

Udhëzues për Përdorimin e Google Colab në Kursin “Modellimi në Fizikë”

Klaudio Peqini
Departamenti i Fizikës

02 Mars, 2026

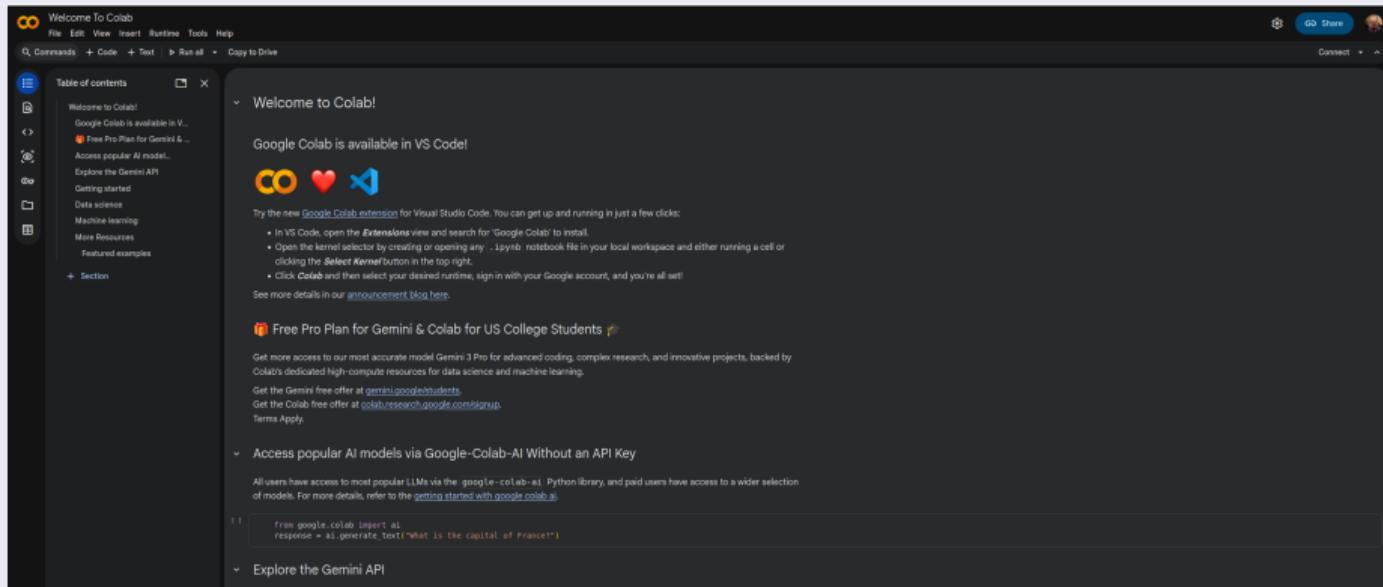
Çfarë është Google Colab?

- Google Colab është një mjedis ekzekutimi për **Jupyter Notebooks** në cloud.
- Nuk kërkon instalim lokal të Python-it.
- Lejon ruajtjen automatike në Google Drive.
- Është veçanërisht i përshtatshëm për fillimin e punës në kurs.

Hapi 1: Hyrja në platformë

- Hapni shfletuesin (Chrome, Firefox, etj.).
- Viziton: <https://colab.research.google.com>
- Hyni me llogarinë tuaj Google.

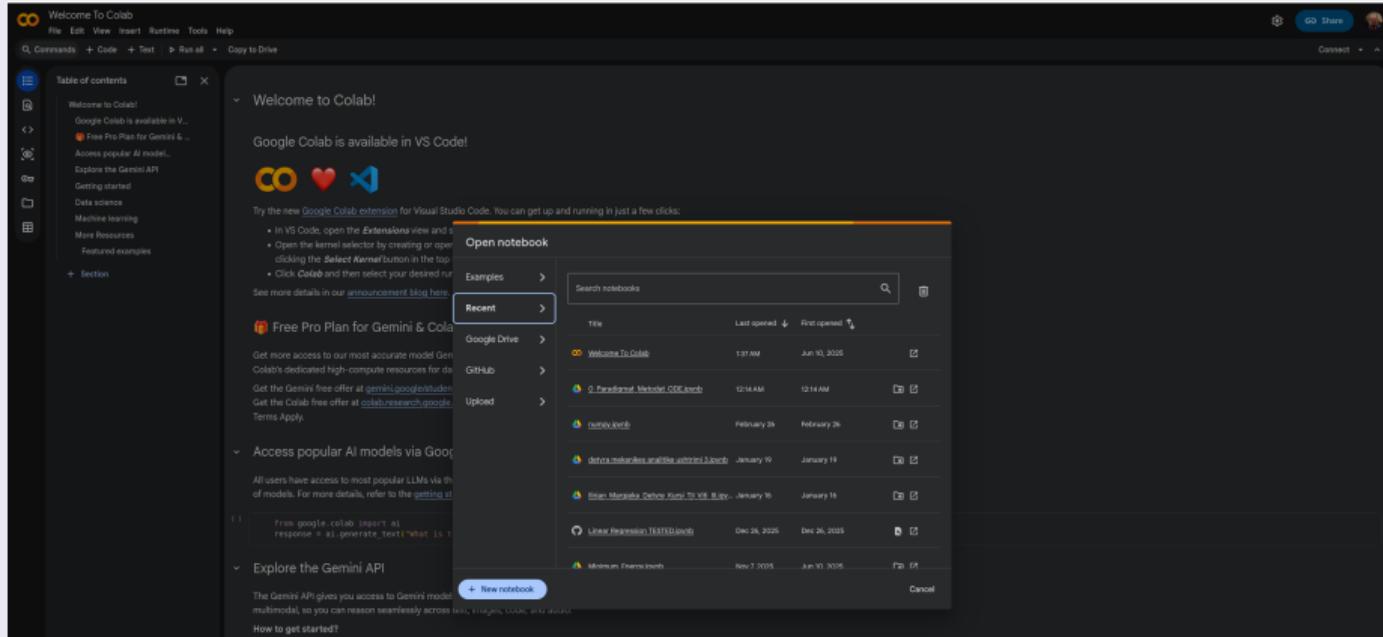
Vendosni screenshot këtu



Hapi 2: Hapja e një Notebook-u

- Zgjidhni “File” → “Upload notebook” për të ngarkuar një skedar .ipynb.
- Ose “GitHub” për të hapur notebook direkt nga repository i kursit.

Vendosni screenshot këtu



Hapi 3: Ekzekutimi i qelizave

- Çdo qelizë kodi ekzekutohet me butonin ▶.
- Mund të përdorni edhe shkurtoren: Shift + Enter.
- Ekzekutimi bëhet në një mjedis virtual në cloud.

Vendosni screenshot këtu

The screenshot shows a Google Colab notebook titled "0_Paradigmat_MetodatODE.ipynb". The code cell contains Python code for solving differential equations using numerical methods. It includes imports for numpy and scipy.optimize, defines a function for relaxation simulation, and uses it to solve for different initial conditions. The code also includes optimization logic to find the minimum of a function related to the simulation results.

```
# (a) Simlioni EDE e shpejtshë d x / dt = -k x (relaksim)
def simulate_relaxation(x0, k, t_end=0.5, dt=0.01):
    t = np.arange(0.0, t_end+dt, dt)
    x = np.zeros_like(t)
    x[0] = x0
    for n in range(len(t)-1):
        x[n+1] = x[n] + dt * (-k * x[n]) # Euler
    return t, x

t, x = simulate_relaxation(k=0.2, x0=1.0)
x[-1], x[0]

# (b) Optimizimi: gjej k që përshtatet me "të dhëna" sintetike të relaksimit
# Kjo është një ilustrim minimal; më vonë do t'ë përmirësojme më i mirë (p.sh. scipy.optimize)

def loss_for_k(k, t_data, x_data, dt):
    t_sim, x_sim = simulate_relaxation(k=k, x0=x0, t_end=t_data[-1], dt=dt)
    return float(np.mean(np.abs(x_sim[-len(x_data):] - x_data)**2))

# Të dhëna rastësish
rng = np.random.default_rng()
t_data = np.arange(0, 5.0+0.01, 0.01)
k_true = 1.5
x_data = simulate_relaxation(k=k_true, x0=1.0, t_end=5.0, dt=0.01)
x_data = x_true + 0.02 * rng.standard_normal(len(x_true)) # dhurrave

# korkim i objektit se projekti
k_grid = np.linspace(0.5, 2.0)
losses = [loss_for_k(k, t_data, x_data, dt=0.01) for k in k_grid]
k_hat = k_grid[int(np.argmin(losses))]

x_true, k_hat, min(losses)
```

Hapi 4: Instalimi i paketave (nëse kërkon)

Në rast se një notebook kërkon biblioteka shtesë:

- Përdorni: !pip install emri_paketes
- Ose krijoni një qelizë të dedikuar për instalim në fillim të notebook-ut.

Shembull

```
!pip install numpy matplotlib
```

Hapi 5: Ruajtja dhe shkarkimi

- Notebook-u ruhet automatikisht në Google Drive.
- Për shkarkim lokal:
 - File → Download → Download .ipynb
 - Ose Download .pdf për raportim.

Rekomandime pedagogjike

- Përdorni Colab për eksplorim fillestare.
- Kaloni gradualisht në strukturë “production” në repository lokal.
- Mos e trajtoni notebook-un si kod final, por si laborator.

Google Colab është mjet hyrës për të filluar shpejt.

Qëllimi i kursit mbetet:

Nga Notebook → Tek Strukturë Profesionale → Tek Mendim Kërkimor