

ELKH

Eötvös Loránd
Research Network

ELKH Cloud Kvantum Referencia Architektúra

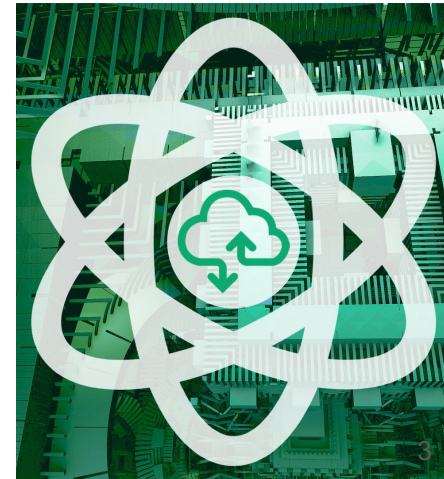
Farkas Zoltán, ELKH SZTAKI LPDS

zfarkas@sztaki.hu

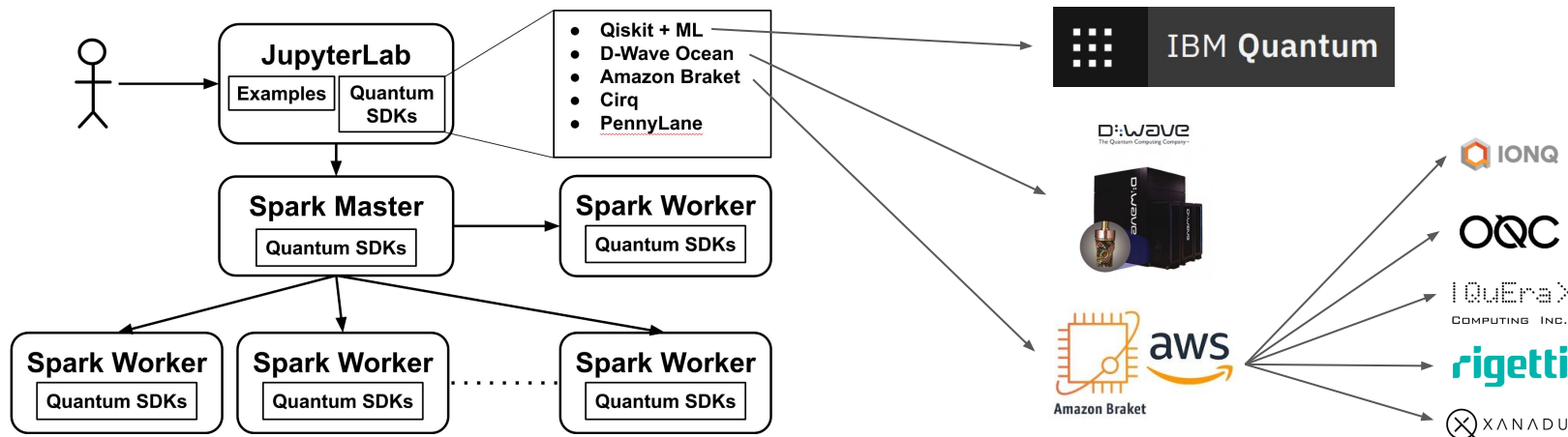
www.elkh.org

- Áttekintés - előzmények
- A referencia architektúra felépítése
- Támogatott erőforrások, keretrendszerek
- Kiegészítő könyvtárak
- Telepítési lehetőségek
- Vonatkozó nemzetközi kapcsolatok
- Egyetemi kapcsolatok
- Példa alkalmazások bemutatása:
 - Egyszerűbb példák
 - Kvantum-specifikus algoritmusok (pl. Deutsch-Jozsa)

- 2022-ben kezdtük el vizsgálni a kvantum erőforrások alkalmazhatóságát
- Először a D-Wave erőforrásaira fókuszálva, kiválasztott D-Wave példákat feldolgozva
- Idővel felmerült az igény az eredmények egy könnyen publikálható, demózható keretrendszerbe való integrálására
- Ebből született meg a kvantum referencia architektúra:
 - ELKH Cloud felhasználói számára
 - De természetesen más felhőben/lokálisan is telepíthető
 - Tartalmaz minden főbb fejlesztőeszközt/keretrendszert
 - Folyamatosan bővülő példákon keresztül segíti a kvantum erőforrások használatának megkezdését
- <https://science-cloud.hu/referencia-architekturak/quantum>



- Két fő konténerizált komponens: JupyterLab, Apache Spark klaszter
- JupyterLab:
 - Interaktív fejlesztői környezet
 - <https://science-cloud.hu/referencia-architekturak/jupyterlab-fejleszto-kornyezet>
- Apache Spark klaszter:
 - Data science feladatok futtatását támogató rendszer
 - <https://science-cloud.hu/referencia-architekturak/apache-spark-klaszter>

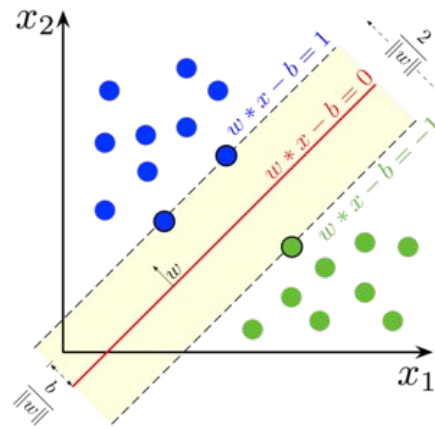
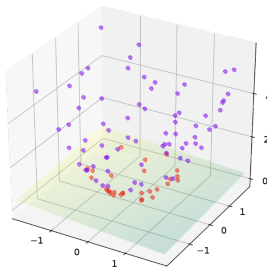
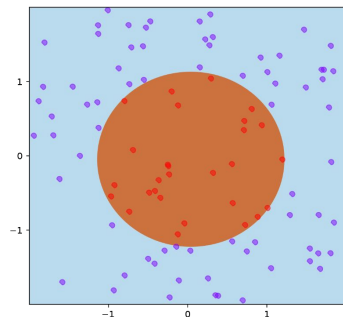


- D-Wave: Ocean SDK (6.2.0)
 - Gyakorlatilag a teljes D-Wave Leap spektrum
 - 2000Q, Advantage gépek
 - Hybrid solver-ek
- Amazon Braket SDK (1.35.3)
 - IonQ, Rigetti, Oxford Quantum Circuits, ...
 - <https://aws.amazon.com/braket/quantum-computers/>
- IBM Qiskit (0.39.5)
 - IBM Quantum előfizetési szinttől függően QPU-k
 - Ingyenesen 5 és 7 qubit-es QPU-k
- Google Cirq (1.1.0)
 - IonQ, Rigetti, Azure Quantum



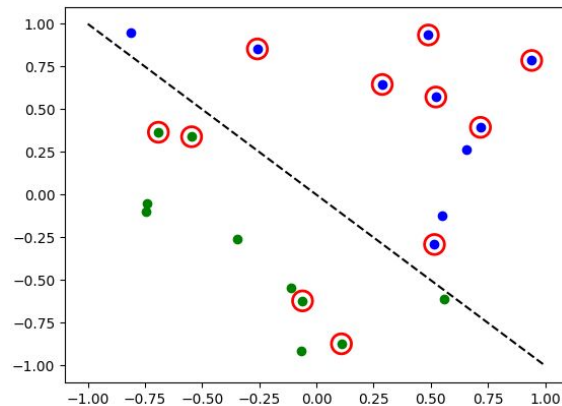
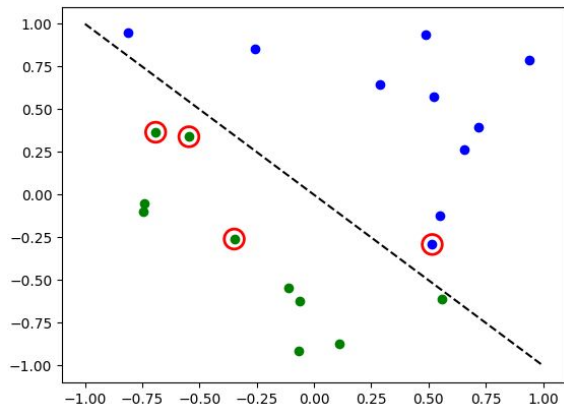
- Qiskit Machine Learning (0.5.0)
- Qiskit Runtime (0.8.0)
- Alapvető, gépi tanulást támogató számítási építőköveket biztosít, pl.:
 - Kvantum kernelek
 - Kvantum neurális hálók
- Ezek később olyan területeken alkalmazhatók, mint:
 - Klasszifikációs problémák
 - Regressziós feladatok
- A modul felépítése lehetővé teszi:
 - Gyors prototípus-építést mélyebb kvantum ismeretek nélkül is
 - Könnyű bővíthetőség további gépi tanulási algoritmusok támogatására
- Futtatási lehetőség kvantum erőforráson: Qiskit Runtime
- Példák: <https://qiskit.org/documentation/machine-learning/tutorials/index.html>

- Kvantum kernel: `FidelityQuantumKernel`,
`TrainableFidelityQuantumKernel`
 - Kernel célja az adatpontok magasabb dimenzióba történő konvertálása, a könnyebb megértés érdekében
 - Az adott osztály segíti adott adatpontokhoz tartozó kernel mátrixok könnyű kiszámítását
 - Ezek aztán klasszifikációs (QSVC) és regressziós (QSVR) algoritmusokban könnyedén felhasználhatóak



- Kvantum neurális hálók: általános interfész biztosított neurális hálók létrehozásához
- Implementációk:
 - EstimatorQNN: paraméterezett kvantumáramkörök kombinálhatóak kvantummechanikai jelenségek megfigyelésével
 - SamplerQNN: szintén paraméterezett kvantumáramköröket használva, bemeneti bitsztring valószínűségek meghatározására használható
- Neurális hálók taníthatóak:
 - NeuralNetworkClassifier: felhasználás klasszifikációhoz
 - NeuralNetworkRegressor: regresszióhoz

- Kapcsolat külső könyvtárhoz: TorchConnector
- A Qiskit által biztosított QNN-ek (kvantum neurális hálók) közvetlenül integrálhatóak a PyTorch-ba
- Pl.: klasszifikáció EstimatorQNN és SamplerQNN segítségével



- A PennyLane egy cross-platform könyvtár kvantumgépek programozásához
- A differenciálprogramozhatósága lehetővé teszi kvantumáramkörök végrehajtását és tanítását különböző backend rendszereken
- Alapvető koncepciója a gépi tanulást támogató keresztrendszerek és a kvantumszámítás összekapcsolása (NumPy, PyTorch, Tensorflow)
- Fő feladata a kvantumszámítások végrehajtása, és azok eredményének átadása a fenti klasszikus keretrendszereknek
- Igyekszik általános lenni: módosítások nélkül lehessen végrehajtani egy adott kódot akár több eszközön is
- Python csomag: `pennylane`
 - Referencia architektúrában telepítve

```
import pennylane as qml
from pennylane import numpy as np

# create a quantum device
dev1 = qml.device("default.qubit", wires=1)

@qml.qnode(dev1)
def circuit(phi1, phi2):
    # a quantum node
    qml.RX(phi1, wires=0)
    qml.RY(phi2, wires=0)
    return qml.expval(qml.PauliZ(0))

def cost(x, y):
    # classical processing
    return np.sin(np.abs(circuit(x, y))) - 1

# calculate the gradient
dcost = qml.grad(cost, argnum=[0, 1]) 10
```

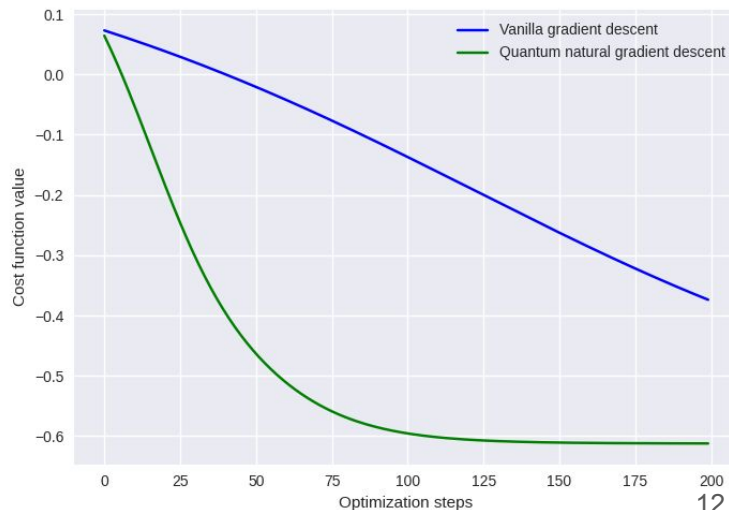
- Kvantum node:
 - Olyan függvény, ami kvantum áramkört definiál
 - Kapcsolódik hozzá egy végrehajtó eszköz
 - Definiálás: qnode dekorátor
- Példa:
 - 2 qubit-es eszköz létrehozása
 - Itt tudjuk megadni az akár fizikai eszközt is
 - Vannak beépített szimulátorok
 - Külső eszközök is megadhatók: Qiskit, IonQ, Amazon Braket, Rigetti, ...
 - Egy paramétert (x) használó függvény definiálása:
 - 0. qubit x fokú forgatása Z tengely körül
 - 0-1 CNOT kapu alkalmazása
 - 1. qubit x fokú forgatása Y tengely körül
 - Végül valamilyen mért érték(ek) visszaadása

```
dev = qml.device('default.qubit', wires=2)

@qml.qnode(dev)
def circuit(x):
    qml.RZ(x, wires=0)
    qml.CNOT(wires=[0,1])
    qml.RY(x, wires=1)
    return qml.expval(qml.PauliZ(1))

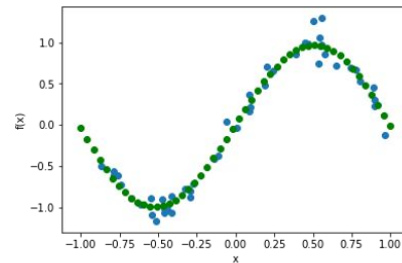
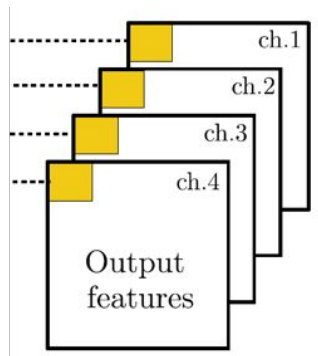
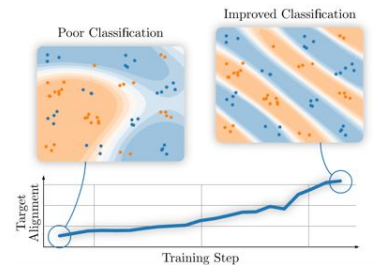
result = circuit(0.543)
```

- Áramkörök paraméterezhetők: variational circuits
- 3 fő részből állnak:
 - Adott fix alapállapot beállítása (pl. fix 0 státusz)
 - Egy paramétereket fogadó kvantumáramkör: $U(\theta)$, ahol θ a szabad paraméterek halmaza
 - Mérések (adott számú qubit alapján)
- Paraméterezés jelentősége:
 - Kvantumáramköröket tudunk integrálni klasszikus optimalizációs algoritmusokba
 - Az algoritmusok adott bemeneti paraméterek alapján mért értékekre tudnak reagálni
 - Majd újabb, finomhangolt paraméterekkel ismételtén tudják futtatni az algoritmust



- Milyen esetekben tudjuk a PennyLane eszközt alkalmazni:

- Kvantum kernelek - klasszifikációs problémákhoz
- Függvény-illesztések
- Konvolúciós neurális hálók támogatása
- ...
- https://pennylane.ai/qml/demos_qml.html



- Futtatáshoz számos backend elérhető
- Alap szimulátorok:
 - default.qubit, default.mixed, ..., lightning.qubit, lightning.gpu
 - Különböző tulajdonságokkal
 - Extra telepítést/konfigurációs nem igényelnek
 - Lokálisan (a Notebook-ot futtató környezetben) futnak
- Plugin-eken keresztül elérhető erőforrások:
 - Qiskit, Amazon Braket, Cirq, Strawberry Fields, IonQ, Rigetti, ...
 - Extra telepítésre, adott esetben konfigurációra, token megadásra szükség van
 - Valódi kvantum erőforrásokon történő futtatást tesznek lehetővé
- Részletes leírás: <https://pennylane.ai/plugins.html>

default.qubit A simple state-vector qubit simulator written in Python, with Autograd, JAX, TensorFlow, and Torch backends. A good choice for optimizations with a moderate number of qubits and parameters with exact expectation values.	default.mixed A mixed-state qubit simulator written in Python, with Autograd, JAX, TensorFlow, and Torch backends. A good choice for simulating noisy circuits and quantum channels.	default.gaussian A simple quantum photonic simulator written in Python. A good choice for optimizing photonic systems.
lightning.qubit A fast state-vector qubit simulator written with a C++ backend. A good choice for optimizations with a moderate number of qubits and parameters, or when using stochastic expectation values.	lightning.gpu A fast state-vector qubit simulator utilizing the NVIDIA cuQuantum SDK for GPU accelerated circuit simulation. A good choice for a large number of qubits and parameters, taking advantage of one or more GPUs for acceleration.	

Plugins

External quantum devices can be easily added to PennyLane by installing plugins. These plugins are installed separately, providing a rich ecosystem integrating popular quantum software development libraries with the hybrid optimization capabilities of PennyLane.

Below are a list of official PennyLane plugins supported by the PennyLane development team.

Qiskit Qiskit is an open-source quantum software framework designed by IBM. Supported hardware backends include the IBM Quantum Experience.	Amazon Braket Amazon Braket is a fully managed AWS service that helps researchers, scientists, and developers get started with quantum computing. Amazon Braket provides on-demand access to managed, high-performance quantum circuit simulators, as well as different types of quantum computing hardware.
---	--

- Jelenleg Docker Compose segítségével:
 - Alap: csak JupyterLab
 - Bővített: JupyterLab + Apache Spark klaszter
- Telepíthető:
 - Saját gépen (Docker Compose szükséges)
 - ELKH Cloud-ban (virtuális gépre)
 - Egyéb erőforráson
- További tervek:
 - Kubernetes alapú telepítés támogatása
- Leírás: <https://git.sztaki.hu/science-cloud/reference-architectures/quantum>

- Indítsunk egy virtuális gépet
 - Ubuntu 22.04
 - 4 GB RAM
 - 128 GB tárhely
 - Floating IP hozzárendelés
 - 22, 8888 TCP portok nyitva
- Lépünk be ssh-val a gépre

```
zfrakas@g14 ~]$ ssh -A ubuntu@193.225.250.224
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-50-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

System information as of Thu Feb 16 08:25:50 UTC 2023

System load:  0.01220703125   Processes:            185
Usage of /:   2.6% of 119.45GB Users logged in:       1
Memory usage: 7%             IPv4 address for docker0: 172.17.0.1
Swap usage:   0%             IPv4 address for enp3s0:  192.168.0.204

0 updates can be applied immediately.

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Thu Feb 16 08:21:36 2023 from 78.92.134.79
ubuntu@kvantum:~$
```

Overview

Interfaces

Log

Console

Action Log

Name

ID

Description

Project ID

Status

Locked

Availability Zone

Created

Age

Kvantum

27b408fd-3360-46c3-ae67-5a84b419a261

-

e117b9af702c402ea24ba61612a52666

Active

False

nova

Feb. 16, 2023, 8:20 a.m.

1 minute

Specs

Flavor

Not available

IP Addresses

default

192.168.0.204, 193.225.250.224

Security Groups

default

ALLOW IPv4 to 0.0.0.0/0

ALLOW IPv4 from default

ALLOW IPv6 from default

ALLOW IPv6 to ::/0

JupyterLab

ALLOW IPv4 to 0.0.0.0/0

ALLOW IPv4 8888/tcp from 0.0.0.0/0

16

- Kvantum RA Git repository klónozás
- Testreszabás (JupyterLab jelszó)
- Indítás

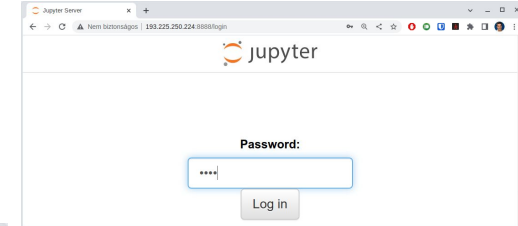
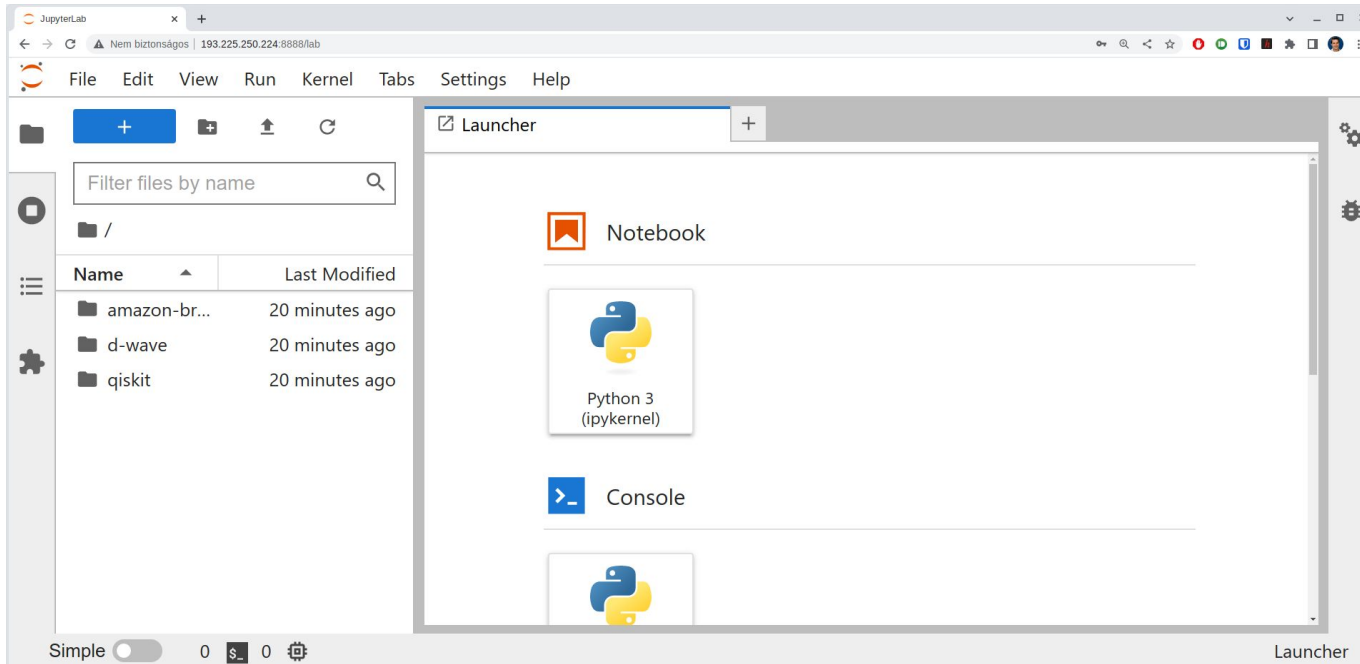
```
ubuntu@kquantum:~/quantum$ docker-compose up -d
Creating network "quantum_default" with the default driver
Pulling jupyterlab (git.sztaki.hu:5050/science-cloud/reference-architectures/quantum/jupyterlab:).
..
latest: Pulling from science-cloud/reference-architectures/quantum/jupyterlab
1efc276f4ff9: Pull complete
a2f2f93da482: Pull complete
1a2de4cc9431: Pull complete
9013b84ebbe7: Pull complete
7a935cd2c7d0: Pull complete
4a201854e60e: Pull complete
961b61ab0982: Pull complete
353e8ab774ba: Pull complete
555da64a4b75: Pull complete
6b546b036db0: Pull complete
d0dc67daa137: Pull complete
e68aa40faf9c: Pull complete
4f332f280780: Pull complete
c669cc129857: Pull complete
Digest: sha256:95440d87be98c30d12b1829050d82b6a0e7ca5a0842a16e11716b3e8f392d56e
Status: Downloaded newer image for git.sztaki.hu:5050/science-cloud/reference-architectures/quantum/jupyterlab:latest
Creating jupyterlab ... done
ubuntu@kquantum:~/quantum$ docker ps -a
```

CONTAINER ID	IMAGE	CREATED	STATUS	PORTS	COMMAND
4da957fc90fa	git.sztaki.hu:5050/science-cloud/reference-architectures/quantum/jupyterlab	2 minutes ago	Up 2 minutes	0.0.0.0:8888->8888/tcp, ::8888->8888/tcp	"/usr/local/bin/star..."

```
ubuntu@kquantum:~/quantum$
```

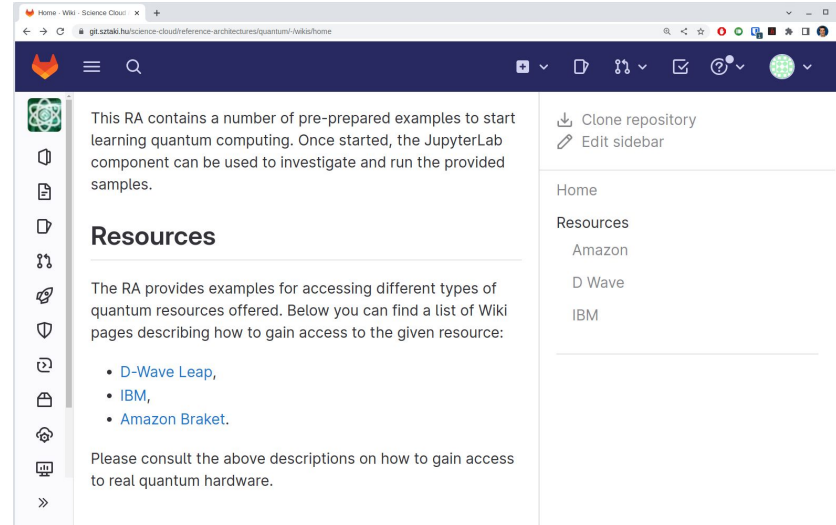
```
ubuntu@kquantum:~$ git clone https://git.sztaki.hu/science-cloud/reference-architectures/quantum.git
Cloning into 'quantum'...
remote: Enumerating objects: 364, done.
remote: Counting objects: 100% (215/215), done.
remote: Compressing objects: 100% (208/208), done.
remote: Total 364 (delta 107), reused 0 (delta 0), pack-reused 149
Receiving objects: 100% (364/364), 2.74 MiB | 16.99 MiB/s, done.
Resolving deltas: 100% (168/168), done.
ubuntu@kquantum:~$ cd quantum/
ubuntu@kquantum:~/quantum$ nano docker-compose.yml
ubuntu@kquantum:~/quantum$ cat docker-compose.yml
version: '3'
services:
  jupyterlab:
    image: git.sztaki.hu:5050/science-cloud/reference-architectures/quantum/jupyterlab
    container_name: jupyterlab
    ports:
      - "8888:8888"
    environment:
      - PASSWORD=demo
    volumes:
      - ./lab/notebook:/notebook
ubuntu@kquantum:~/quantum$
```

- Böngészőből: `http://<floating_ip>:8888`
 - Példánknál: <http://193.225.250.224:8888/>

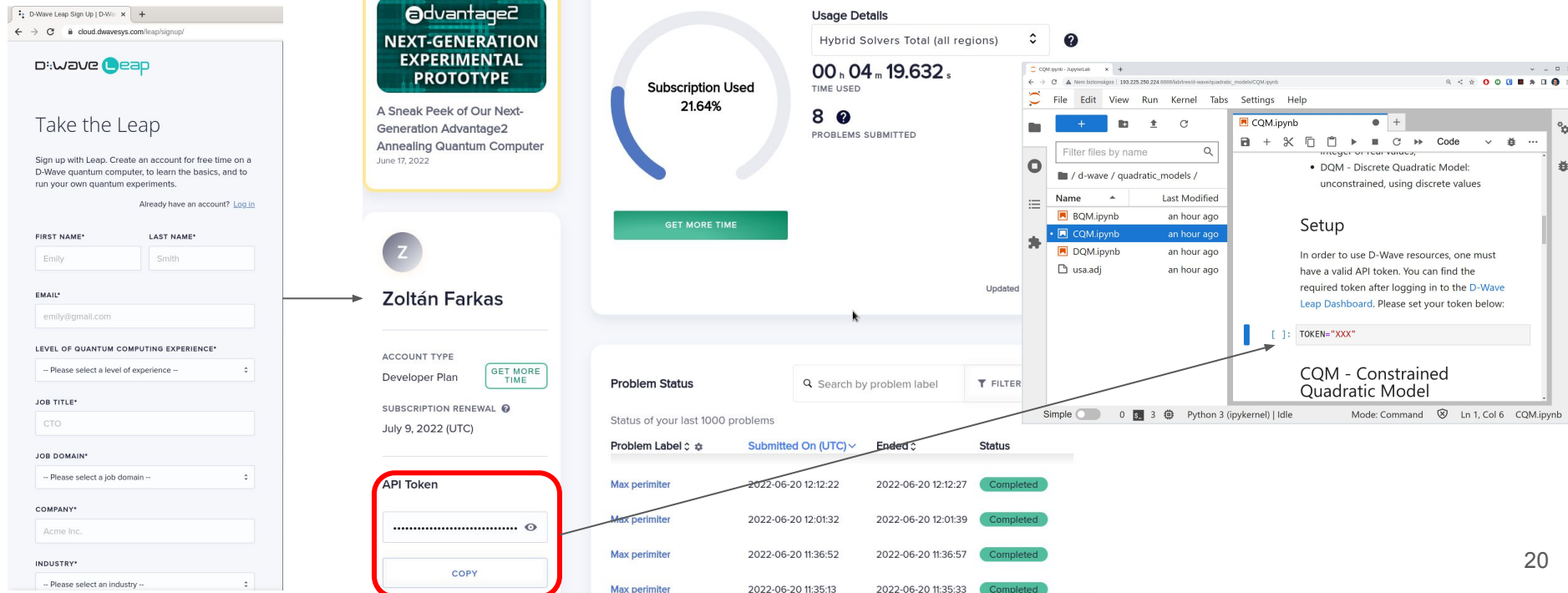


- Lokálisan használt szimulátorok használata nem igényel konfigurációt
- Felhő alapú szimulátorok, illetve valódi kvantum erőforrások viszont igen!
- Részletes leírás az erőforrások használatáról elérhető a referencia architektúra Wiki oldalon:

<https://git.sztaki.hu/science-cloud/reference-architectures/quantum/-/wikis/home>



- Regisztráció szükséges a D-Wave Leap rendszerbe
 - <https://cloud.dwavesys.com/leap/>



Take the Leap

Sign up with Leap. Create an account for free time on a D-Wave quantum computer, to learn the basics, and to run your own quantum experiments.

Already have an account? [Log in](#)

First Name* **Last Name***

EMAIL*

LEVEL OF QUANTUM COMPUTING EXPERIENCE*

JOB TITLE*

JOB DOMAIN*

COMPANY*

INDUSTRY*

Account Information:

Advantage2
NEXT-GENERATION EXPERIMENTAL PROTOTYPE

A Sneak Peek of Our Next-Generation Advantage2 Annealing Quantum Computer
June 17, 2022

Subscription Used
21.64%

[GET MORE TIME](#)

Usage Details

Hybrid Solvers Total (all regions)

00 h 04 m 19.632 s
TIME USED

8 PROBLEMS SUBMITTED

Problem Status

Status of your last 1000 problems

Problem Label	Submitted On (UTC)	Ended	Status
Max perimiter	2022-06-20 12:12:22	2022-06-20 12:12:27	Completed
Max perimiter	2022-06-20 12:01:32	2022-06-20 12:01:39	Completed
Max perimiter	2022-06-20 11:36:52	2022-06-20 11:36:57	Completed
Max perimiter	2022-06-20 11:35:13	2022-06-20 11:35:33	Completed

API Token

..... [COPY](#)

JupyterLab - CQM.ipynb

Filter files by name

Files:

- BQM.ipynb an hour ago
- CQM.ipynb an hour ago
- DQM.ipynb an hour ago
- usa.adj an hour ago

Setup

In order to use D-Wave resources, one must have a valid API token. You can find the required token after logging in to the [D-Wave Leap Dashboard](#). Please set your token below:

TOKEN= "xxx"

CQM - Constrained Quadratic Model

- Adott kvóta mellett használható
 - QPU idő: 1 perc
 - Hibrid solver-ek: 20 perc
- 1 hónapig ingyenesen
- Utána open-source kód megosztásával havonta újítható
- Bővebb igény esetén céges és kutatói megoldások elérhetőek

Usage Details

Advantage_system5.3



00 h 00 m 00.000 s
TIME USED

0 ?

PROBLEMS SUBMITTED

00 h 01 m 00.000 s ?
TIME AVAILABLE

Usage Details

hybrid_constrained_quadratic_mod... ▾

00 h 00 m 00.000 s
TIME USED

0 ?

PROBLEMS SUBMITTED

00 h 20 m 00.000 s ?
TIME AVAILABLE

Free Developer Access

If you make the code you develop for D-Wave solvers open-source, you can continue to get free time on the system every month.

Provide your GitHub details to qualify for Developer Access.

Service Details

- ✓ Flexible access to both D-Wave QPUs and hybrid solvers
- ✓ Continued access to Leap learning resources and community
- ✓ All software developed using this plan must be contributed to the open-source community

GET DEVELOPER ACCESS

Commercial Access

D-Wave's customers have developed over 250 early applications. Use D-Wave's cloud resources to develop your own.

Whether you have an early idea you want to explore, an enterprise-scale application already defined, or anything in between, contact us so that we can determine how to help you best.

Service Details

- ✓ Flexible access to both D-Wave QPUs and hybrid solvers
- ✓ Community and email support
- ✓ No obligation to open source

CONTACT US

Research and Education

Incorporate quantum computing in research and education.

Instructors and researchers with accredited organizations can qualify for discounted time on the system. Contact us so that we can discuss the available options.

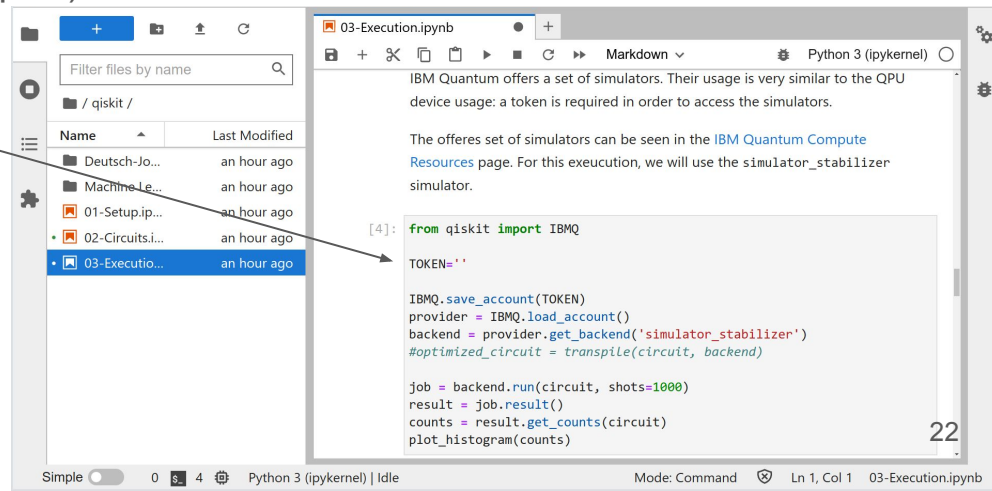
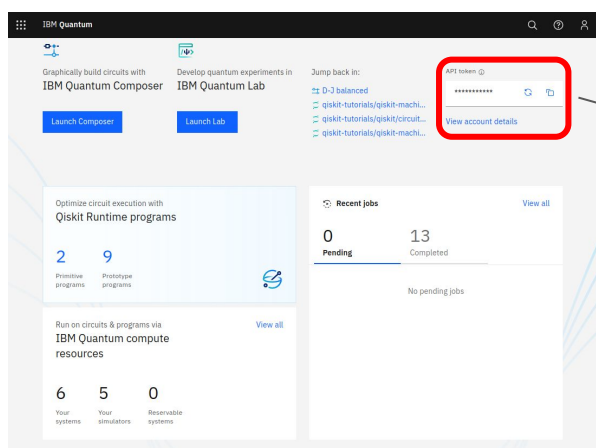
Service Details

- ✓ Flexible access to both D-Wave QPUs and hybrid solvers
- ✓ Community and email support
- ✓ Research discounts on commercial access plans
- ✓ Scientific and academic collaborations *

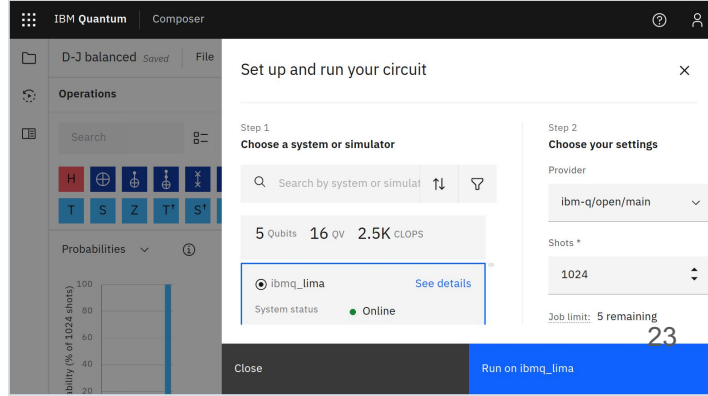
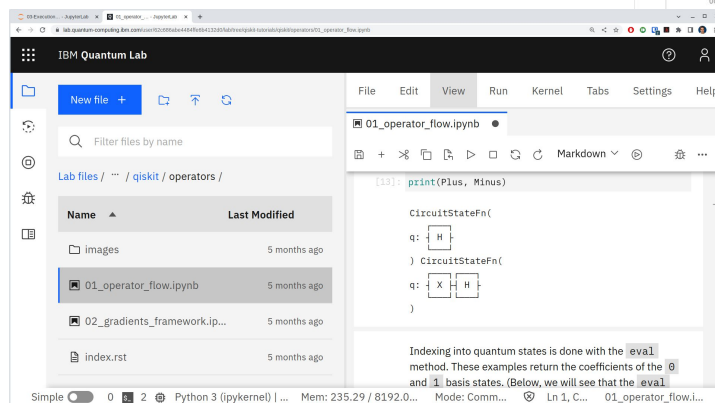
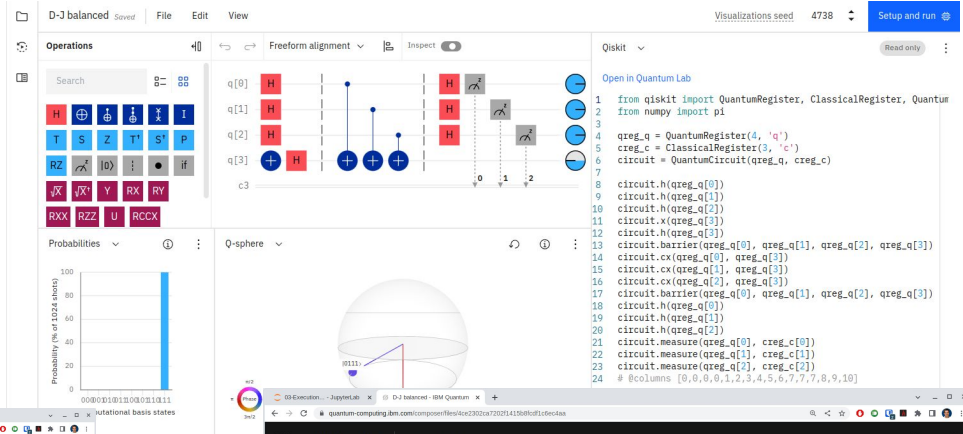
* Eligibility determined on a project-by-project basis.

CONTACT US

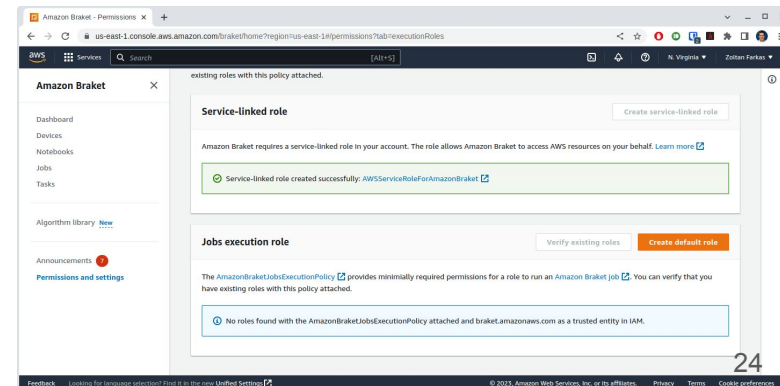
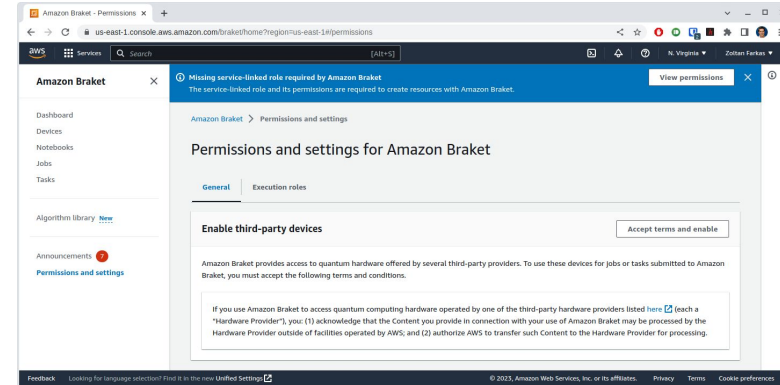
- Regisztráció szükséges az IBM Quantum oldalon:
 - <https://quantum-computing.ibm.com/>
 - Számos regisztrációs mód
- Három előfizetési kategória:
 - Ingyenes: felhő alapú szimulátorok, max. 7 qubit-es QPU-k
 - Pay as you go: ingyenes + 27 qubit-es QPU-k
 - Speciális: legújabb QPU-k elérése (127 qubit)



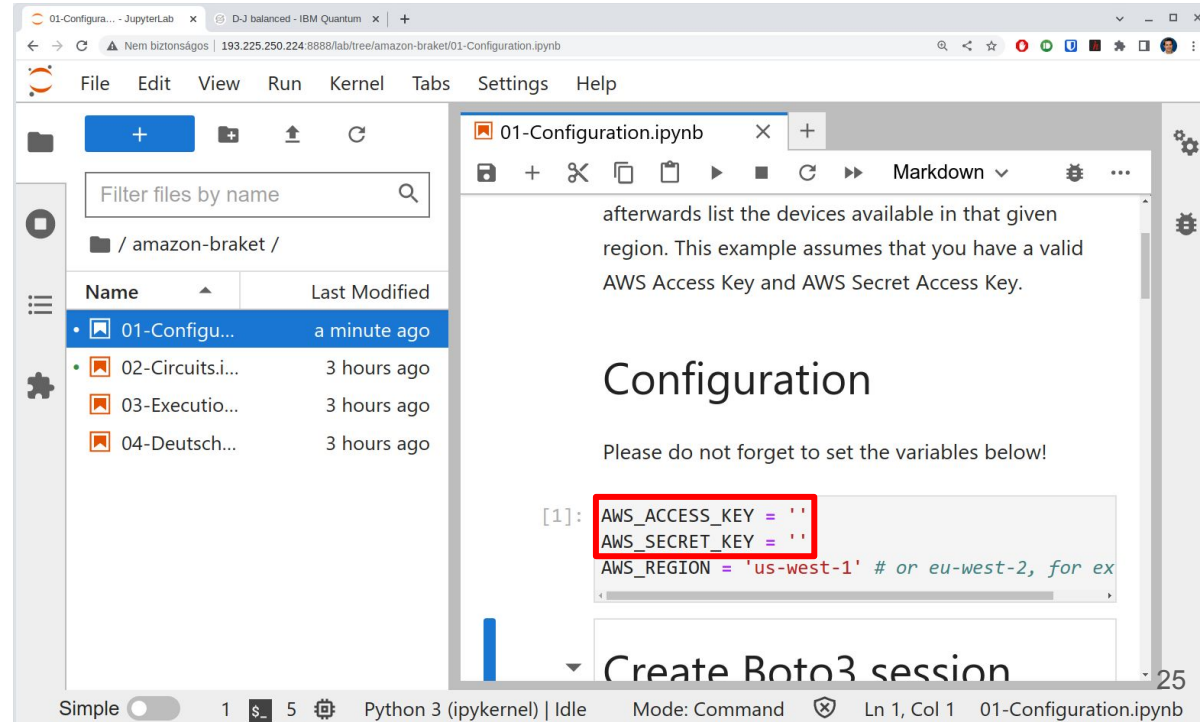
- Saját IBM Quantum szolgáltatások is elérhetőek
- IBM Quantum Composer
 - Grafikus áramkör szerkesztő
 - Kód generálás
 - Futtatás
- IBM Quantum Lab
 - Interaktív fejlesztői környezet
 - Integrált IBM példákkal



- Változatos QPU elérési lehetőségek:
 - IonQ, Rigetti, OQC, ...
- Többféle regisztrációs mód
- Free tier-ben ingyenes szimulátor használat
- Egyébként pay as you go séma
- Regisztráció után elvégzendő feladatok:
 - Third-party provider-ek használatának elfogadása
 - Service-hez kapcsolt role létrehozása
 - Kvantum job futtatását támogató role hozzáadása
- Elérés access key és secret key segítségével



- Access key és secret key szükséges
- AWS konzol “Identity and Access Management” (IAM) felületén kezelhetők
- Példákba közvetlenül beírhatók



01-Configura... - JupyterLab x D-3 balanced - IBM Quantum x +

Nem biztonságos | 193.225.250.224:8888/lab/tree/amazon-braket/01-Configuration.ipynb

File Edit View Run Kernel Tabs Settings Help

Filter files by name

/ amazon-braket /

Name	Last Modified
• 01-Configu...	a minute ago
• 02-Circuits.i...	3 hours ago
03-Executio...	3 hours ago
04-Deutsch...	3 hours ago

01-Configuration.ipynb

afterwards list the devices available in that given region. This example assumes that you have a valid AWS Access Key and AWS Secret Access Key.

Configuration

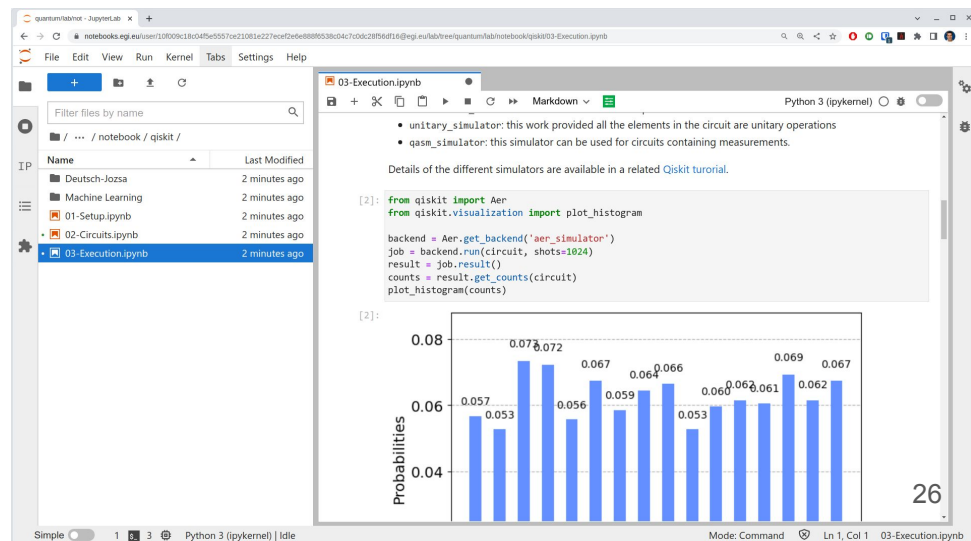
Please do not forget to set the variables below!

```
[1]: AWS_ACCESS_KEY = ''  
     AWS_SECRET_KEY = ''  
     AWS_REGION = 'us-west-1' # or eu-west-2, for ex
```

Create Boto3 session

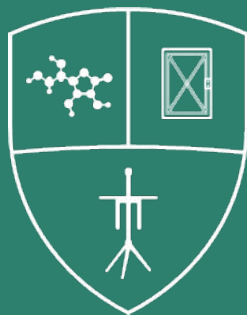
Simple 1 5 Python 3 (ipykernel) | Idle Mode: Command Ln 1, Col 1 01-Configuration.ipynb

- <https://notebooks.egi.eu/>
- Jupyter Notebook szolgáltatás pár kattintással indítható
- Federált hozzáférés a szolgáltatáshoz
- Főbb kvantum SDK-k előtelepítve:
 - D-Wave Ocean SDK
 - Amazon Braket
 - Qiskit (+ML)
 - PennyLane
- Példákat nem tartalmazza, viszont a kvantum referencia architektúra Git repository-ból letölthetőek



- Szakdolgozatok:
 - Óbudai Egyetemen három szakdolgozat készül a témában
 - A referencia architektúrák példáinak bővítése
 - Algoritmusok teljesítményvizsgálata
 - További erőforrások támogatásának megvalósítása
- Tantárgy: Bevezetés a kvantumprogramozásba - Óbudai Egyetem
 - Cél: betekintést adni az elérhető kvantum erőforrások körébe, azok programozásába, alkalmazásuk lehetőségeibe
 - Modellek bemutatása
 - D-Wave, Amazon Braket, IBM Qiskit, Google Cirq, Quantum Machine Learning

- Telepítés az ELKH Cloud-on
- Alap kvantum példák:
 - Qiskit: áramkör építés és végrehajtás
 - Amazon Braket: egyszerűbb végrehajtás
 - D-Wave: CQM végrehajtás
 - D-Wave: gráf minimális csúcslefedés meghatározása
- Kvantum-specifikus algoritmus példa:
 - Deutsch-Jozsa algoritmus - Qiskit
- Demo telepítés: <http://193.225.250.172:8888/>



ELKH

Eötvös Loránd
Research Network

Köszönöm a figyelmet!

www.elkh.org