

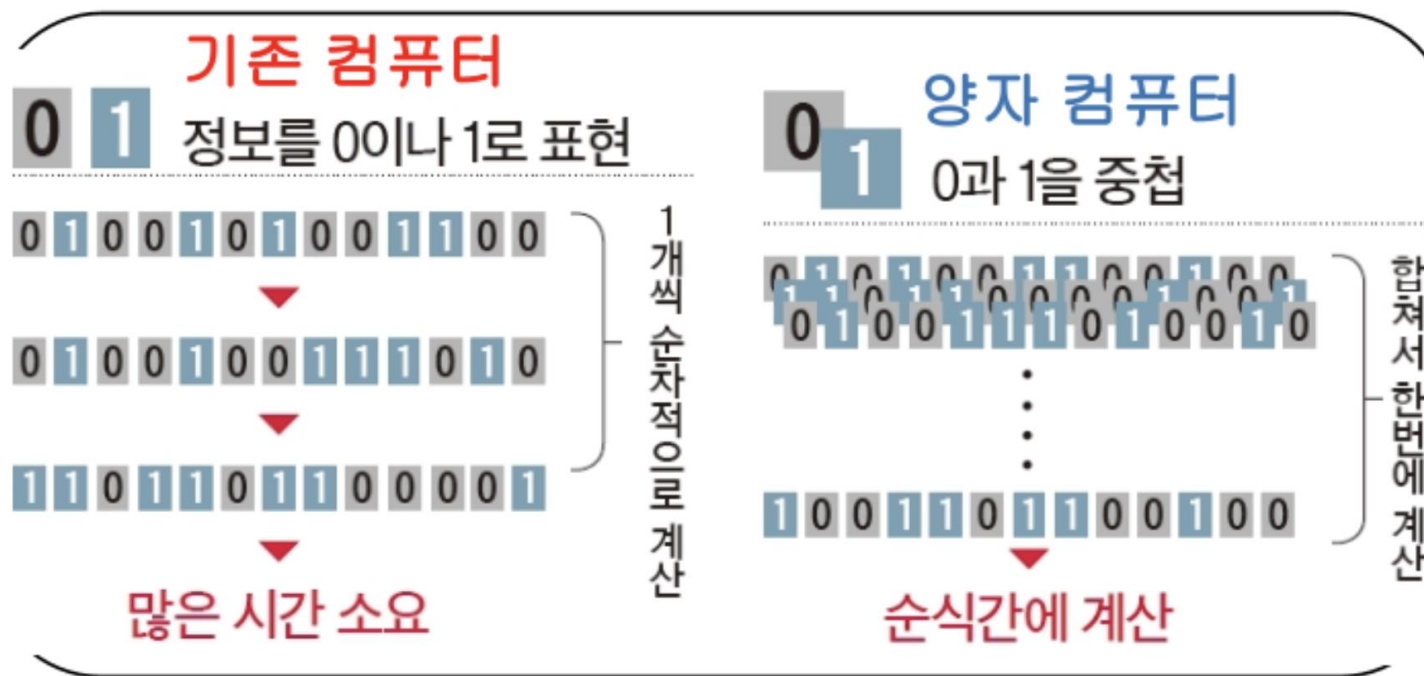
# 양자 프로그래밍 1강

## quantum programing basic

Quantum Ant

# 양자프로그래밍 기초

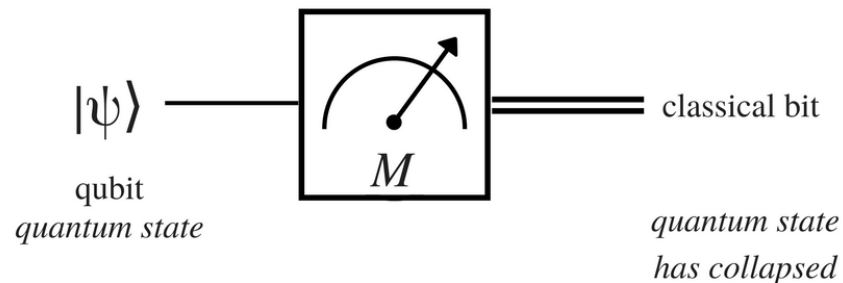
- 양자컴퓨터 : 중첩 및 얽힘의 양자 상태를 활용하여 계산을 수행하는 컴퓨터, 기존 컴퓨터와 다르게 0과1을 동시에 공존시킬 수 있어 연산속도가 굉장히 빠름
- 많은 연구가 이뤄지고 있지만 양자컴퓨터를 만들고 다루기 까다롭기 때문에 아직 실용성에는 문제가...



# 양자프로그래밍 기초

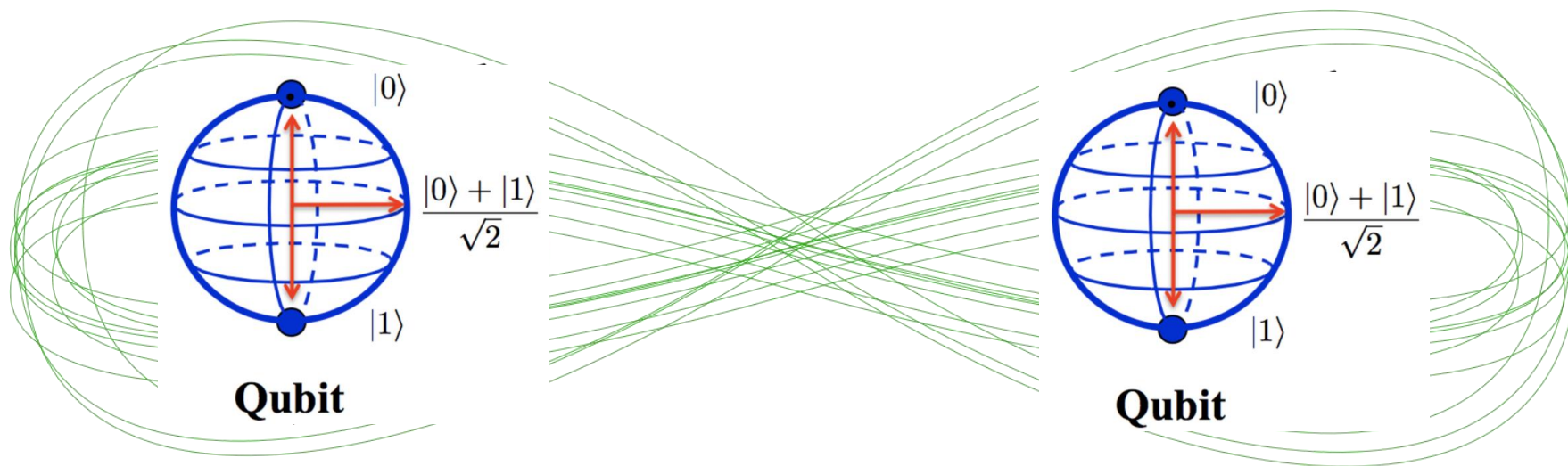
## • 양자 중첩

- 인자는 여러 상태를 확률적으로 가지고 있음.
- 측정 시 하나의 상태로 결정됨.



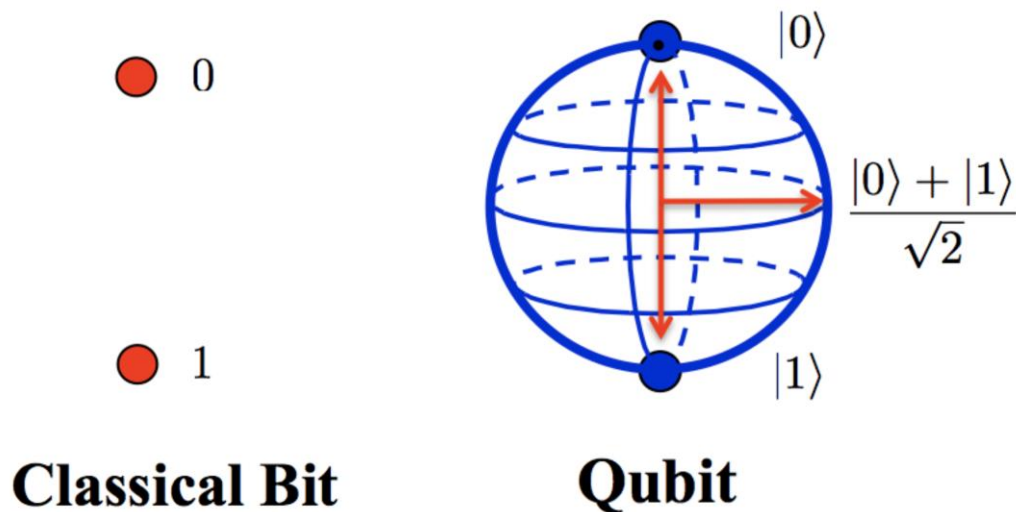
## • 양자 얽힘

- 과거에 상호작용했던 입자들이 특별한 관계를 유지하는 것.
- 한 입자의 특성을 측정했을때 멀리 떨어진 다른 한 입자의 특성이 즉시 바뀜.

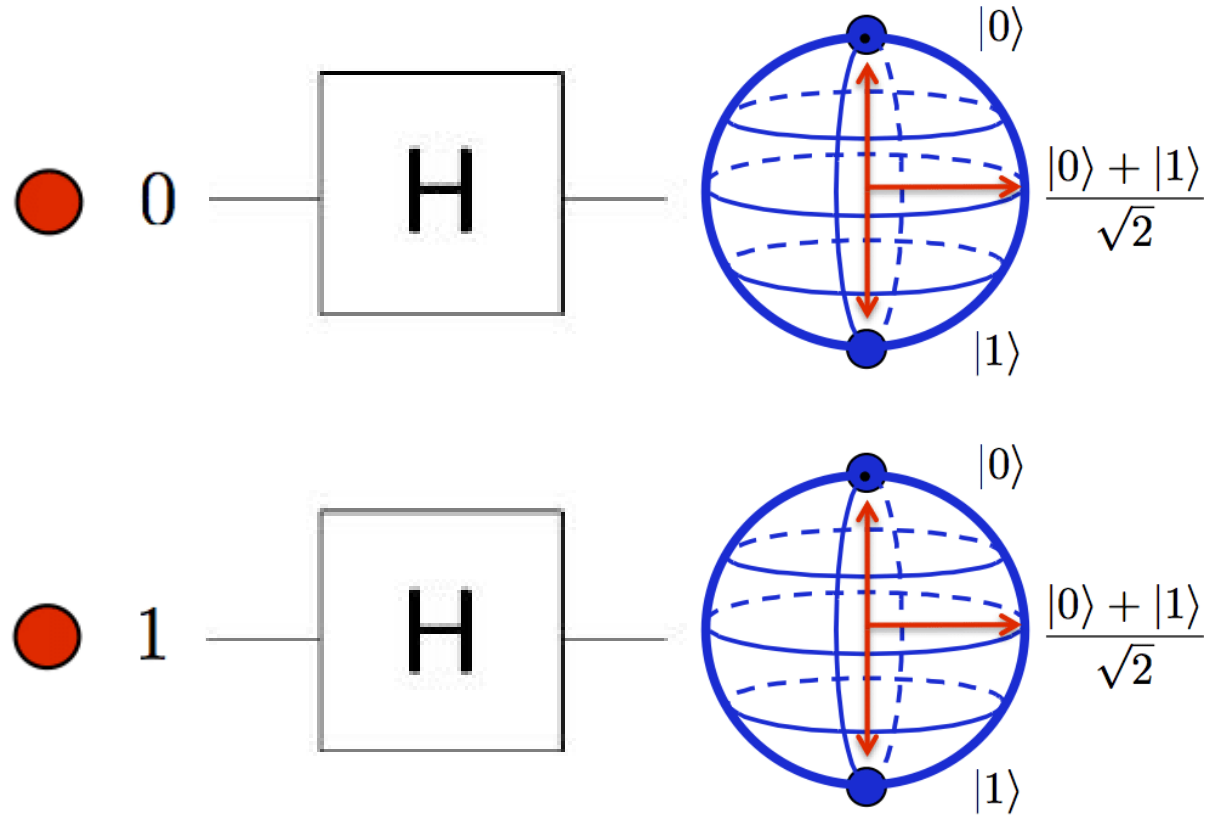


# 양자프로그래밍 기초

- 큐비트: 0과 1의 확률을 동시에 갖는 중첩상태.
- 큐비트는 모든 값이 확률로서 존재하므로  $n$ -qubit 으로  $2^n$  개의 상태를 나타낼 수 있다.
- 측정을 통해 상태가 정해진다.



# 양자 게이트



H gate

Input의 상태와 상관없이 qubit을 중첩 상태로 만든다.

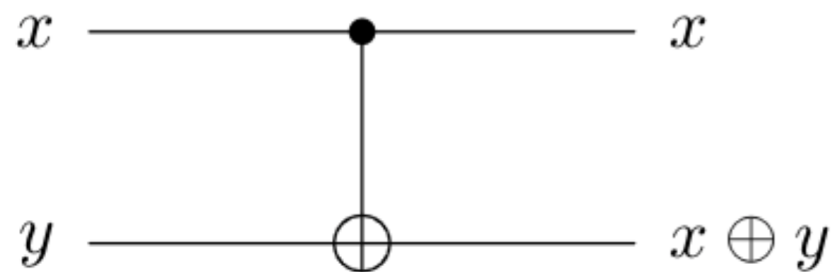
# 양자 게이트



(a) NOT/X gate

## NOT/X gate

1개의 input 큐비트의 상태를 반전시킨다.  
(Ex.  $0 \rightarrow 1$ ,  $1 \rightarrow 0$ )

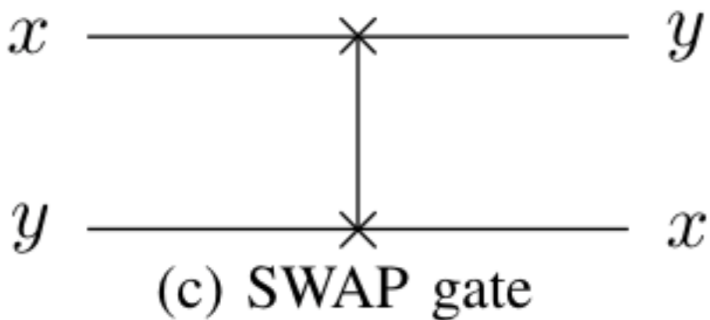


(b) CNOT gate

## CNOT -gate

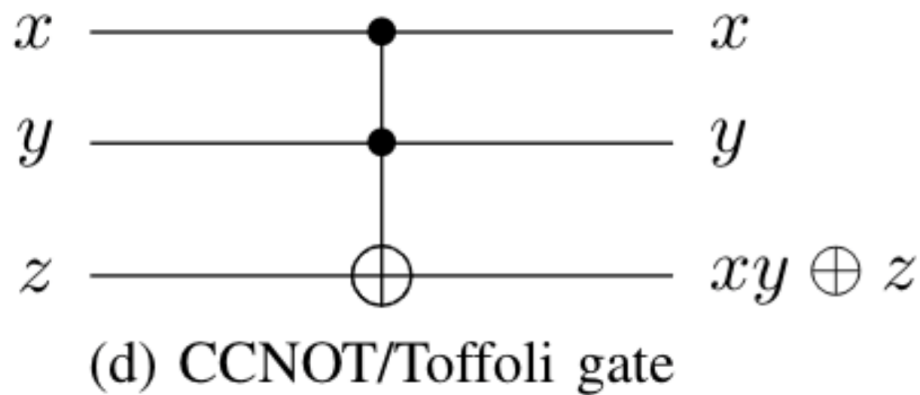
두개의 input 큐비트 중 하나의 큐비트는 control 큐비트의 역할을 한다.  
control 큐비트가 1일 경우 대상 큐비트의 값이 반전된다.

# 양자 게이트



**Swap gate**

두개의 input 큐비트의 상태를 교환한다.



**Toffoli gate**

세개의 input 큐비트 중 하나의 큐비트는 control 큐비트의 역할을 한다.  
control 큐비트가 1일 경우 대상 큐비트의 값이 반전된다.

# 양자 프로그래밍 2강

## 환경세팅

Quantum Ant

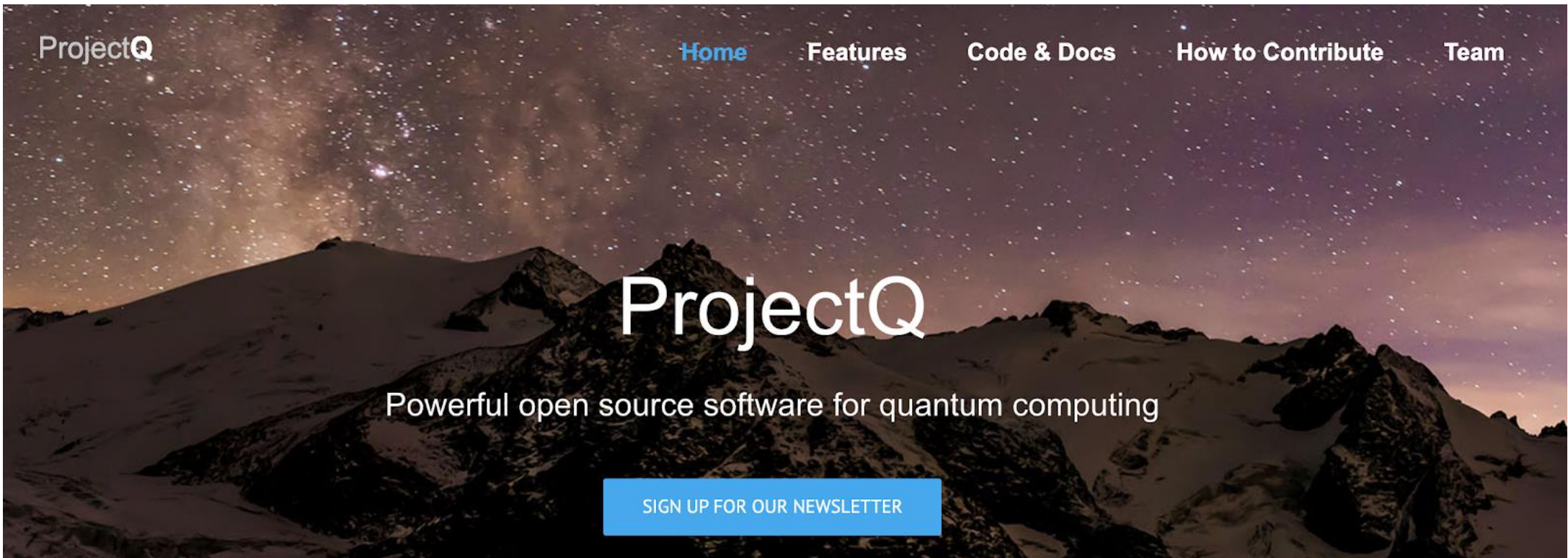


# 양자프로그래밍

- 큐비트(양자 비트), 양자 게이트를 사용하여 알고리즘 및 프로그램 구현
- IBM, 마이크로소프트, ETRI 등, 양자 프로그래밍을 위한 플랫폼을 제공중
  - IBM
    - ProjectQ
    - Qiskit
  - 마이크로소프트
    - Q#
  - ETRI
    - Q | Crypton

# IBM ProjectQ

IBM에서 제공하는 파이썬 기반의 양자 프로그래밍 플랫폼



# Python 설치



## Active Python Releases

For more information visit the [Python Developer's Guide](#).

Python version	Maintenance status	First released	End of support	Release schedule
3.9	bugfix	2020-10-05	2025-10	<a href="#">PEP 596</a>
3.8	security	2019-10-14	2024-10	<a href="#">PEP 569</a>
3.7	security	2018-06-27		
3.6	security	2016-12-23		
2.7	end-of-life	2010-07-03		

<https://www.python.org/downloads/>

Python 홈페이지에서 환경에 맞는 python 3.x 설치

# PyCharm 설치

PyCharm

[What's New](#) [Features](#) [Learn](#) [Buy](#)

[Download](#)



Version: 2021.2  
Build: 212.4746.96  
28 July 2021

[System requirements](#)

[Installation Instructions](#)

[Other versions](#)

## Download PyCharm

[Windows](#)

[macOS](#)

[Linux](#)

### Professional

For both Scientific and Web Python development. With HTML, JS, and SQL support.

[Download](#)

[.dmg \(Intel\) ▼](#)

Free trial

Available for Intel and Apple Silicon

### Community

For pure Python development

[Download](#)

[.dmg \(Intel\) ▼](#)

Free, open-source

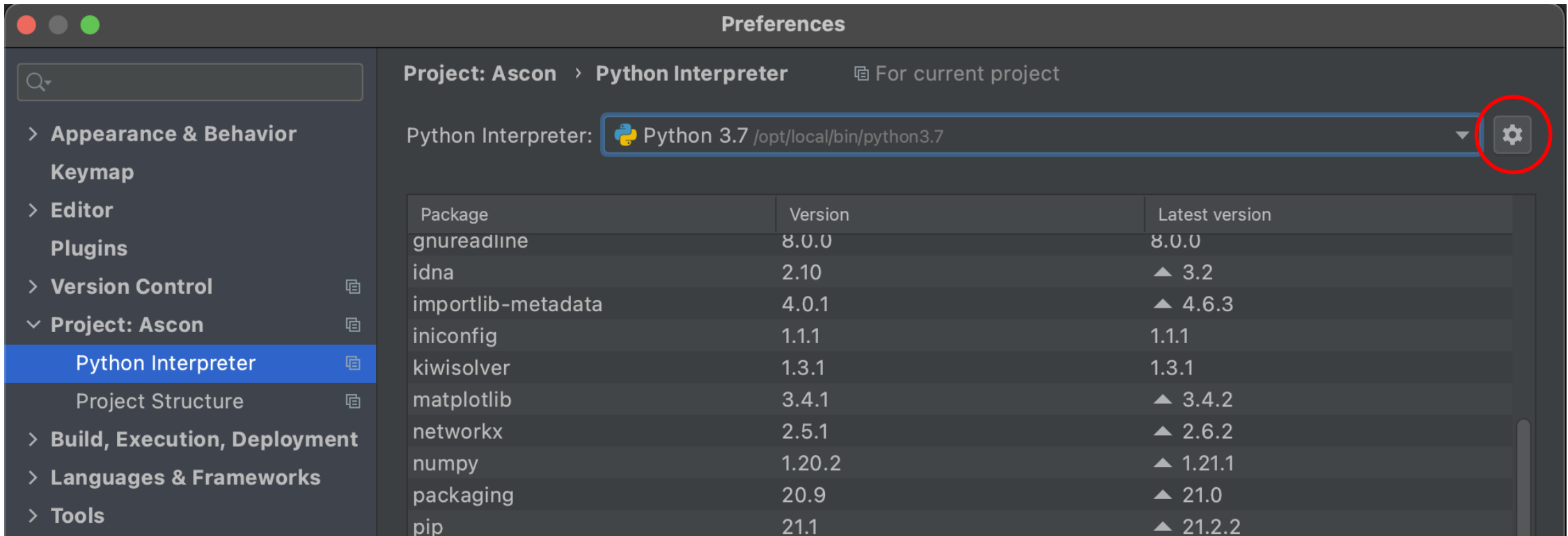
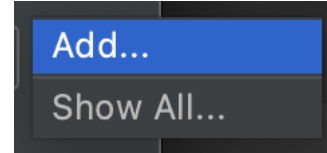
Available for Intel and Apple Silicon

<https://www.jetbrains.com/pycharm/download/>


PyCharm Community 설치

# Python, PyCharm 연결


PyCharm → Preferences → Python Interpreter →



# Python, PyCharm 연결

 ☒ New environment  
New environment 체크

Location:

Base interpreter:   설치한 python 선택

☐ Inherit global site-packages

☐ Make available to all projects

☐ Existing environment

Interpreter:

☐ Make available to all projects

# ProjectQ 설치

Pycharm Terminal에서 명령어를 통해 projectq 설치

```
(venv) (base) song-gyeongju@song-gyeongjuui-MacBookPro quantum_basic % pip install projectQ 설치 명령어
Collecting projectQ
  Downloading projectq-0.7.0.tar.gz (417 kB)
    |████████████████████████████████████████| 417 kB 810 kB/s
  Installing build dependencies ... done
  Getting requirements to build wheel ... done
  Preparing wheel metadata ... done
Collecting numpy
  Downloading numpy-1.21.1-cp39-cp39-macosx_10_9_x86_64.whl (17.0 MB)
    |████████████████████████████████████████| 17.0 MB 825 kB/s
Collecting requests
  Downloading requests-2.26.0-py2.py3-none-any.whl (62 kB)
    |████████████████████████████████████████| 62 kB 1.6 MB/s
Collecting matplotlib>=2.2.3
  Using cached matplotlib-3.4.2-cp39-cp39-macosx_10_9_x86_64.whl (7.2 MB)
Collecting networkx>=2
  Downloading networkx-2.6.2-py3-none-any.whl (1.9 MB)
    |████████████████████████████████████████| 1.9 MB 6.3 MB/s
```

Terminal: Local × +

```
(base) song-gyeongju@song-gyeongjuui-MacBookPro Ascon % pip install projectq
Requirement already satisfied: projectq in /Users/song-gyeongju/.local/lib/python3.8/site-packages (0.5.1)
```

이미 설치된 경우 출력되는 메시지

# ProjectQ 설치

The image shows the PyCharm Preferences dialog box, specifically the 'Python Interpreter' section for the project 'Ascon'. The 'Python Interpreter' dropdown is set to 'Python 3.7 /opt/local/bin/python3.7'. Below this, a table lists installed packages and their versions. The 'projectq' package is highlighted with a red box, showing version 0.5.1, with a red arrow pointing to the '0.7.0' latest version. A red text label 'Projectq 설치 확인' (Check Projectq installation) is next to the latest version.

**PyCharm** File Edit View

About PyCharm  
Check for Updates...

**Preferences...** ⌘ ,

Services >

Hide PyCharm ⌘ H  
Hide Others ⌘ ⌘ H  
Show All

Quit PyCharm ⌘ Q

**Preferences**

**Project: Ascon** > **Python Interpreter** For current project

Python Interpreter: Python 3.7 /opt/local/bin/python3.7

Package	Version	Latest version
gnureadline	8.0.0	8.0.0
idna	2.10	▲ 3.2
importlib-metadata	4.0.1	▲ 4.6.3
iniconfig	1.1.1	1.1.1
kiwisolver	1.3.1	1.3.1
matplotlib	3.4.1	▲ 3.4.2
networkx	2.5.1	▲ 2.6.2
numpy	1.20.2	▲ 1.21.1
packaging	20.9	▲ 21.0
pip	21.1	▲ 21.2.2
pluggy	0.13.1	0.13.1
<b>projectq</b>	<b>0.5.1</b>	<b>▲ 0.7.0</b>
py	1.10.0	1.10.0
pybind11	2.6.2	▲ 2.7.0
pyparsing	2.4.7	2.4.7

Projectq 설치 확인



감사합니다

# 양자 프로그래밍 3강

## 양자 프로그래밍 실습

Quantum Ant

# 양자 프로그래밍

```
from projectq.ops import H, CNOT, Measure, Toffoli, X, All, Swap  사용할 양자 게이트 import
```

```
x1 = eng.allocate_qubit()  하나의 큐비트 선언  
x2 = eng.allocate_quireg(5)  5개의 큐비트 배열 선언
```

```
with Control(eng, x0):  큐비트에 if 문을 적용할 때 사용  
    X | x1  대상 큐비트들을 얽힘상태로 만듦  
            X0 이 1일때만 Control문 동작
```

```
with Compute(eng):  Compute 안에 코드를 작성하면  
    CNOT | (x0, x2[4])  Uncompute 명령어로 Reverse 연산 수행  
    X | x0  *Reverse : 수행한 연산을 되돌림 (역연산)  
Uncompute(eng)
```

```
Resource = ClassicalSimulator()  
eng = MainEngine(Resource)  큐비트 상태 확인  
main(eng)  
print(Resource)
```

```
Resource = ResourceCounter()  
eng = MainEngine(Resource)  사용한 양자 자원 확인  
main(eng)  
print(Resource)
```

# 양자 프로그래밍

간단한 실습으로 확인..

# 양자 프로그래밍

측정을 통한 큐비트 상태 출력 및 양자 리소스 확인

```
/opt/local/bin/python3.7 /Users/song-gyeongju/PycharmProjects/quantum_basic/test.py  
1  
011111
```

Gate counts:

Allocate : 7

CCX : 1

CX : 2

Deallocate : 7

Measure : 1

X : 7

Max. width (number of qubits) : 3.

Depth : 5.

CCX (Toffoli-gate) : 1개

CX (CNOT-gate) : 2개

X (Not-gate): 7개

Depth : 5

감사합니다