

# ITEC

## CS-300 Artificial Intelligence

### Lab - Decision Tree

### Jupyter Lab & Decision Tree

***Dr. Nguyen Ngoc Thao***

***Msc. Nguyen Hai Dang***

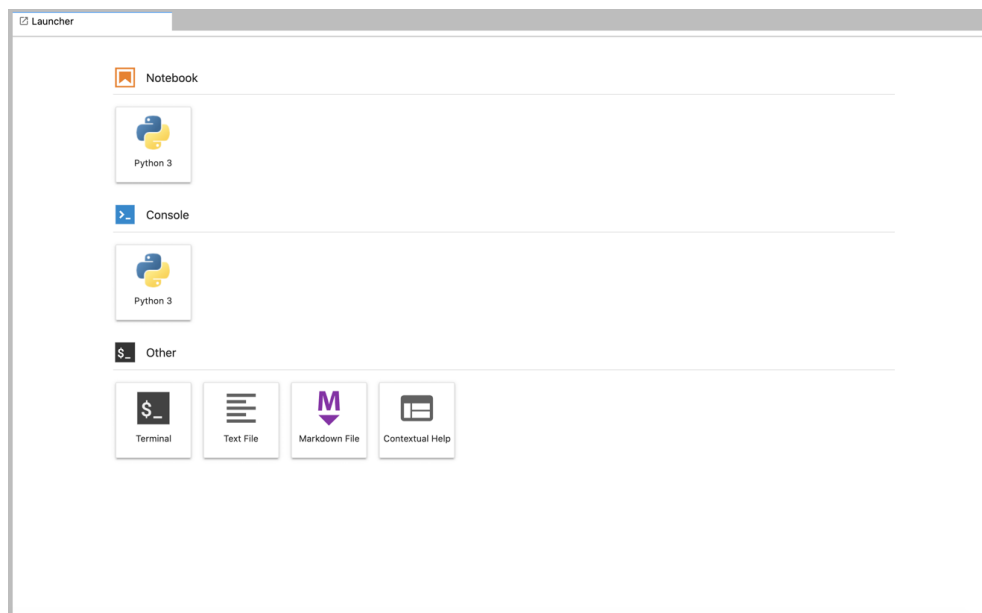
***Msc. Do Trong Le***

***Mr. Nguyen Quang Thuc***

## Giới thiệu về Jupyter Lab

Jupyter Lab là một môi trường phát triển trên nền tảng tương tác dạng web (web-based interactive development environment). Jupyter Lab cho phép người dùng có thể viết code python, chạy lệnh (command line), viết báo cáo (documentation) một cách tiện lợi.

- ❖ Website: <https://jupyter.org/>
- ❖ Hướng dẫn cài đặt Jupyter Lab:
  - Có thể dùng conda / pip
  - <https://jupyter.org/install>
- ❖ Để khởi động Jupyter Lab, chạy command sau trong terminal:
  - “jupyter lab” hoặc “jupyter notebook”
- ❖ Giao diện Jupyter Lab sẽ xuất hiện trong web browser như bên dưới:



- ❖ Jupyter Lab cho phép tạo Notebook, có thể chạy được python bên trong, chọn vào icon như bên dưới để tạo một notebook mới:



- ❖ Trong một notebook, ta có thể tạo ra các blocks bao gồm 2 loại chính
  - Python code block (executable block)
  - Markdown text block (syntax: <https://www.markdownguide.org/basic-syntax/>)
- ❖ Lưu ý:
  - Python Jupyter Notebook sử dụng các thư viện được cài đặt trong **môi trường của conda**. Do đó, nếu muốn import thêm các thư viện khác, ta phải cài đặt vào trong môi trường conda trước.
  - Các code block trong một notebook có thể được execute theo bất kỳ thứ tự nào, tuy nhiên logic code sẽ theo đúng thứ tự các block được execute.

## Decision Tree on MNIST

Input: ảnh (28x28) pixels (ảnh grayscale)

Output: loại chữ số (0-9)



Hình 1. Ví dụ về ảnh trong tập dữ liệu MNIST

Sử dụng duy nhất tập dữ liệu MNIST.

Link tham khảo dùng keras.datasets để [load tập MNIST](#).

Gợi ý sử dụng thư viện sklearn:

1. Import mô hình DecisionTreeClassifier có sẵn trong thư viện sklearn:

```
from sklearn.tree import DecisionTreeClassifier
```

2. Tạo một instance của mô hình:

```
model = DecisionTreeClassifier()
```

3. Gọi hàm fit để huấn luyện mô hình dựa vào dữ liệu có sẵn với input là X và label là y:

```
model.fit(X, y)
```