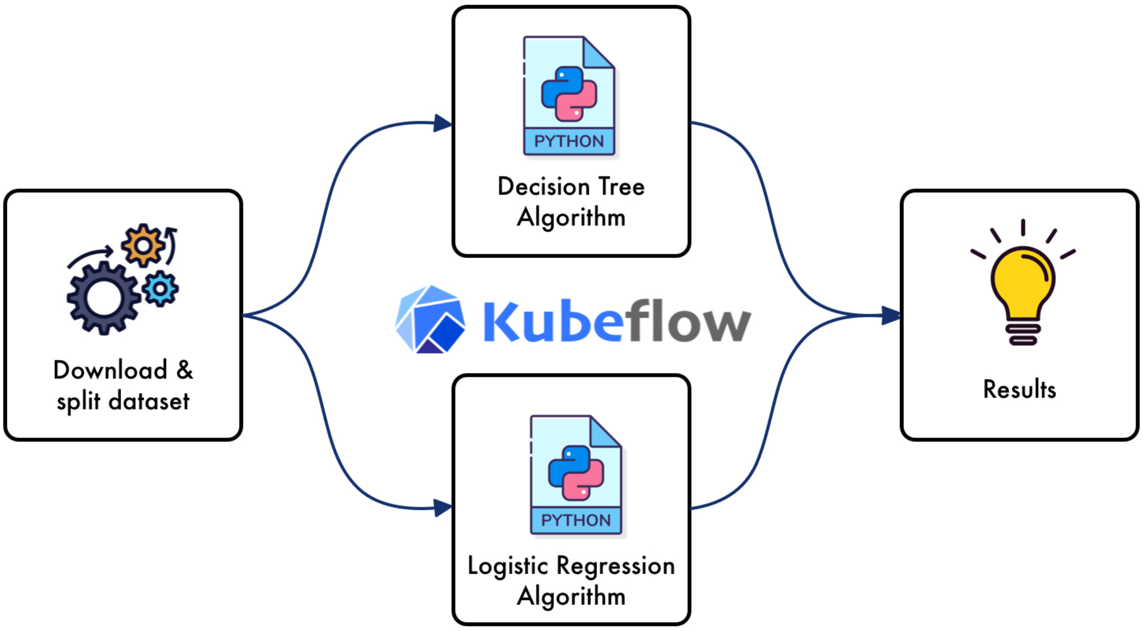
# Kubeflow Pipelines: How to Build your First Kubeflow Pipeline from Scratch

This tutorial aims to develop a step-by-step tutorial on how to build a Kubeflow Pipeline from scratch in your local machine.

If you want to know in detail about the detailed explanation of how to develop your first kubeflow pipeline, I recommend you take a look at the **kubeflow** home page.



## 1. Files

- **decision_tree**: Contains the files to build the decision_tree component as well as the Dockerfile used to generate the component image.
- **logistic_regression**: Contains the files to build the logistic_regression component as well as the Dockerfile used to generate the component image.
- **download_data**: Contains the files to build the load_data component as well as the Dockerfile used to generate the component image.
- **diabetes_pipeline.py**: Contains the definition of the pipeline, which when executed generates the **diabetes_pipeline.yaml** file.

# 2. How to use

It is recommended to have previously installed **kfp** as well as configured kubeflow on top of kubernets or a minimal version such as **kind** or **minikube**. Morover, the project has been developed under **kfp 1.8.19**. If you have some installed a newer version, some of decorators could not work properly. In this latter case, you can install the right version by running:

```
pip install 'kfp==1.8.19' --force-reinstall
```

After you have installed all the requirements, you should prepare the pipeline by running the following commands:

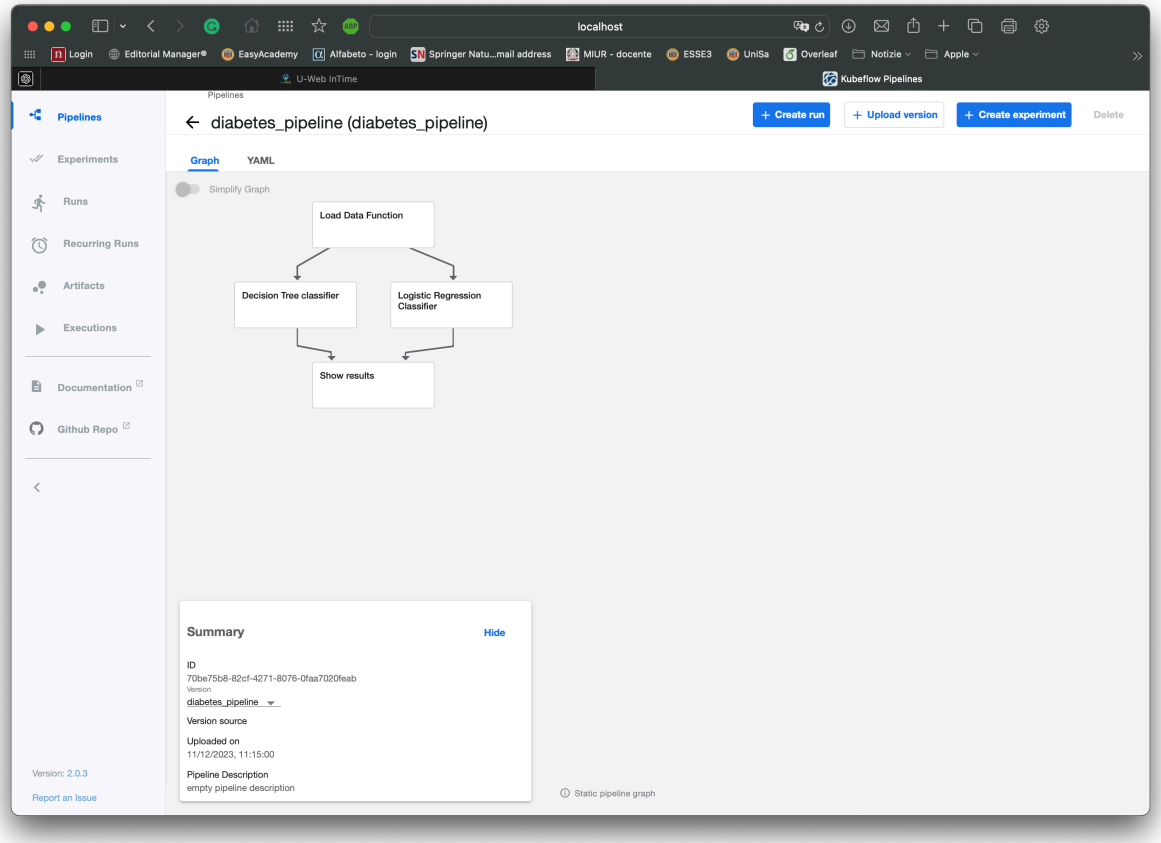
```
docker build --tag load_data_v3 .
docker tag load_data_v3 adellacioppa/load_data_v3
docker push docker.io/adellacioppa/load_data_v3

docker build --tag decision_tree_v3 .
docker tag decision_tree_v3 adellacioppa/decision_tree_v3
docker push docker.io/adellacioppa/decision_tree_v3

docker build --tag logistic_regression_v3 .
docker tag logistic_regression_v3 adellacioppa/logistic_regression_v3
docker push docker.io/adellacioppa/logistic_regression_v3
```

Then, to create the **yaml** file for your pipeline, you should run

```
python diabetes_pipeline.py
```

and you get the file **diabetes_pipeline.yaml**.

Finally, access to kubeflow dashboard and upload the the file **diabetes_pipeline.yaml** and create a run.



diabetes_pipeline (diabetes_pipeline)

Summary

ID
70be75b8-82cf-4271-8076-0faa7020feab

Version
diabetes_pipeline

Version source

Uploaded on
11/12/2023, 11:15:00

Pipeline Description
empty pipeline description