Apply 눌러서 설치

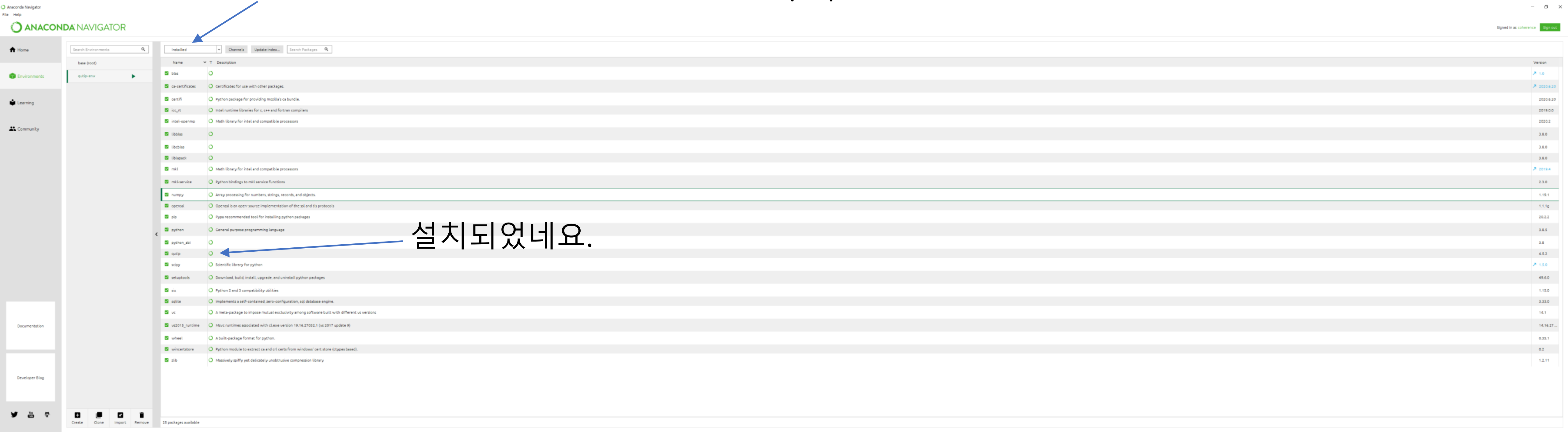


16 packages will be installed

	Name	Unlink	Link	Channel
1	qutip	-	4.5.2	conda-forge
2	*blas	-	1.0	pkgs/main
3	*ca-certificates	2020.7.22	2020.6.20	conda-forge
4	*certifi	2020.6.20	2020.6.20	conda-forge
5	*icc_rt	-	2019.0.0	pkgs/main
6	*intel-openmp	-	2020.2	pkgs/main
7	*libblas	-	3.8.0	conda-forge

* indicates the package is a dependency of a selected package

17. 설치 완료후 installed 로 바꿔서 현재 qutip-env 안에 있는 설치된 패키지 확인

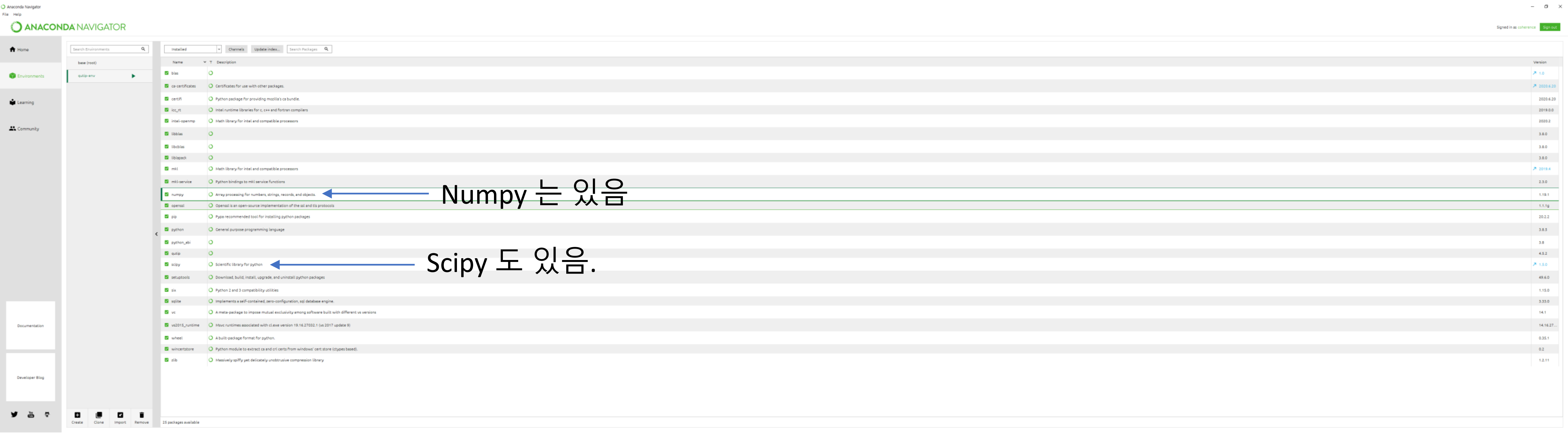


Name	T	Description	Version
☑ blas	○		↗ 1.0
☑ ca-certificates	○	Certificates for use with other packages.	↗ 2020.6.20
☑ certifi	○	Python package for providing mozilla's ca bundle.	2020.6.20
☑ icc_rt	○	Intel runtime libraries for c, c++ and fortran compilers	2019.0.0
☑ intel-openmp	○	Math library for intel and compatible processors	2020.2
☑ libblas	○		3.8.0
☑ libcblas	○		3.8.0
☑ liblapack	○		3.8.0
☑ mkl	○	Math library for intel and compatible processors	↗ 2019.4
☑ mkl-service	○	Python bindings to mkl service functions	2.3.0
☑ numpy	○	Array processing for numbers, strings, records, and objects.	1.19.1
☑ openssl	○	Openssl is an open-source implementation of the ssl and tls protocols	1.1.1g
☑ pip	○	Pypa recommended tool for installing python packages	20.2.2
☑ python	○	General purpose programming language	3.8.5
☑ python_abi	○		3.8
☑ qutip	○		4.5.2
☑ scipy	○	Scientific library for python	↗ 1.5.0
☑ setuptools	○	Download, build, install, upgrade, and uninstall python packages	49.6.0
☑ six	○	Python 2 and 3 compatibility utilities	1.15.0
☑ sqlite	○	Implements a self-contained, zero-configuration, sql database engine.	3.33.0
☑ vc	○	A meta-package to impose mutual exclusivity among software built with different vs versions	14.1
☑ vs2015_runtime	○	Msvc runtimes associated with cl.exe version 19.16.27032.1 (vs 2017 update 9)	14.16.27...
☑ wheel	○	A built-package format for python.	0.35.1
☑ wincertstore	○	Python module to extract ca and crl certs from windows' cert store (ctypes based).	0.2
☑ zlib	○	Massively spiffy yet delicately unobtrusive compression library	1.2.11

25 packages available

설치되었네요.

추가로 필요한 패키지들 설치: numpy scipy cython matplotlib pytest pytest-cov jupyter notebook spyder 들이 필요한데…



base (root)

qutip-env

Installed

Name	Description	Version
☑ blas	○	↗ 1.0
☑ ca-certificates	○ Certificates for use with other packages.	↗ 2020.6.20
☑ certifi	○ Python package for providing mozilla's ca bundle.	2020.6.20
☑ icc_rt	○ Intel runtime libraries for c, c++ and fortran compilers	2019.0.0
☑ intel-openmp	○ Math library for intel and compatible processors	2020.2
☑ libblas	○	3.8.0
☑ libcblas	○	3.8.0
☑ liblapack	○	3.8.0
☑ mkl	○ Math library for intel and compatible processors	↗ 2019.4
☑ mkl-service	○ Python bindings to mkl service functions	2.3.0
☑ numpy	○ Array processing for numbers, strings, records, and objects.	1.19.1
☑ openssl	○ Openssl is an open-source implementation of the ssl and tls protocols	1.1.1g
☑ pip	○ Pypa recommended tool for installing python packages	20.2.2
☑ python	○ General purpose programming language	3.8.5
☑ python_abi	○	3.8
☑ qutip	○	4.5.2
☑ scipy	○ Scientific library for python	↗ 1.5.0
☑ setuptools	○ Download, build, install, upgrade, and uninstall python packages	49.6.0
☑ six	○ Python 2 and 3 compatibility utilities	1.15.0
☑ sqlite	○ Implements a self-contained, zero-configuration, sql database engine.	3.33.0
☑ vc	○ A meta-package to impose mutual exclusivity among software built with different vs versions	14.1
☑ vs2015_runtime	○ Msvc runtimes associated with cl.exe version 19.16.27032.1 (vs 2017 update 9)	14.16.27…
☑ wheel	○ A built-package format for python.	0.35.1
☑ wincertstore	○ Python module to extract ca and crl certs from windows' cert store (ctypes based).	0.2
☑ zlib	○ Massively spiffy yet delicately unobtrusive compression library	1.2.11

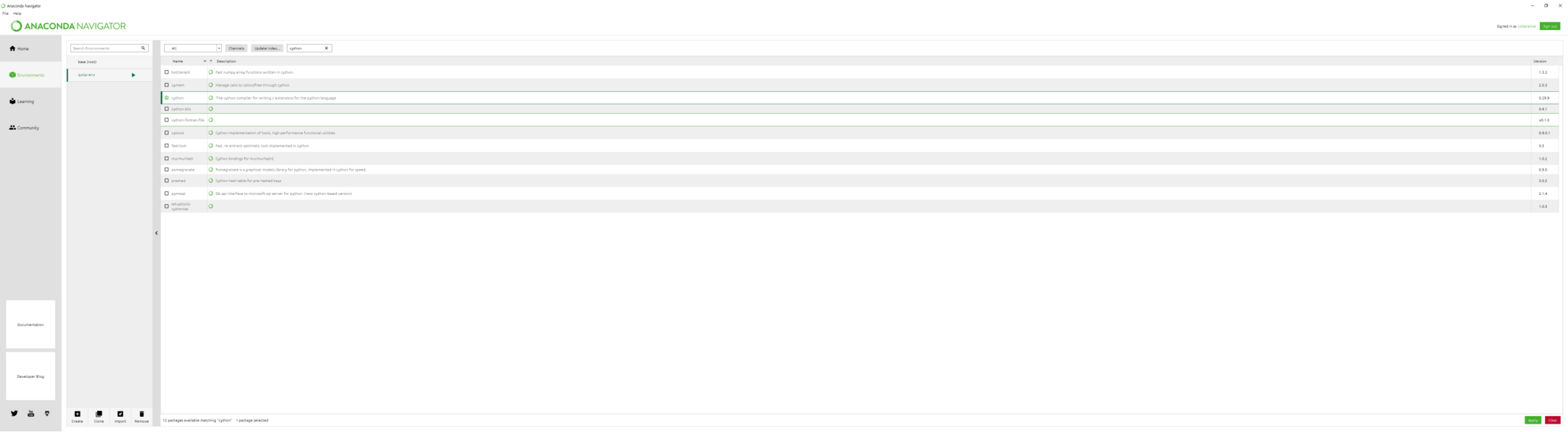
25 packages available

Numpy 는 있음

Scipy 도 있음.

나머지도 위의 방법으로 검색해서 설치 (All) 로 바꾼다음 검색해봄.

예를 들어 cython



Name	T	Description	Version
☐ bottleneck	○	Fast numpy array functions written in cython.	1.3.2
☐ cymem	○	Manage calls to calloc/free through cython	2.0.3
⊞ cython	○	The cython compiler for writing c extensions for the python language	0.29.9
☐ cython-blis	○		0.4.1
☐ cython-fortran-file	○		v0.1.0
☐ cytoolz	○	Cython implementation of toolz. high performance functional utilities	0.9.0.1
☐ fastrlock	○	Fast, re-entrant optimistic lock implemented in cython	0.5
☐ murmurhash	○	Cython bindings for murmurhash2	1.0.2
☐ pomegranate	○	Pomegranate is a graphical models library for python, implemented in cython for speed.	0.9.0
☐ preshed	○	Cython hash table for pre-hashed keys	3.0.2
☐ pymssql	○	Db-api interface to microsoft sql server for python. (new cython-based version)	2.1.4
☐ setuptools-cythonize	○		1.0.5

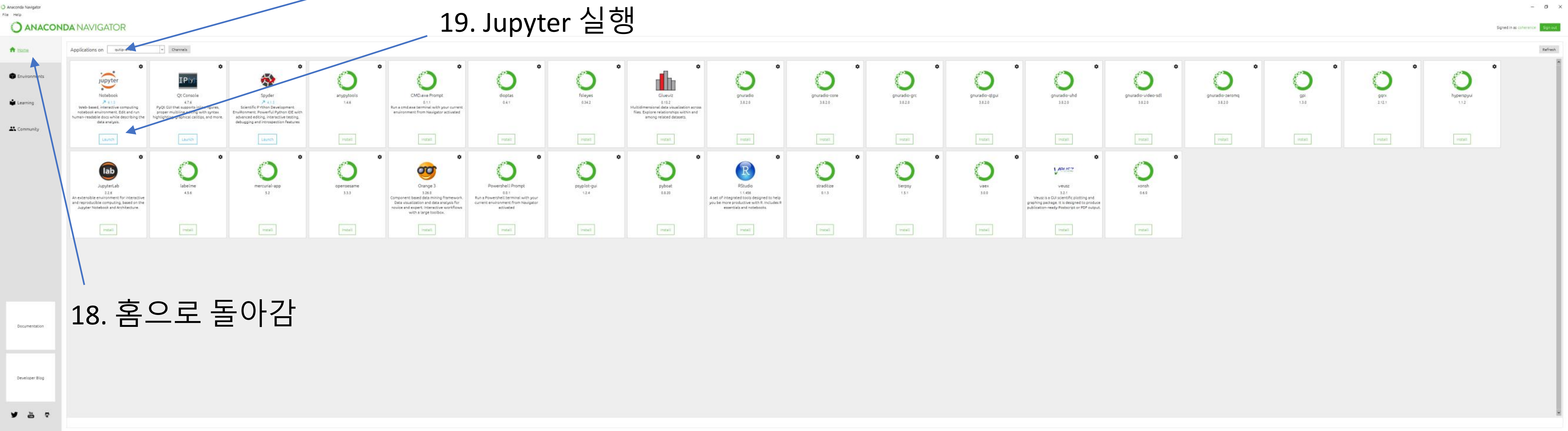
12 packages available matching "cython"   1 package selected

나머지 matplotlib pytest pytest-cov jupyter notebook spyder 도 각각 검색하여 설치

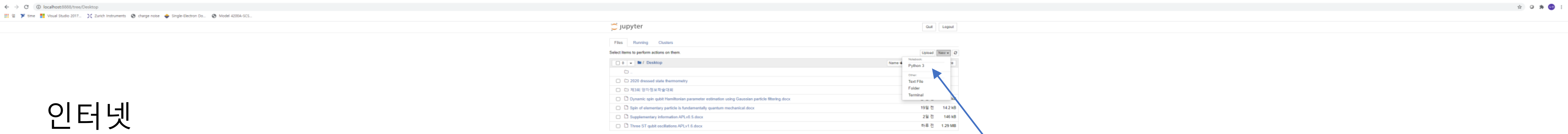
잘 설치되었는지 테스트.

현재 환경이 qutip-env 임을 확인

19. Jupyter 실행



18. 홈으로 돌아감



Select items to perform actions on them.

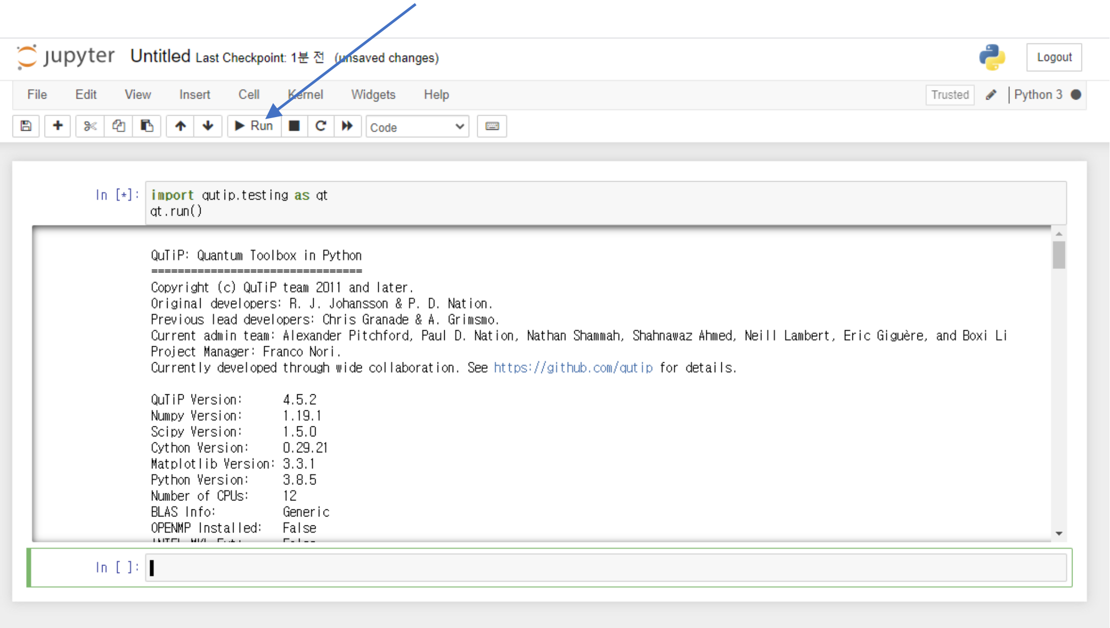
Desktop

- ..
- 2020 dressed state thermometry
- 제3회 양자정보학술대회
- Dynamic spin qubit Hamiltonian parameter estimation using Gaussian particle filtering.docx
- Spin of elementary particle is fundamentally quantum mechanical.docx 19일 전 14.2 kB
- Supplementary information APLv0.5.docx 2일 전 146 kB
- Three ST qubit oscillations APLv1.6.docx 하루 전 1.29 MB

인터넷
브라우져에서
jupyter 열림.

아무폴더에서나
(저는 desktop)
python 3 로 새로운
파일 생성

20. import qutip.testing as qt
qt.run() 을 입력하고 Run 누름



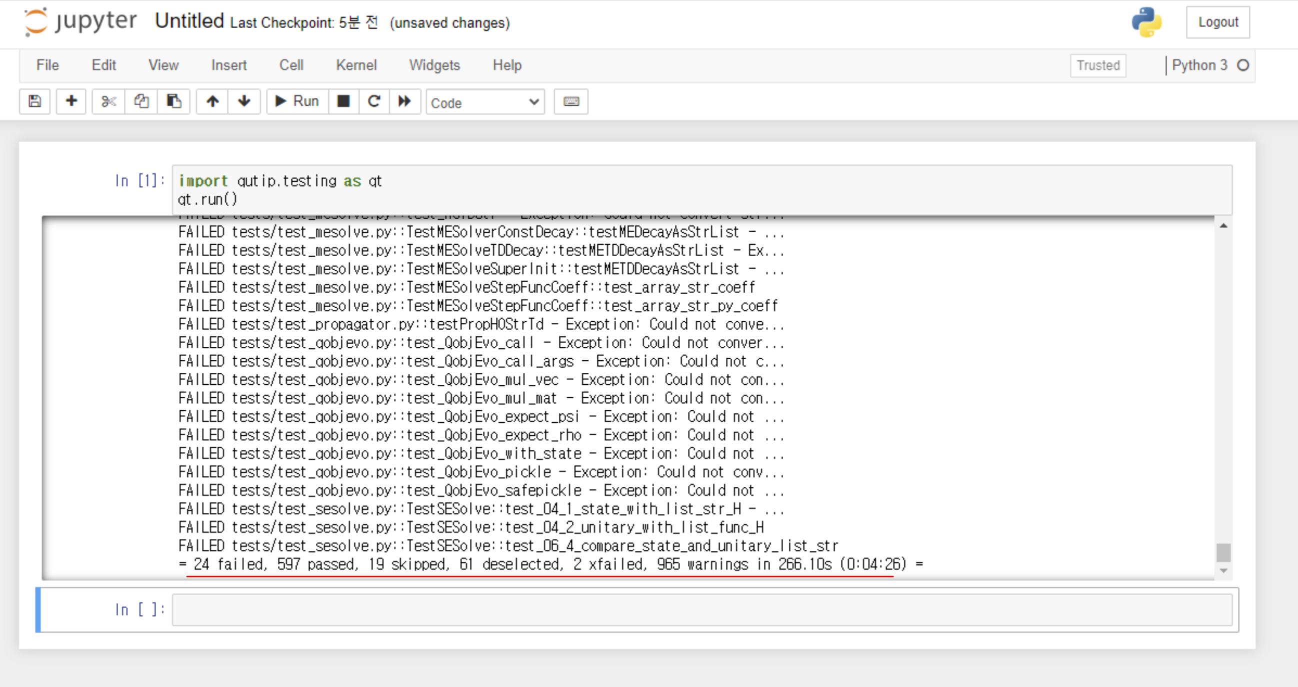
jupyter Untitled Last Checkpoint: 1분 전 (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code

```
In [*]: import qutip.testing as qt
        qt.run()

        QuTiP: Quantum Toolbox in Python
        ================================
        Copyright (c) QuTiP team 2011 and later.
        Original developers: R. J. Johansson & P. D. Nation.
        Previous lead developers: Chris Granade & A. Grimsmo.
        Current admin team: Alexander Pitchford, Paul D. Nation, Nathan Shammah, Shahnawaz Ahmed, Neill Lambert, Eric Giguère, and Boxi Li
        Project Manager: Franco Nori.
        Currently developed through wide collaboration. See https://github.com/qutip for details.

        QuTiP Version:      4.5.2
        Numpy Version:      1.19.1
        Scipy Version:      1.5.0
        Cython Version:     0.29.21
        Matplotlib Version: 3.3.1
        Python Version:     3.8.5
        Number of CPUs:     12
        BLAS Info:          Generic
        OPENMP Installed:   False

In [ ]:
```



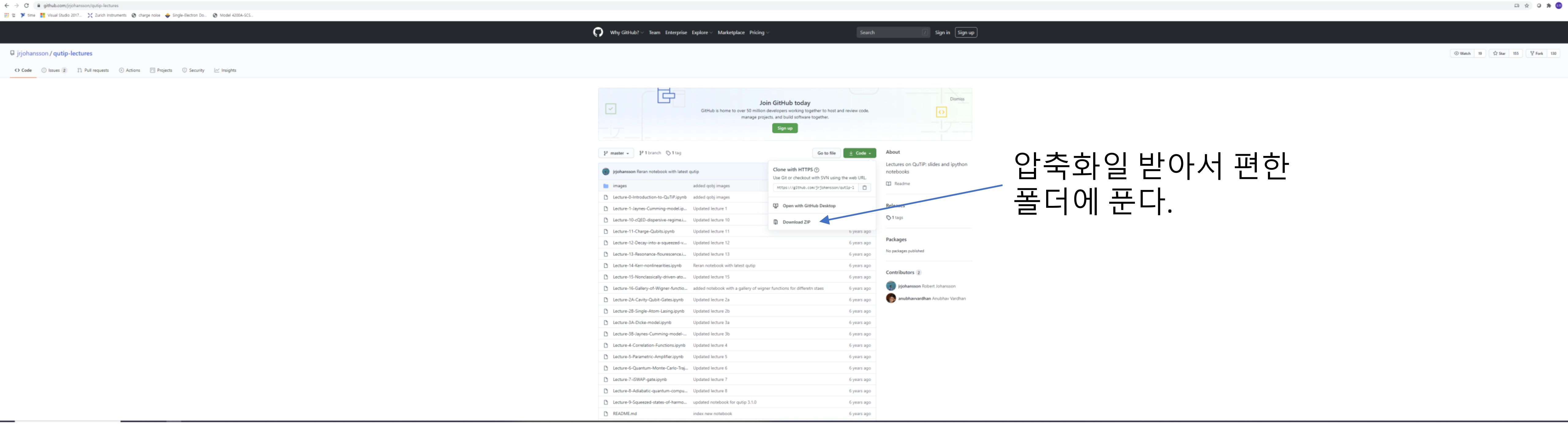
jupyter Untitled Last Checkpoint: 5분 전 (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code

```
In [1]: import qutip.testing as qt
        qt.run()
        FAILED tests/test_mesolve.py::TestMESolverConstDecay::testMEDecayAsStrList - ...
        FAILED tests/test_mesolve.py::TestMESolveTDDecay::testMETDDecayAsStrList - Ex...
        FAILED tests/test_mesolve.py::TestMESolveSuperInit::testMETDDecayAsStrList - ...
        FAILED tests/test_mesolve.py::TestMESolveStepFuncCoeff::test_array_str_coeff
        FAILED tests/test_mesolve.py::TestMESolveStepFuncCoeff::test_array_str_py_coeff
        FAILED tests/test_propagator.py::testPropHOStrTd - Exception: Could not conve...
        FAILED tests/test_qobjevo.py::test_QobjEvo_call - Exception: Could not conver...
        FAILED tests/test_qobjevo.py::test_QobjEvo_call_args - Exception: Could not c...
        FAILED tests/test_qobjevo.py::test_QobjEvo_mul_vec - Exception: Could not con...
        FAILED tests/test_qobjevo.py::test_QobjEvo_mul_mat - Exception: Could not con...
        FAILED tests/test_qobjevo.py::test_QobjEvo_expect_psi - Exception: Could not ...
        FAILED tests/test_qobjevo.py::test_QobjEvo_expect_rho - Exception: Could not ...
        FAILED tests/test_qobjevo.py::test_QobjEvo_with_state - Exception: Could not ...
        FAILED tests/test_qobjevo.py::test_QobjEvo_pickle - Exception: Could not conv...
        FAILED tests/test_qobjevo.py::test_QobjEvo_safepickle - Exception: Could not ...
        FAILED tests/test_sesolve.py::TestSESolve::test_04_1_state_with_list_str_H - ...
        FAILED tests/test_sesolve.py::TestSESolve::test_04_2_unitary_with_list_func_H
        FAILED tests/test_sesolve.py::TestSESolve::test_06_4_compare_state_and_unitary_list_str
        = 24 failed, 597 passed, 19 skipped, 61 deselected, 2 xfailed, 965 warnings in 266.10s (0:04:26) =

In [ ]:
```

21. 테스트 시간이 좀 걸림. 이와 같이 몇 개의 기능은 Fail 이 뜰수도 있습니다.

22. 더 정확한 테스트를 위해, https://github.com/jrjohansson/qutip-lectures 에 들어감



Join GitHub today

GitHub is home to over 50 million developers working together to host and review code, manage projects, and build software together.

Sign up

Clone with HTTPS

Use Git or checkout with SVN using the web URL.

`https://github.com/jrjohansson/qutip-l`

Open with GitHub Desktop

Download ZIP

jrjohansson Reran notebook with latest qutip

images	added qobj images	
Lecture-0-Introduction-to-QuTiP.ipynb	added qobj images	
Lecture-1-Jaynes-Cumming-model.ip...	Updated lecture 1	
Lecture-10-cQED-dispersive-regime.i...	Updated lecture 10	
Lecture-11-Charge-Qubits.ipynb	Updated lecture 11	6 years ago
Lecture-12-Decay-into-a-squeezed-v...	Updated lecture 12	6 years ago
Lecture-13-Resonance-flourescence.i...	Updated lecture 13	6 years ago
Lecture-14-Kerr-nonlinearities.ipynb	Reran notebook with latest qutip	6 years ago
Lecture-15-Nonclassically-driven-ato...	Updated lecture 15	6 years ago
Lecture-16-Gallery-of-Wigner-functio...	added notebook with a gallery of wigner functions for differetn staes	6 years ago
Lecture-2A-Cavity-Qubit-Gates.ipynb	Updated lecture 2a	6 years ago
Lecture-2B-Single-Atom-Lasing.ipynb	Updated lecture 2b	6 years ago
Lecture-3A-Dicke-model.ipynb	Updated lecture 3a	6 years ago
Lecture-3B-Jaynes-Cumming-model-...	Updated lecture 3b	6 years ago
Lecture-4-Correlation-Functions.ipynb	Updated lecture 4	6 years ago
Lecture-5-Parametric-Amplifier.ipynb	Updated lecture 5	6 years ago
Lecture-6-Quantum-Monte-Carlo-Traj...	Updated lecture 6	6 years ago
Lecture-7-iSWAP-gate.ipynb	Updated lecture 7	6 years ago
Lecture-8-Adiabatic-quantum-compu...	Updated lecture 8	6 years ago
Lecture-9-Squeezed-states-of-harmo...	updated notebook for qutip 3.1.0	6 years ago
README.md	index new notebook	6 years ago

About

Lectures on QuTiP: slides and ipython notebooks

Readme

Releases

1 tags

Packages

No packages published

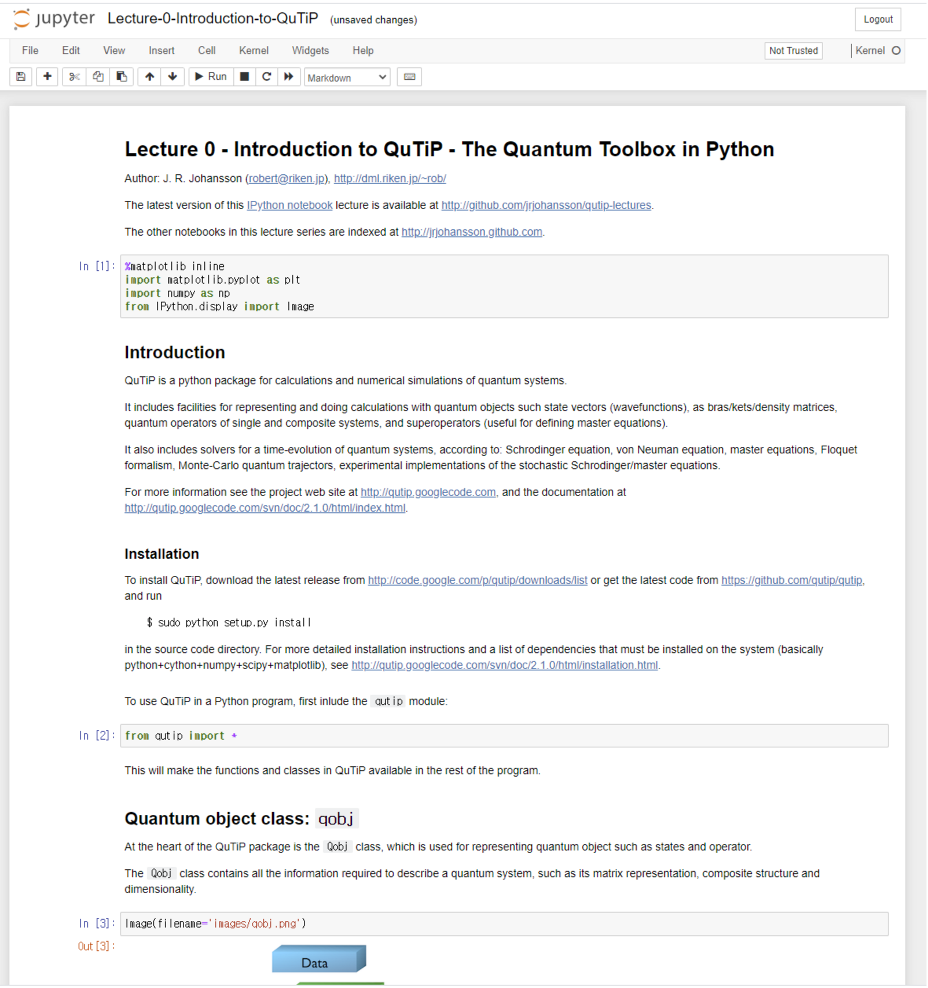
Contributors 2

jrjohansson Robert Johansson

anubhavvardhan Anubhav Vardhan

압축화일 받아서 편한
폴더에 푼다.

Jupyter 를 사용하여 lecture 0 파일을 연다.



Lecture 0 - Introduction to QuTiP - The Quantum Toolbox in Python

Author: J. R. Johansson (robert@riken.jp), http://dml.riken.jp/~rob/

The latest version of this IPython notebook lecture is available at http://github.com/jrjohansson/qutip-lectures.

The other notebooks in this lecture series are indexed at http://jrjohansson.github.com.

```
In [1]: %matplotlib inline
        import matplotlib.pyplot as plt
        import numpy as np
        from IPython.display import Image
```

Introduction

QuTiP is a python package for calculations and numerical simulations of quantum systems.

It includes facilities for representing and doing calculations with quantum objects such state vectors (wavefunctions), as bras/kets/density matrices, quantum operators of single and composite systems, and superoperators (useful for defining master equations).

It also includes solvers for a time-evolution of quantum systems, according to: Schrodinger equation, von Neuman equation, master equations, Floquet formalism, Monte-Carlo quantum trajectors, experimental implementations of the stochastic Schrodinger/master equations.

For more information see the project web site at http://qutip.googlecode.com, and the documentation at http://qutip.googlecode.com/svn/doc/2.1.0/html/index.html.

Installation

To install QuTiP, download the latest release from http://code.google.com/p/qutip/downloads/list or get the latest code from https://github.com/qutip/qutip, and run

```
$ sudo python setup.py install
```

in the source code directory. For more detailed installation instructions and a list of dependencies that must be installed on the system (basically python+cython+numpy+scipy+matplotlib), see http://qutip.googlecode.com/svn/doc/2.1.0/html/installation.html.

To use QuTiP in a Python program, first inlude the `qutip` module:

```
In [2]: from qutip import *
```

This will make the functions and classes in QuTiP available in the rest of the program.

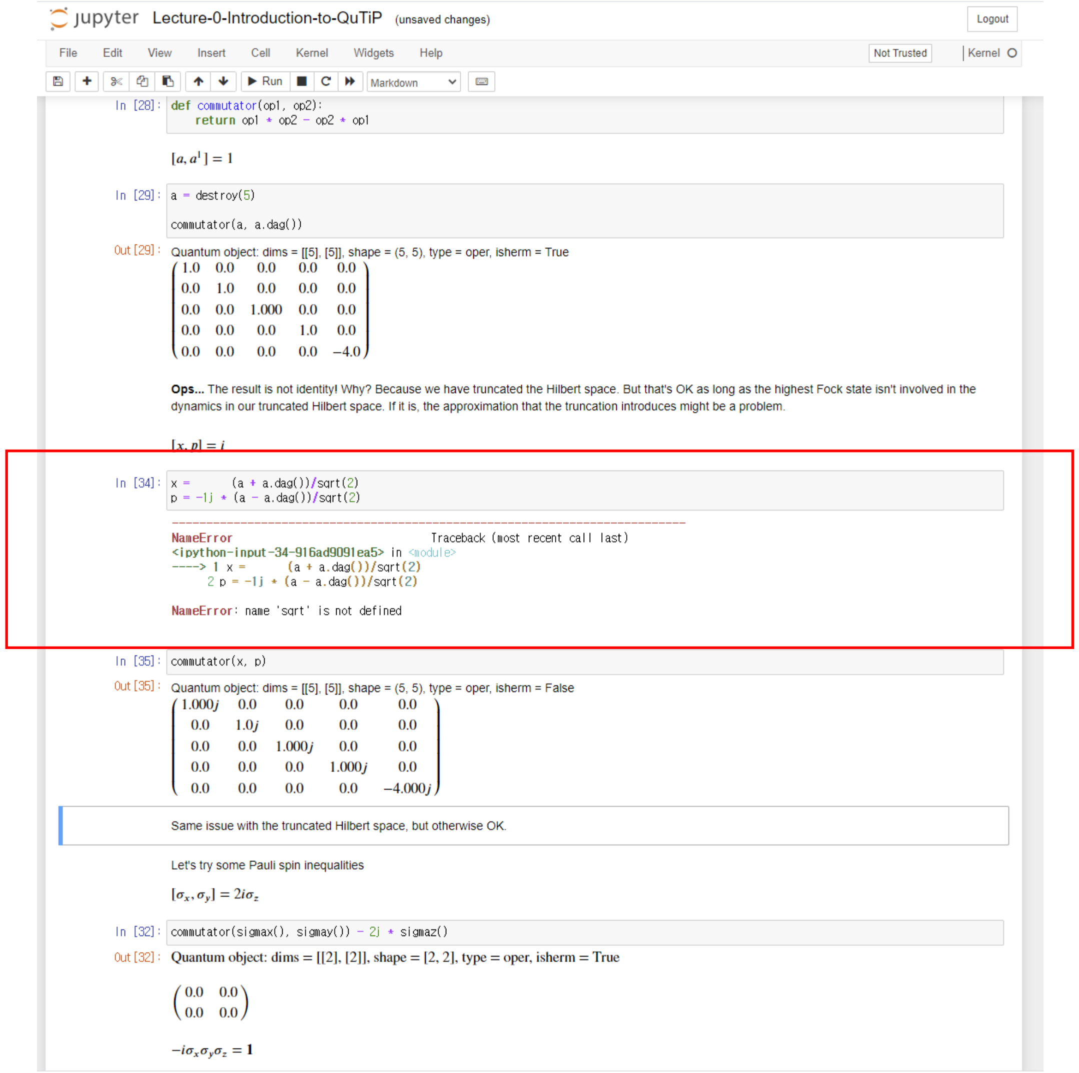
Quantum object class: `qobj`

At the heart of the QuTiP package is the `Qobj` class, which is used for representing quantum object such as states and operator.

The `Qobj` class contains all the information required to describe a quantum system, such as its matrix representation, composite structure and dimensionality.

```
In [3]: Image(filename='images/qobj.png')
Out[3]:
```

이제 Run 을 한번씩 눌러서 한줄한줄 실행해본다.
(철째줄에서 run 누르면 그다음 줄로 자동으로 넘어감. Run 버튼만 계속 누르면 되요)



```
In [28]: def commutator(op1, op2):
             return op1 * op2 - op2 * op1
```

$[a, a^\dagger] = 1$

```
In [29]: a = destroy(5)

         commutator(a, a.dag())
```

Out [29]: Quantum object: dims = [[5], [5]], shape = (5, 5), type = oper, isherm = True

$$\begin{pmatrix} 1.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.000 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & -4.0 \end{pmatrix}$$

Ops... The result is not identity! Why? Because we have truncated the Hilbert space. But that's OK as long as the highest Fock state isn't involved in the dynamics in our truncated Hilbert space. If it is, the approximation that the truncation introduces might be a problem.

$[x, p] = i$

```
In [34]: x =       (a + a.dag())/sqrt(2)
         p = -1j * (a - a.dag())/sqrt(2)

---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-34-916ad9091ea5> in <module>
----> 1 x =       (a + a.dag())/sqrt(2)
      2 p = -1j * (a - a.dag())/sqrt(2)

NameError: name 'sqrt' is not defined
```

```
In [35]: commutator(x, p)
```

Out [35]: Quantum object: dims = [[5], [5]], shape = (5, 5), type = oper, isherm = False

$$\begin{pmatrix} 1.000j & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.0j & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.000j & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 1.000j & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & -4.000j \end{pmatrix}$$

Same issue with the truncated Hilbert space, but otherwise OK.

Let's try some Pauli spin inequalities

$[\sigma_x, \sigma_y] = 2i\sigma_z$

```
In [32]: commutator(sigmax(), sigmay()) - 2j * sigmaz()
```

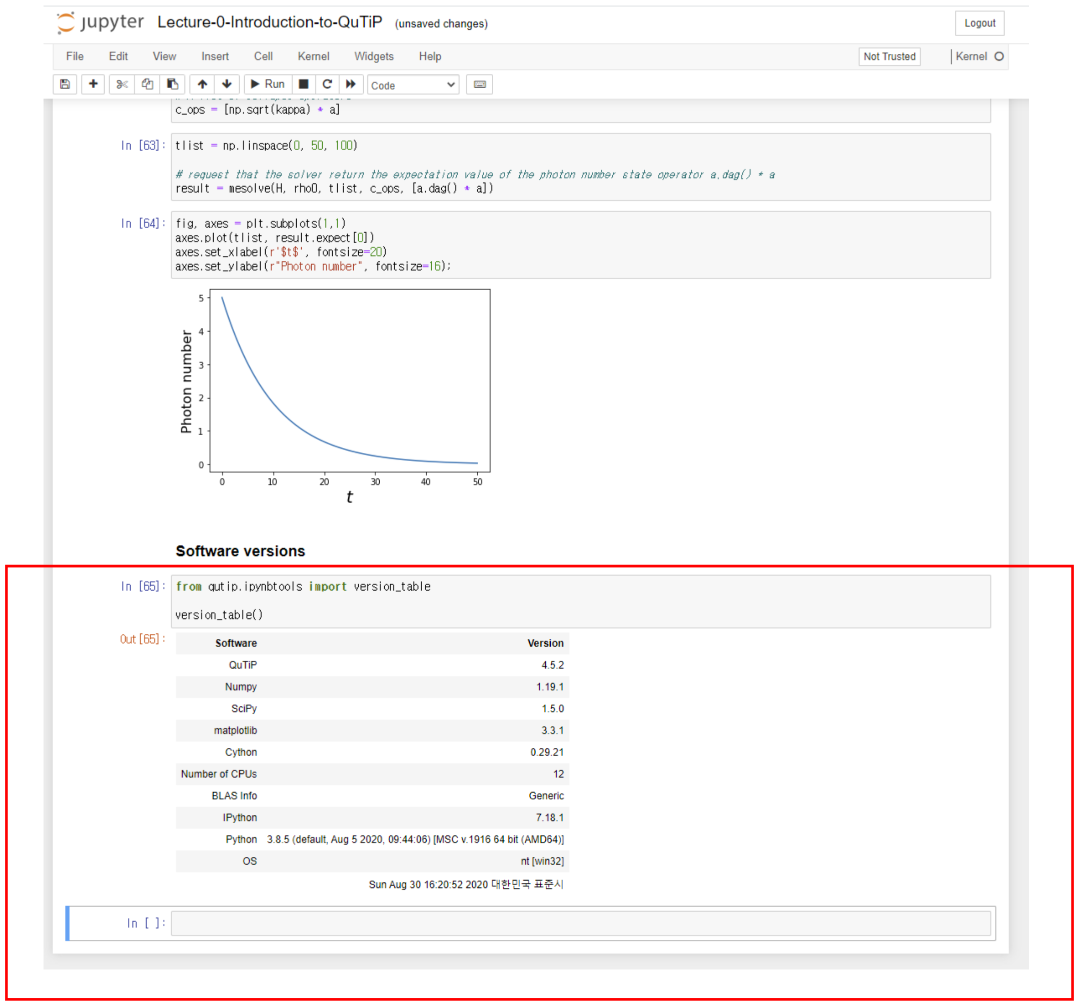
Out [32]: Quantum object: dims = [[2], [2]], shape = [2, 2], type = oper, isherm = True

$$\begin{pmatrix} 0.0 & 0.0 \\ 0.0 & 0.0 \end{pmatrix}$$

$-i\sigma_x\sigma_y\sigma_z = 1$

Line 34 에서는 에러날겁니다. 이는 qutip 문제가 아니고 python syntax 문제. Sqrt(2) 대신 np.sqrt(2) 로 하면 해결

마지막까지 에러없이 잘 실행되면 OKAY. (중간에 np. 붙여야 하는 부분 하나 더 있습니다.)



```
    c_ops = [np.sqrt(kappa) * a]

In [63]: tlist = np.linspace(0, 50, 100)

         # request that the solver return the expectation value of the photon number state operator a.dag() * a
         result = mesolve(H, rho0, tlist, c_ops, [a.dag() * a])

In [64]: fig, axes = plt.subplots(1,1)
         axes.plot(tlist, result.expect[0])
         axes.set_xlabel(r'$t$', fontsize=20)
         axes.set_ylabel(r"Photon number", fontsize=16);
```

Software versions

```
In [65]: from qutip.ipynbtools import version_table

         version_table()
```

Out [65]:

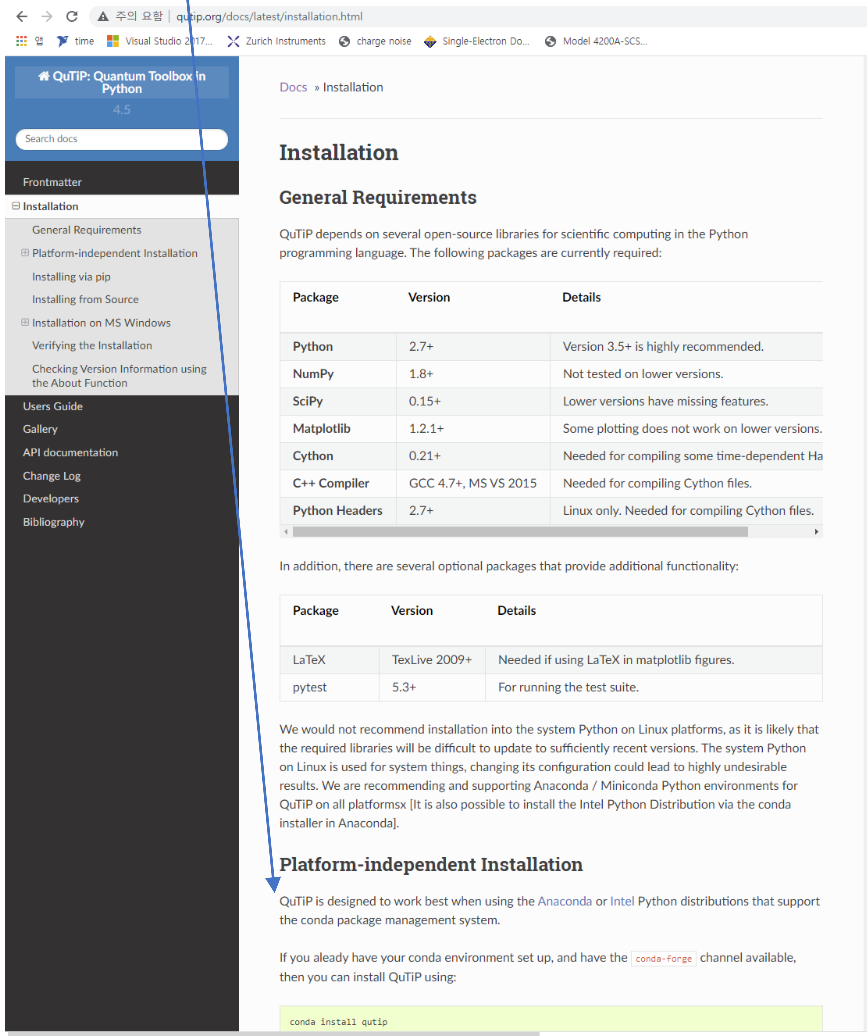
Software	Version
QuTiP	4.5.2
Numpy	1.19.1
SciPy	1.5.0
matplotlib	3.3.1
Cython	0.29.21
Number of CPUs	12
BLAS Info	Generic
IPython	7.18.1
Python	3.8.5 (default, Aug 5 2020, 09:44:06) [MSC v.1916 64 bit (AMD64)]
OS	nt [win32]

Sun Aug 30 16:20:52 2020 대한민국 표준시

```
In [ ]:
```

자신의 quTip 버전 확인.

참고사항: qutip 사이트의 user guide 에 나와있는 방법을 이용하여 conda 커멘드 프롬프트에 집접 타이핑해서 설치할 수도 있습니다. 우리의 설치법은 이 과정을 더 편하게 한 것이며 결과는 동일합니다.



Installation

General Requirements

QuTiP depends on several open-source libraries for scientific computing in the Python programming language. The following packages are currently required:

Package	Version	Details
Python	2.7+	Version 3.5+ is highly recommended.
NumPy	1.8+	Not tested on lower versions.
SciPy	0.15+	Lower versions have missing features.
Matplotlib	1.2.1+	Some plotting does not work on lower versions.
Cython	0.21+	Needed for compiling some time-dependent Ha
C++ Compiler	GCC 4.7+, MS VS 2015	Needed for compiling Cython files.
Python Headers	2.7+	Linux only. Needed for compiling Cython files.

In addition, there are several optional packages that provide additional functionality:

Package	Version	Details
LaTeX	TexLive 2009+	Needed if using LaTeX in matplotlib figures.
pytest	5.3+	For running the test suite.

We would not recommend installation into the system Python on Linux platforms, as it is likely that the required libraries will be difficult to update to sufficiently recent versions. The system Python on Linux is used for system things, changing its configuration could lead to highly undesirable results. We are recommending and supporting Anaconda / Miniconda Python environments for QuTiP on all platformsx [It is also possible to install the Intel Python Distribution via the conda installer in Anaconda].

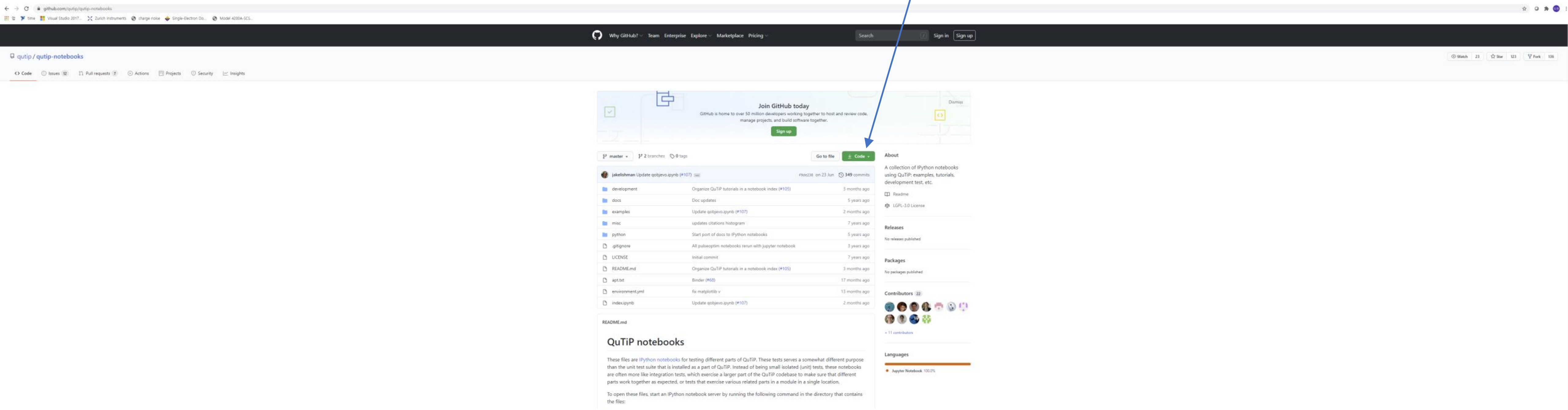
Platform-independent Installation

QuTiP is designed to work best when using the Anaconda or Intel Python distributions that support the conda package management system.

If you aleady have your conda environment set up, and have the `conda-forge` channel available, then you can install QuTiP using:

```
conda install qutip
```

한가지 더 ! : 여기까지 하면 많은 부분이 문제없이 실행될텐데: Time dependent Hamiltonian 풀어야하는 경우 먹통일 겁니다. 이번에는 https://github.com/qutip/qutip-notebooks 에 들어가서 ZIP 파일로 다운받습니다.



GitHub is home to over 50 million developers working together to host and review code, manage projects, and build software together.

jakelishman Update qobjevo.ipynb (#107) f9de238 on 23 Jun 349 commits

development	Organize QuTiP tutorials in a notebook index (#105)	3 months ago
docs	Doc updates	5 years ago
examples	Update qobjevo.ipynb (#107)	2 months ago
misc	updates citations histogram	7 years ago
python	Start port of docs to IPython notebooks	5 years ago
.gitignore	All pulseoptim notebooks rerun with jupyter notebook	3 years ago
LICENSE	Initial commit	7 years ago
README.md	Organize QuTiP tutorials in a notebook index (#105)	3 months ago
apt.txt	Binder (#68)	17 months ago
environment.yml	fix matplotlib v	13 months ago
index.ipynb	Update qobjevo.ipynb (#107)	2 months ago

README.md

QuTiP notebooks

These files are IPython notebooks for testing different parts of QuTiP. These tests serves a somewhat different purpose than the unit test suite that is installed as a part of QuTiP. Instead of being small isolated (unit) tests, these notebooks are often more like integration tests, which exercise a larger part of the QuTiP codebase to make sure that different parts work together as expected, or tests that exercise various related parts in a module in a single location.

To open these files, start an IPython notebook server by running the following command in the directory that contains the files:

About

A collection of IPython notebooks using QuTiP: examples, tutorials, development test, etc.

Releases

No releases published

Packages

No packages published

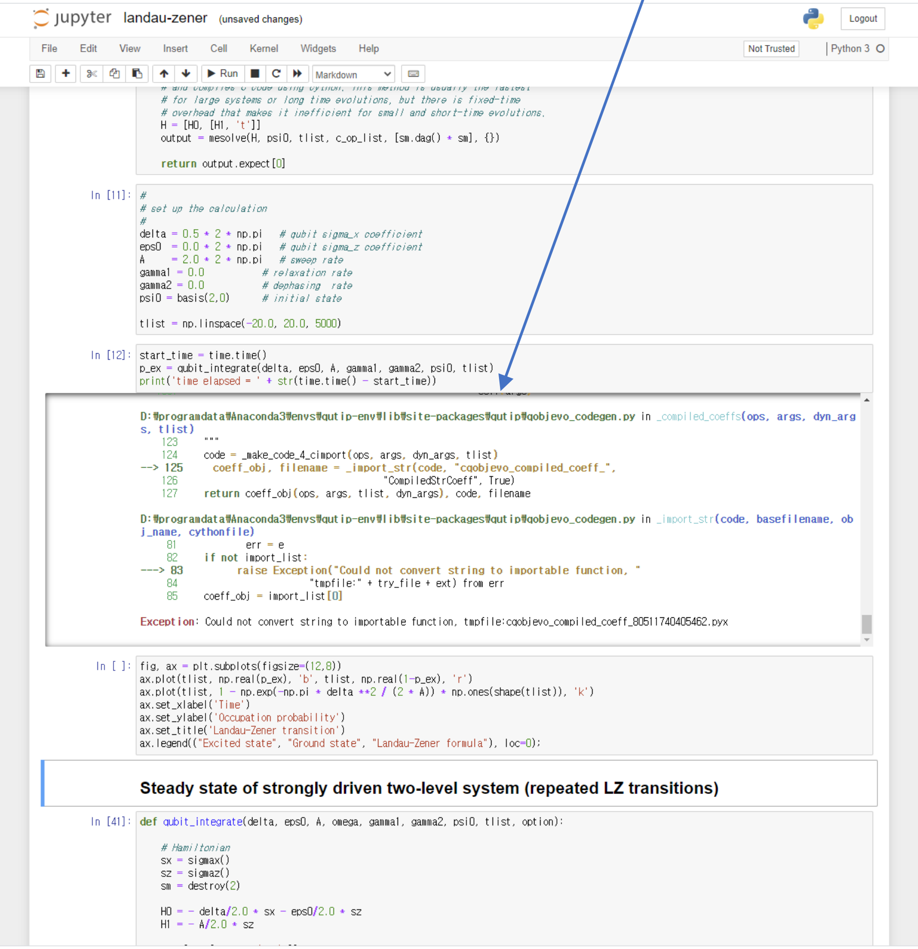
Contributors 22

+ 11 contributors

Languages

Jupyter Notebook 100.0%

다운받은 폴더/qutip-notebooks-master/qutip-notebooks-master/examples 폴더 안의 landau-Zener 파일을 열어서 순차적으로 실행합니다. 에러가 날겁니다.



```
    # and compiles C code using cython. This method is usually the fastest
    # for large systems or long time evolutions, but there is fixed-time
    # overhead that makes it inefficient for small and short-time evolutions.
    H = [H0, [H1, 't']]
    output = mesolve(H, psi0, tlist, c_op_list, [sm.dag() * sm], {})

    return output.expect[0]
```

```
In [11]: #
         # set up the calculation
         #
         delta = 0.5 * 2 * np.pi   # qubit sigma_x coefficient
         eps0  = 0.0 * 2 * np.pi   # qubit sigma_z coefficient
         A     = 2.0 * 2 * np.pi   # sweep rate
         gamma1 = 0.0            # relaxation rate
         gamma2 = 0.0            # dephasing  rate
         psi0 = basis(2,0)       # initial state

         tlist = np.linspace(-20.0, 20.0, 5000)
```

```
In [12]: start_time = time.time()
         p_ex = qubit_integrate(delta, eps0, A, gamma1, gamma2, psi0, tlist)
         print('time elapsed = ' + str(time.time() - start_time))
```

```
D:₩programdata₩Anaconda3₩envs₩qutip-env₩lib₩site-packages₩qutip₩qobjevo_codegen.py in _compiled_coeffs(ops, args, dyn_args, tlist)
    123     """
    124     code = _make_code_4_cimport(ops, args, dyn_args, tlist)
--> 125     coeff_obj, filename = _import_str(code, "cqobjevo_compiled_coeff_",
    126                                       "CompiledStrCoeff", True)
    127     return coeff_obj(ops, args, tlist, dyn_args), code, filename

D:₩programdata₩Anaconda3₩envs₩qutip-env₩lib₩site-packages₩qutip₩qobjevo_codegen.py in _import_str(code, basefilename, obj_name, cythonfile)
     81             err = e
     82     if not import_list:
---> 83         raise Exception("Could not convert string to importable function, "
     84                         "tmpfile:" + try_file + ext) from err
     85     coeff_obj = import_list[0]

Exception: Could not convert string to importable function, tmpfile:cqobjevo_compiled_coeff_80511740405462.pyx
```

```
In [ ]: fig, ax = plt.subplots(figsize=(12,8))
        ax.plot(tlist, np.real(p_ex), 'b', tlist, np.real(1-p_ex), 'r')
        ax.plot(tlist, 1 - np.exp(-np.pi * delta **2 / (2 * A)) * np.ones(shape(tlist)), 'k')
        ax.set_xlabel('Time')
        ax.set_ylabel('Occupation probability')
        ax.set_title('Landau-Zener transition')
        ax.legend(("Excited state", "Ground state", "Landau-Zener formula"), loc=0);
```

Steady state of strongly driven two-level system (repeated LZ transitions)

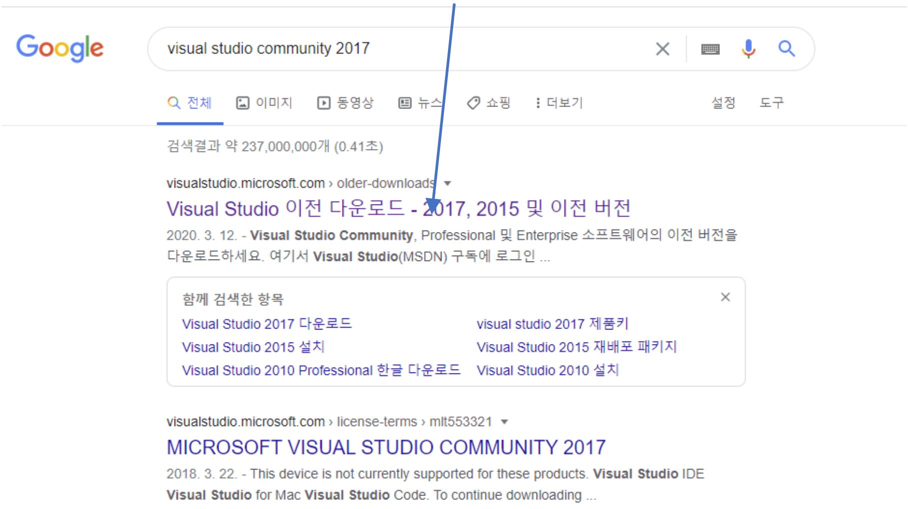
```
In [41]: def qubit_integrate(delta, eps0, A, omega, gamma1, gamma2, psi0, tlist, option):

             # Hamiltonian
             sx = sigmax()
             sz = sigmaz()
             sm = destroy(2)

             H0 = - delta/2.0 * sx - eps0/2.0 * sz
             H1 = - A/2.0 * sz
```

이 에러를 없애기 위해서는 Visual Studio 를 설치해야 합니다. 정확히는 Visual Studio Community 2017 버전의 C++ 데스트톱 개발 패키지와 Python 개발 패키지 입니다.

1. Visual studio community 2017 구글검색, 첫번째 아이템 클릭



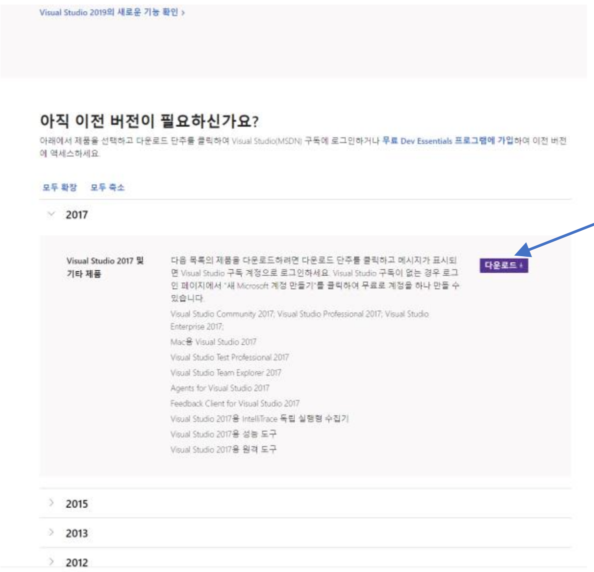
검색결과 약 237,000,000개 (0.41초)

visualstudio.microsoft.com › older-downloads ▾
Visual Studio 이전 다운로드 - 2017, 2015 및 이전 버전
2020. 3. 12. - **Visual Studio Community**, Professional 및 Enterprise 소프트웨어의 이전 버전을 다운로드하세요. 여기서 **Visual Studio**(MSDN) 구독에 로그인 ...

함께 검색한 항목

Visual Studio 2017 다운로드	visual studio 2017 제품키
Visual Studio 2015 설치	Visual Studio 2015 재배포 패키지
Visual Studio 2010 Professional 한글 다운로드	Visual Studio 2010 설치

visualstudio.microsoft.com › license-terms › mlt553321 ▾
MICROSOFT VISUAL STUDIO COMMUNITY 2017
2018. 3. 22. - This device is not currently supported for these products. **Visual Studio** IDE **Visual Studio** for Mac **Visual Studio** Code. To continue downloading ...



아직 이전 버전이 필요하신가요?

아래에서 제품을 선택하고 다운로드 단추를 클릭하여 Visual Studio(MSDN) 구독에 로그인하거나 무료 Dev Essentials 프로그램에 가입하여 이전 버전에 액세스하세요.

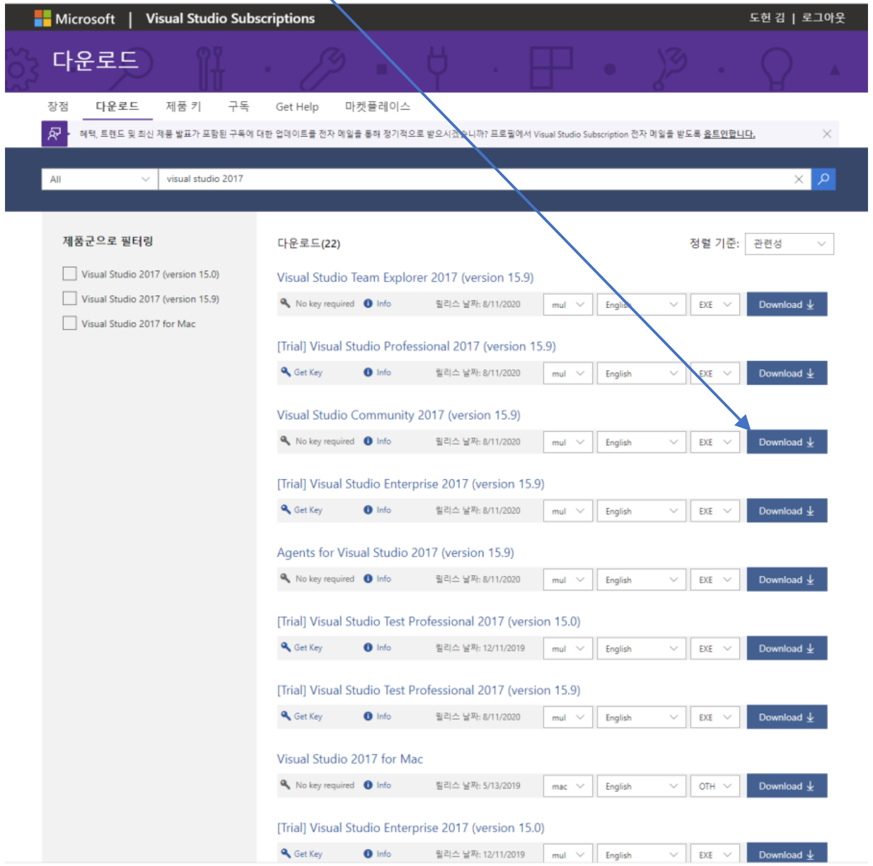
Visual Studio 2017 및 기타 제품

다음 목록의 제품을 다운로드하려면 다운로드 단추를 클릭하고 메시지가 표시되면 Visual Studio 구독 계정으로 로그인하세요. Visual Studio 구독이 없는 경우 로그인 페이지에서 "새 Microsoft 계정 만들기"를 클릭하여 무료로 계정을 하나 만들 수 있습니다.

Visual Studio Community 2017; Visual Studio Professional 2017; Visual Studio Enterprise 2017;
Mac용 Visual Studio 2017
Visual Studio Test Professional 2017
Visual Studio Team Explorer 2017
Agents for Visual Studio 2017
Feedback Client for Visual Studio 2017
Visual Studio 2017용 IntelliTrace 독립 실행형 수집기
Visual Studio 2017용 성능 도구
Visual Studio 2017용 원격 도구

2. 클릭

3. 요거 다운후 실행 – 더 새로운 버전은 테스트 안해봤으나 저는 2017 버전으로 잘 돌아갑니다.



Microsoft | Visual Studio Subscriptions   도헌 김 | 로그아웃

다운로드

장점 다운로드 제품 키 구독 Get Help 마켓플레이스

혜택, 트렌드 및 최신 제품 발표가 포함된 구독에 대한 업데이트를 전자 메일을 통해 정기적으로 받으시겠습니까? 프로필에서 Visual Studio Subscription 전자 메일을 받도록 옵트인합니다.

All | visual studio 2017

제품군으로 필터링
- □ Visual Studio 2017 (version 15.0)
- □ Visual Studio 2017 (version 15.9)
- □ Visual Studio 2017 for Mac

다운로드(22)   정렬 기준: 관련성

Visual Studio Team Explorer 2017 (version 15.9)
No key required | Info | 릴리스 날짜: 8/11/2020 | mul | English | EXE | Download

[Trial] Visual Studio Professional 2017 (version 15.9)
Get Key | Info | 릴리스 날짜: 8/11/2020 | mul | English | EXE | Download

Visual Studio Community 2017 (version 15.9)
No key required | Info | 릴리스 날짜: 8/11/2020 | mul | English | EXE | Download

[Trial] Visual Studio Enterprise 2017 (version 15.9)
Get Key | Info | 릴리스 날짜: 8/11/2020 | mul | English | EXE | Download

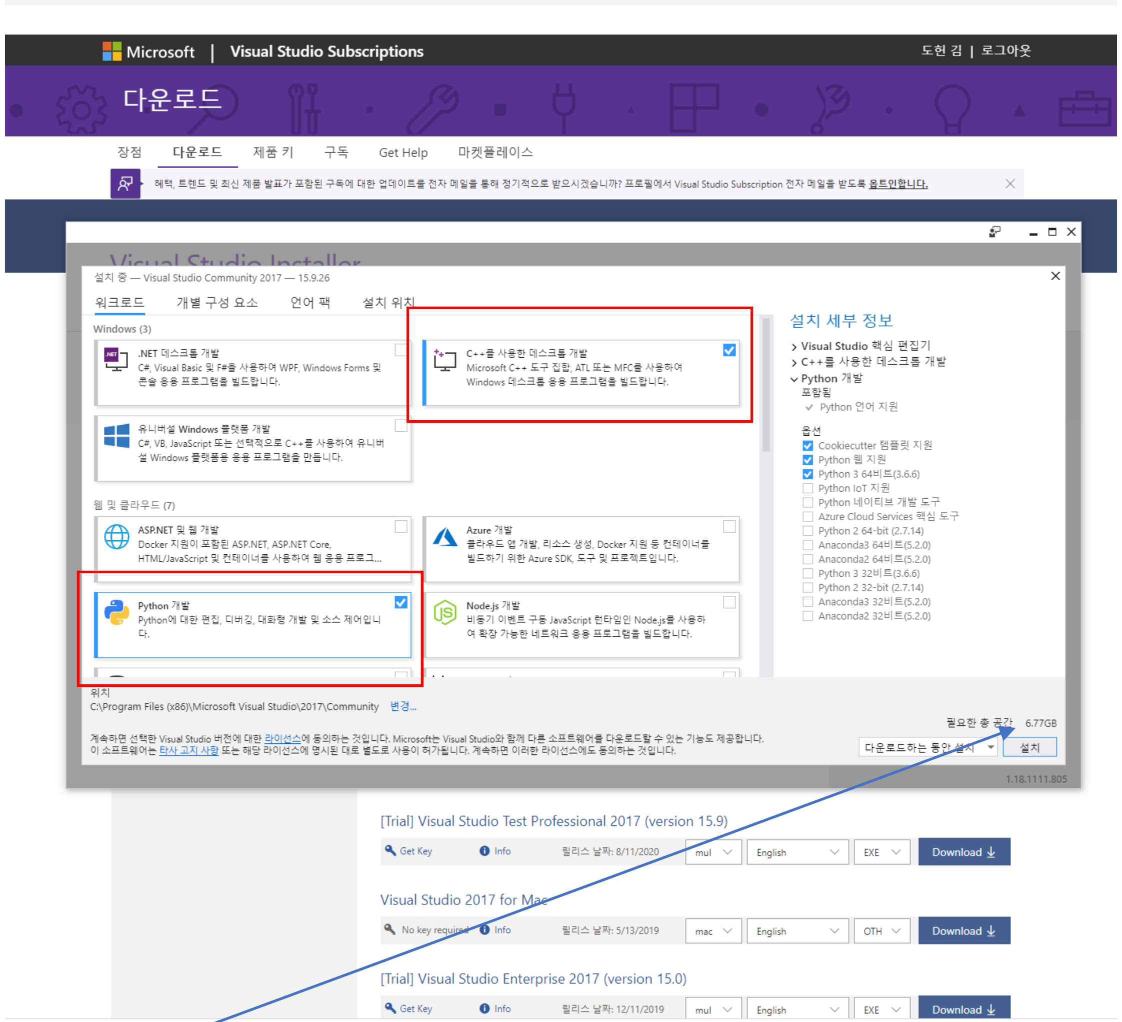
Agents for Visual Studio 2017 (version 15.9)
No key required | Info | 릴리스 날짜: 8/11/2020 | mul | English | EXE | Download

[Trial] Visual Studio Test Professional 2017 (version 15.0)
Get Key | Info | 릴리스 날짜: 12/11/2019 | mul | English | EXE | Download

[Trial] Visual Studio Test Professional 2017 (version 15.9)
Get Key | Info | 릴리스 날짜: 8/11/2020 | mul | English | EXE | Download

Visual Studio 2017 for Mac
No key required | Info | 릴리스 날짜: 5/13/2019 | mac | English | OTH | Download

[Trial] Visual Studio Enterprise 2017 (version 15.0)
Get Key | Info | 릴리스 날짜: 12/11/2019 | mul | English | EXE | Download



Microsoft | Visual Studio Subscriptions   도헌 김 | 로그아웃

다운로드

장점 다운로드 제품 키 구독 Get Help 마켓플레이스

혜택, 트렌드 및 최신 제품 발표가 포함된 구독에 대한 업데이트를 전자 메일을 통해 정기적으로 받으시겠습니까? 프로필에서 Visual Studio Subscription 전자 메일을 받도록 옵트인합니다.

Visual Studio Installer

설치 중 — Visual Studio Community 2017 — 15.9.26

워크로드 개별 구성 요소 언어 팩 설치 위치

Windows (3)

- □ .NET 데스크톱 개발
  C#, Visual Basic 및 F#을 사용하여 WPF, Windows Forms 및 콘솔 응용 프로그램을 빌드합니다.
- ☑ C++를 사용한 데스크톱 개발
  Microsoft C++ 도구 집합, ATL 또는 MFC를 사용하여 Windows 데스크톱 응용 프로그램을 빌드합니다.
- □ 유니버설 Windows 플랫폼 개발
  C#, VB, JavaScript 또는 선택적으로 C++를 사용하여 유니버설 Windows 플랫폼용 응용 프로그램을 만듭니다.

웹 및 클라우드 (7)

- □ ASP.NET 및 웹 개발
  Docker 지원이 포함된 ASP.NET, ASP.NET Core, HTML/JavaScript 및 컨테이너를 사용하여 웹 응용 프로그...
- □ Azure 개발
  클라우드 앱 개발, 리소스 생성, Docker 지원 등 컨테이너를 빌드하기 위한 Azure SDK, 도구 및 프로젝트입니다.
- ☑ Python 개발
  Python에 대한 편집, 디버깅, 대화형 개발 및 소스 제어입니다.
- □ Node.js 개발
  비동기 이벤트 구동 JavaScript 런타임인 Node.js를 사용하여 확장 가능한 네트워크 응용 프로그램을 빌드합니다.

설치 세부 정보
- › Visual Studio 핵심 편집기
- › C++를 사용한 데스크톱 개발
- ⌄ Python 개발
  - 포함됨
    - ✓ Python 언어 지원
  - 옵션
    - ☑ Cookiecutter 템플릿 지원
    - ☑ Python 웹 지원
    - ☑ Python 3 64비트(3.6.6)
    - □ Python IoT 지원
    - □ Python 네이티브 개발 도구
    - □ Azure Cloud Services 핵심 도구
    - □ Python 2 64-bit (2.7.14)
    - □ Anaconda3 64비트(5.2.0)
    - □ Anaconda2 64비트(5.2.0)
    - □ Python 3 32비트(3.6.6)
    - □ Python 2 32-bit (2.7.14)
    - □ Anaconda3 32비트(5.2.0)
    - □ Anaconda2 32비트(5.2.0)

위치
C:\Program Files (x86)\Microsoft Visual Studio\2017\Community 변경...

계속하면 선택한 Visual Studio 버전에 대한 라이선스에 동의하는 것입니다. Microsoft는 Visual Studio와 함께 다른 소프트웨어를 다운로드할 수 있는 기능도 제공합니다. 이 소프트웨어는 타사 고지 사항 또는 해당 라이선스에 명시된 대로 별도로 사용이 허가됩니다. 계속하면 이러한 라이선스에도 동의하는 것입니다.

필요한 총 공간 6.77GB

다운로드하는 동안 설치 | 설치

1.18.1111.805

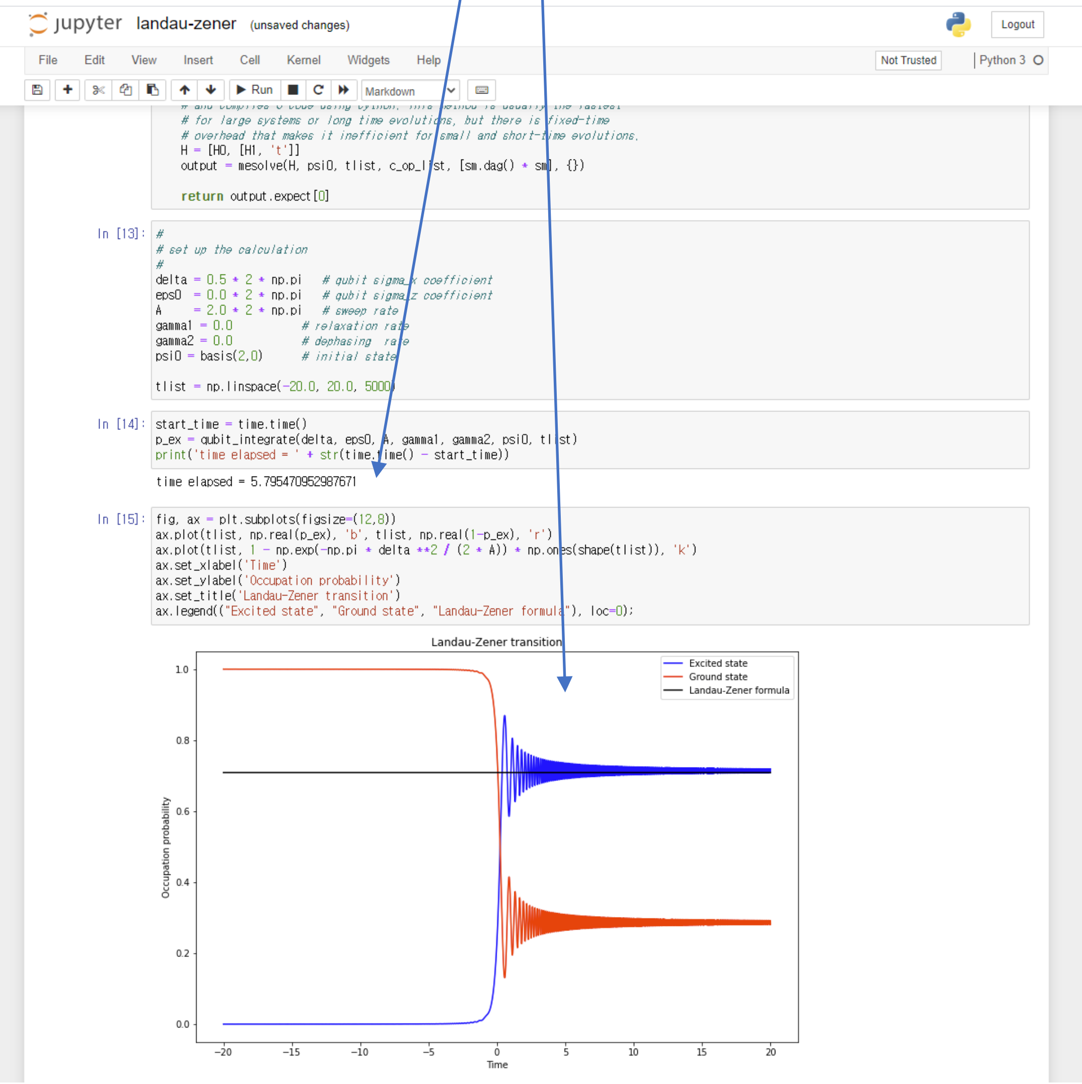
[Trial] Visual Studio Test Professional 2017 (version 15.9)
Get Key | Info | 릴리스 날짜: 8/11/2020 | mul | English | EXE | Download

Visual Studio 2017 for Mac
No key required | Info | 릴리스 날짜: 5/13/2019 | mac | English | OTH | Download

[Trial] Visual Studio Enterprise 2017 (version 15.0)
Get Key | Info | 릴리스 날짜: 12/11/2019 | mul | English | EXE | Download

4. 설치시 빨간박스의 두 아이템 체크하고 설치 클릭. 시간좀 걸립니다.

5. 설치 후 바로 (재부팅이나 jupyter 노트북 다시 열 필요 없습니다.) 15 슬라이드의 landau-Zener 다시 실행하면 에러없이 돌아감.



```
    # and compiles C code using Cython. This method is usually the fastest
    # for large systems or long time evolutions, but there is fixed-time
    # overhead that makes it inefficient for small and short-time evolutions.
    H = [H0, [H1, 't']]
    output = mesolve(H, psi0, tlist, c_op_list, [sm.dag() * sm], {})

    return output.expect[0]
```

In [13]:
```
#
# set up the calculation
#
delta = 0.5 * 2 * np.pi   # qubit sigma_x coefficient
eps0  = 0.0 * 2 * np.pi   # qubit sigma_z coefficient
A     = 2.0 * 2 * np.pi   # sweep rate
gamma1 = 0.0           # relaxation rate
gamma2 = 0.0           # dephasing  rate
psi0 = basis(2,0)      # initial state

tlist = np.linspace(-20.0, 20.0, 5000)
```

In [14]:
```
start_time = time.time()
p_ex = qubit_integrate(delta, eps0, A, gamma1, gamma2, psi0, tlist)
print('time elapsed = ' + str(time.time() - start_time))
```

```
time elapsed = 5.795470952987671
```

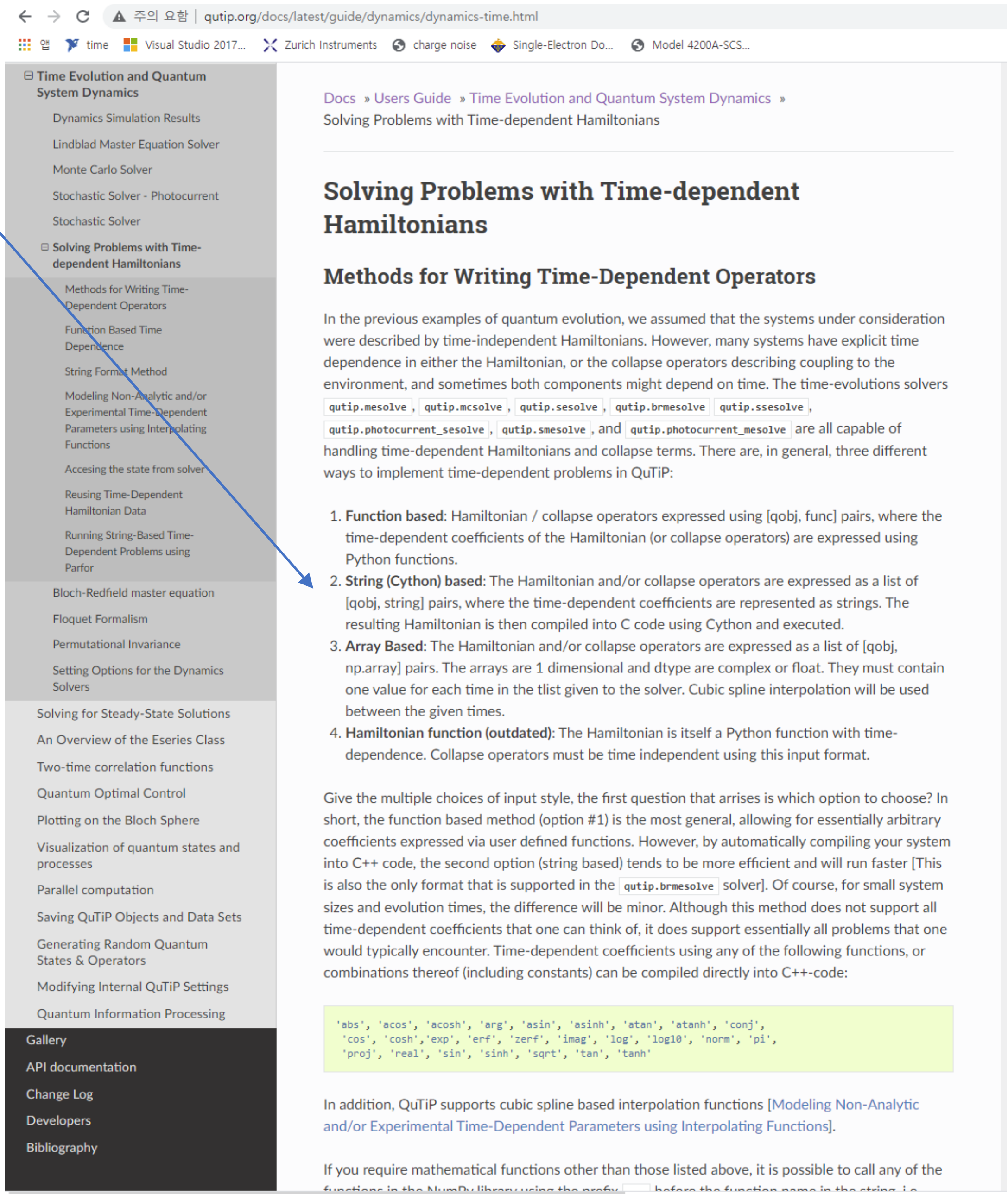
In [15]:
```
fig, ax = plt.subplots(figsize=(12,8))
ax.plot(tlist, np.real(p_ex), 'b', tlist, np.real(1-p_ex), 'r')
ax.plot(tlist, 1 - np.exp(-np.pi * delta **2 / (2 * A)) * np.ones(shape(tlist)), 'k')
ax.set_xlabel('Time')
ax.set_ylabel('Occupation probability')
ax.set_title('Landau-Zener transition')
ax.legend(("Excited state", "Ground state", "Landau-Zener formula"), loc=0);
```

6. 이 부분 아래의 코드들도 실행해서 에러나지 않는 것을 확인하세요. (visual studio community 2017 의 패키지들 설치 전에는 아래도 다 에러 납니다.)

주의: 이 중 2차원 그림을 그리는 계산도 있습니다. 그냥 실행하면 몇시간 걸리는 계산도 있으니 계산 개수를 줄여서 실행하세요)

7. 마지막으로 슬라이드 10의 import qutip.testing as qt qt.run() 을 다시한번 실행해봅니다. Fail 또는 기능이 아직 많이 있는지 확인합니다. (저는 시간이 오래걸려서 패스~)

참고: qutip 은 time dependent Hamiltonian 을 풀기위해서 3가지 서로 다른 방법을 제공하고 있고 여기서 설명한 방법은 그 중 두번째 cython 을 이용한 방법에 해당합니다. 예제 노트북들에서 time dependent Hamiltonian 이 등장하는 경우 대부분 세가지 방법에 대한 코딩을 작성해놓고 선택하게 되어 있으니, 한번 실행했을때 에러가 나더라도 string-based 방법으로 선택 (commented out 된 부분 수정)해서 실행하면 에러 없이 실행됩니다.


주의 요함 | qutip.org/docs/latest/guide/dynamics/dynamics-time.html
앱
time
Visual Studio 2017...
Zurich Instruments
charge noise
Single-Electron Do...
Model 4200A-SCS...
⊟ Time Evolution and Quantum System Dynamics
Dynamics Simulation Results
Lindblad Master Equation Solver
Monte Carlo Solver
Stochastic Solver - Photocurrent
Stochastic Solver
⊟ Solving Problems with Time-dependent Hamiltonians
Methods for Writing Time-Dependent Operators
Function Based Time Dependence
String Format Method
Modeling Non-Analytic and/or Experimental Time-Dependent Parameters using Interpolating Functions
Accesing the state from solver
Reusing Time-Dependent Hamiltonian Data
Running String-Based Time-Dependent Problems using Parfor
Bloch-Redfield master equation
Floquet Formalism
Permutational Invariance
Setting Options for the Dynamics Solvers
Solving for Steady-State Solutions
An Overview of the Eseries Class
Two-time correlation functions
Quantum Optimal Control
Plotting on the Bloch Sphere
Visualization of quantum states and processes
Parallel computation
Saving QuTiP Objects and Data Sets
Generating Random Quantum States & Operators
Modifying Internal QuTiP Settings
Quantum Information Processing
Gallery
API documentation
Change Log
Developers
Bibliography
Docs » Users Guide » Time Evolution and Quantum System Dynamics »
Solving Problems with Time-dependent Hamiltonians

Solving Problems with Time-dependent Hamiltonians

Methods for Writing Time-Dependent Operators

In the previous examples of quantum evolution, we assumed that the systems under consideration were described by time-independent Hamiltonians. However, many systems have explicit time dependence in either the Hamiltonian, or the collapse operators describing coupling to the environment, and sometimes both components might depend on time. The time-evolutions solvers `qutip.mesolve`, `qutip.mcsolve`, `qutip.sesolve`, `qutip.brmesolve` `qutip.ssesolve`, `qutip.photocurrent_sesolve`, `qutip.smesolve`, and `qutip.photocurrent_mesolve` are all capable of handling time-dependent Hamiltonians and collapse terms. There are, in general, three different ways to implement time-dependent problems in QuTiP:

1. **Function based**: Hamiltonian / collapse operators expressed using [qobj, func] pairs, where the time-dependent coefficients of the Hamiltonian (or collapse operators) are expressed using Python functions.
2. **String (Cython) based**: The Hamiltonian and/or collapse operators are expressed as a list of [qobj, string] pairs, where the time-dependent coefficients are represented as strings. The resulting Hamiltonian is then compiled into C code using Cython and executed.
3. **Array Based**: The Hamiltonian and/or collapse operators are expressed as a list of [qobj, np.array] pairs. The arrays are 1 dimensional and dtype are complex or float. They must contain one value for each time in the tlist given to the solver. Cubic spline interpolation will be used between the given times.
4. **Hamiltonian function (outdated)**: The Hamiltonian is itself a Python function with time-dependence. Collapse operators must be time independent using this input format.

Give the multiple choices of input style, the first question that arises is which option to choose? In short, the function based method (option #1) is the most general, allowing for essentially arbitrary coefficients expressed via user defined functions. However, by automatically compiling your system into C++ code, the second option (string based) tends to be more efficient and will run faster [This is also the only format that is supported in the `qutip.brmesolve` solver]. Of course, for small system sizes and evolution times, the difference will be minor. Although this method does not support all time-dependent coefficients that one can think of, it does support essentially all problems that one would typically encounter. Time-dependent coefficients using any of the following functions, or combinations thereof (including constants) can be compiled directly into C++-code:

```
'abs', 'acos', 'acosh', 'arg', 'asin', 'asinh', 'atan', 'atanh', 'conj',
 'cos', 'cosh','exp', 'erf', 'zerf', 'imag', 'log', 'log10', 'norm', 'pi',
 'proj', 'real', 'sin', 'sinh', 'sqrt', 'tan', 'tanh'
```

In addition, QuTiP supports cubic spline based interpolation functions [Modeling Non-Analytic and/or Experimental Time-Dependent Parameters using Interpolating Functions].

If you require mathematical functions other than those listed above, it is possible to call any of the functions in the NumPy library using the prefix before the function name in the string, i.e.

중시계 스쿨 강의 전 할일 !

1. 다운받은 두가지 노트북 묶음 (qutip-lectures 어쩌구.. 와 qutip-notebooks)의 모든 주피터 노트북을 열어서 순차적으로 실행해보고 에러 안나는지 체크하기
- *참고: 아마 qutip 의 기능 중 계산결과를 에니메이션으로 만들어서 동영상으로 만드는 기능은 아직 안될것입니다. 본 강의에서는 크게 필요없어 해결법은 건너뛰겠습니다. 그러나 어렵지 않으니 원하는 분은 동영상 코덱 설치 등.. 방법을 검색해보세요 ㅎ*

2. 2019년도 중시계 강의노트 한번 읽어보고 오기

9월3일 목요일에는 qutip 의 아주 간략한 사용법과 이를 이용한 몇가지 재미있는 양자역학 문제 풀기를 실습하겠습니다.