How to build a financial model with Kubeflow

葉信和 / Hsin-Ho Yeh
Software Engineer / CEO @ 信誠金融科技
hsinho.yeh@footprint-ai.com

信誠金融科技股份有限公司
XINCHEN FINTECH CO., LTD.

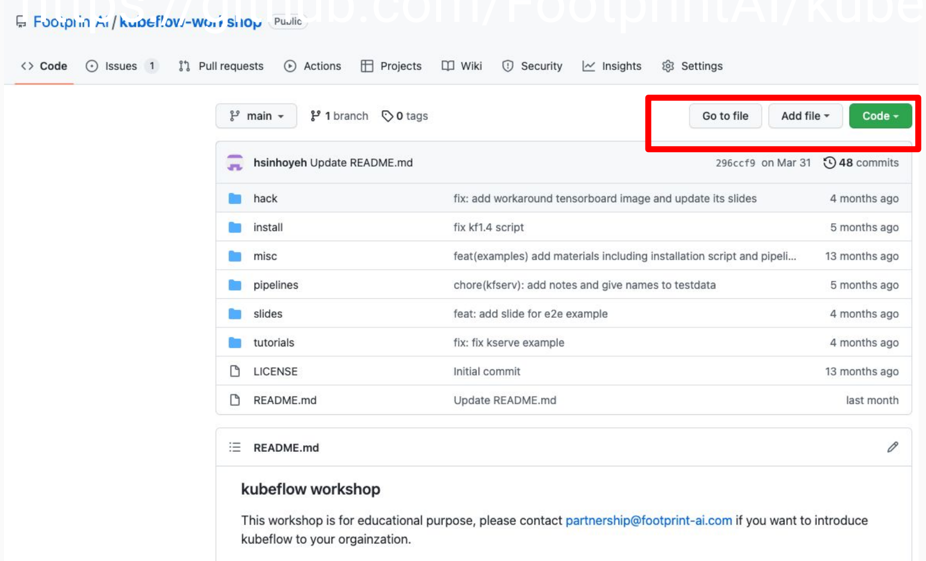
2022/05/12
2022/05/19

Download Slides

https://reurl.cc/yrkGrE

Materials

```
git clone
https://github.com/FootprintAI/kubeflow-workshop
```


FootprintAI / kubeflow-workshop Public

hsinhoyeh Update README.md 296ccf9 on Mar 31 48 commits

hack	fix: add workaround tensorboard image and update its slides	4 months ago
install	fix kf1.4 script	5 months ago
misc	feat(examples) add materials including installation script and pipeli...	13 months ago
pipelines	chore(kfserv): add notes and give names to testdata	5 months ago
slides	feat: add slide for e2e example	4 months ago
tutorials	fix: fix kserve example	4 months ago
LICENSE	Initial commit	13 months ago
README.md	Update README.md	last month

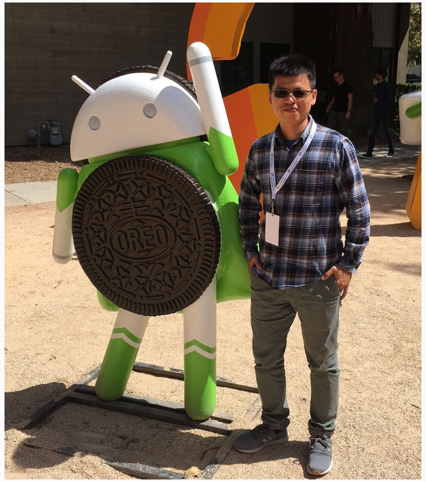
README.md

kubeflow workshop

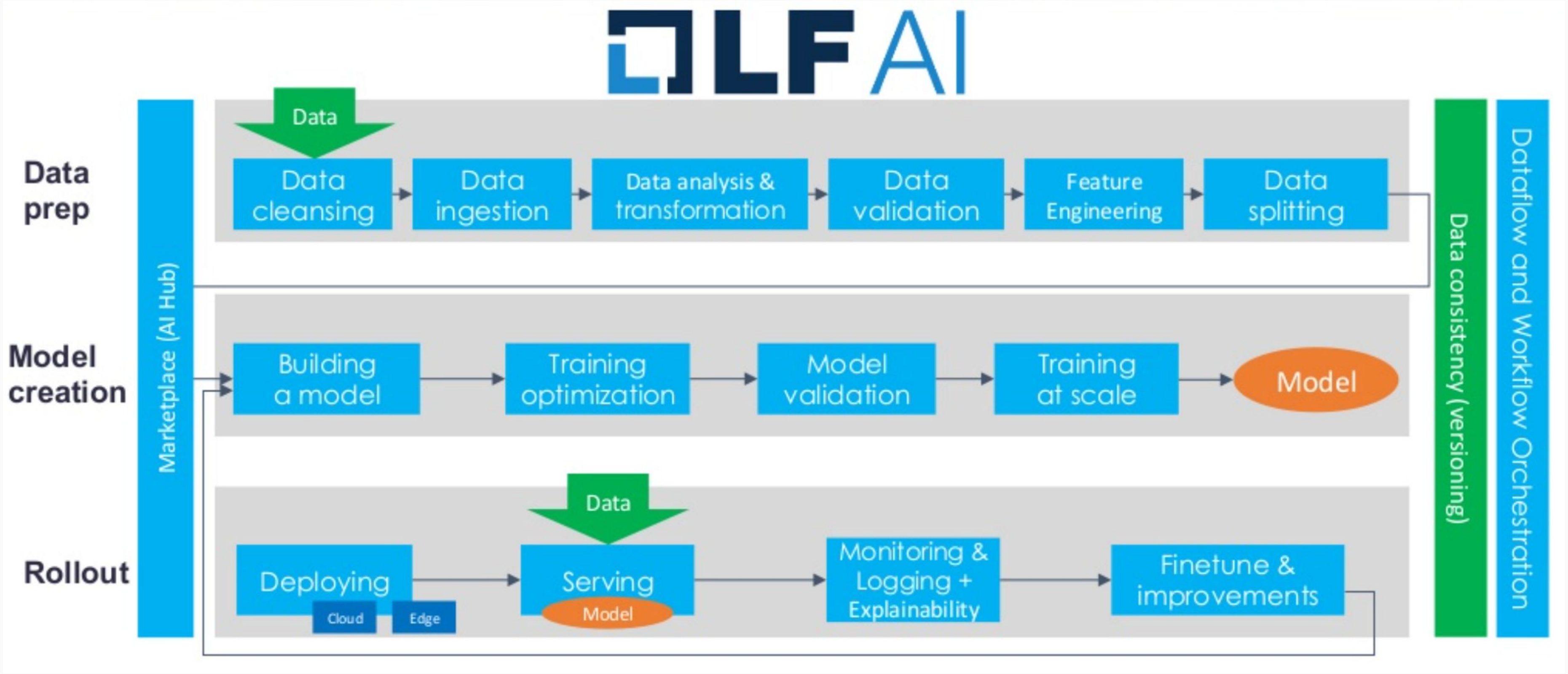
This workshop is for educational purpose, please contact partnership@footprint-ai.com if you want to introduce kubeflow to your orgainzation.

About me

- 2020 - Present at 信誠金融科技
  - Shrimping: A data-sharing platform
    - https://get-shrimping.footprint-ai.com
  - Tintin: a machine learning platform for everyone
    - https://get-tintin.footprint-ai.com
- 2016 - 2020 at IglooInsure (16M+ in series A+ 2020)
  - Provide digital insurance for e-conomic world
  - Funded in KUL, Headquartered in Singapore
  - First employee/ Engineering Lead / Regional Head/ Chief Engineer
- 2013 - 2016 at Studio Engineering @ hTC
  - Principal Engineer on Cloud Infrastructure Team
- 2009 - 2012 at IIS @ Academia Sinica
  - Computer vision, pattern recognition, and data mining
- CS@CCU, CS@NCKU alumni

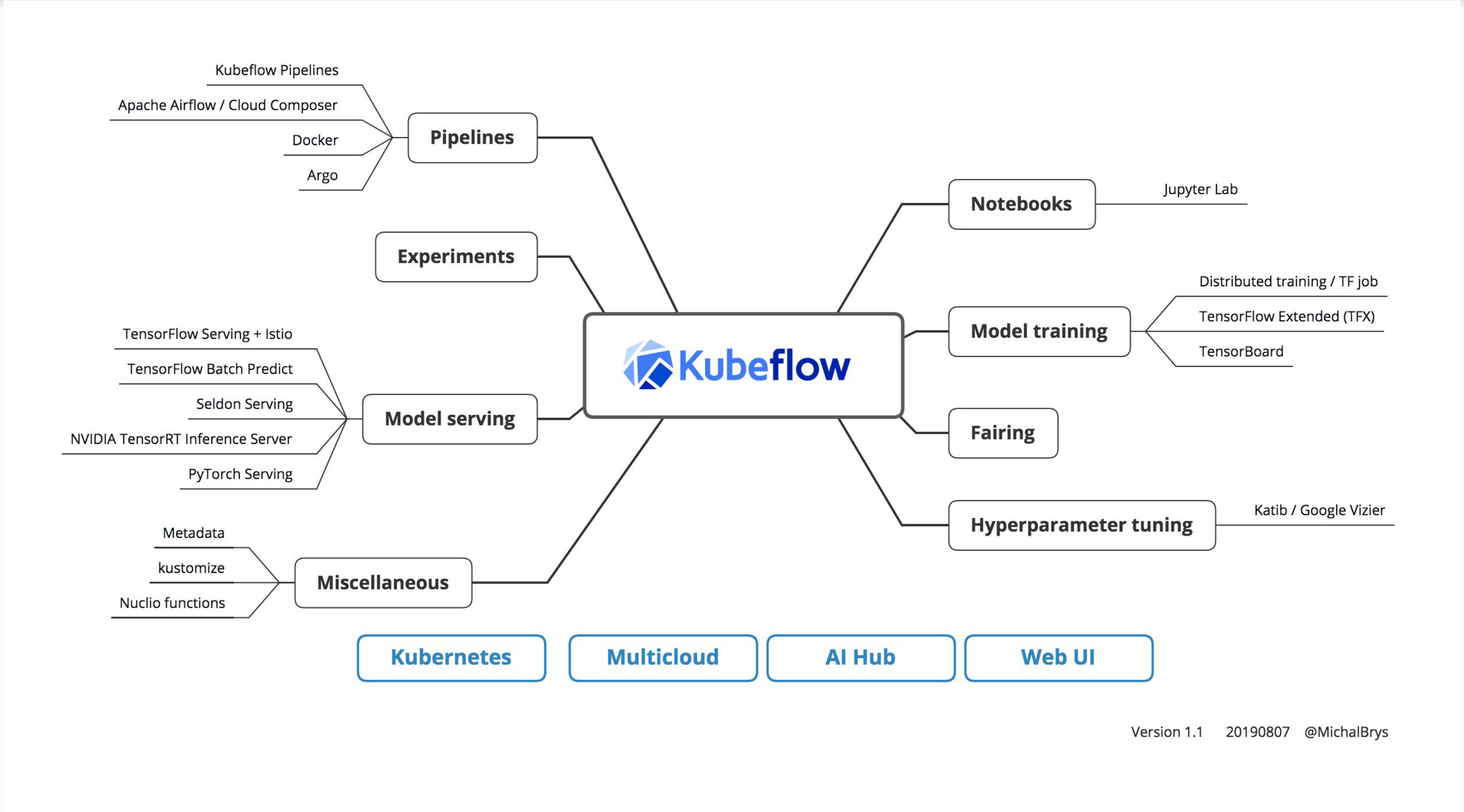


Real-world Machine Learning Application - End-to-End ML LifeCycle



Source: https://www.slideshare.net/AnimeshSingh/advanced-model-inferencing-leveraging-kubeflow-serving-knative-and-istio-196096385

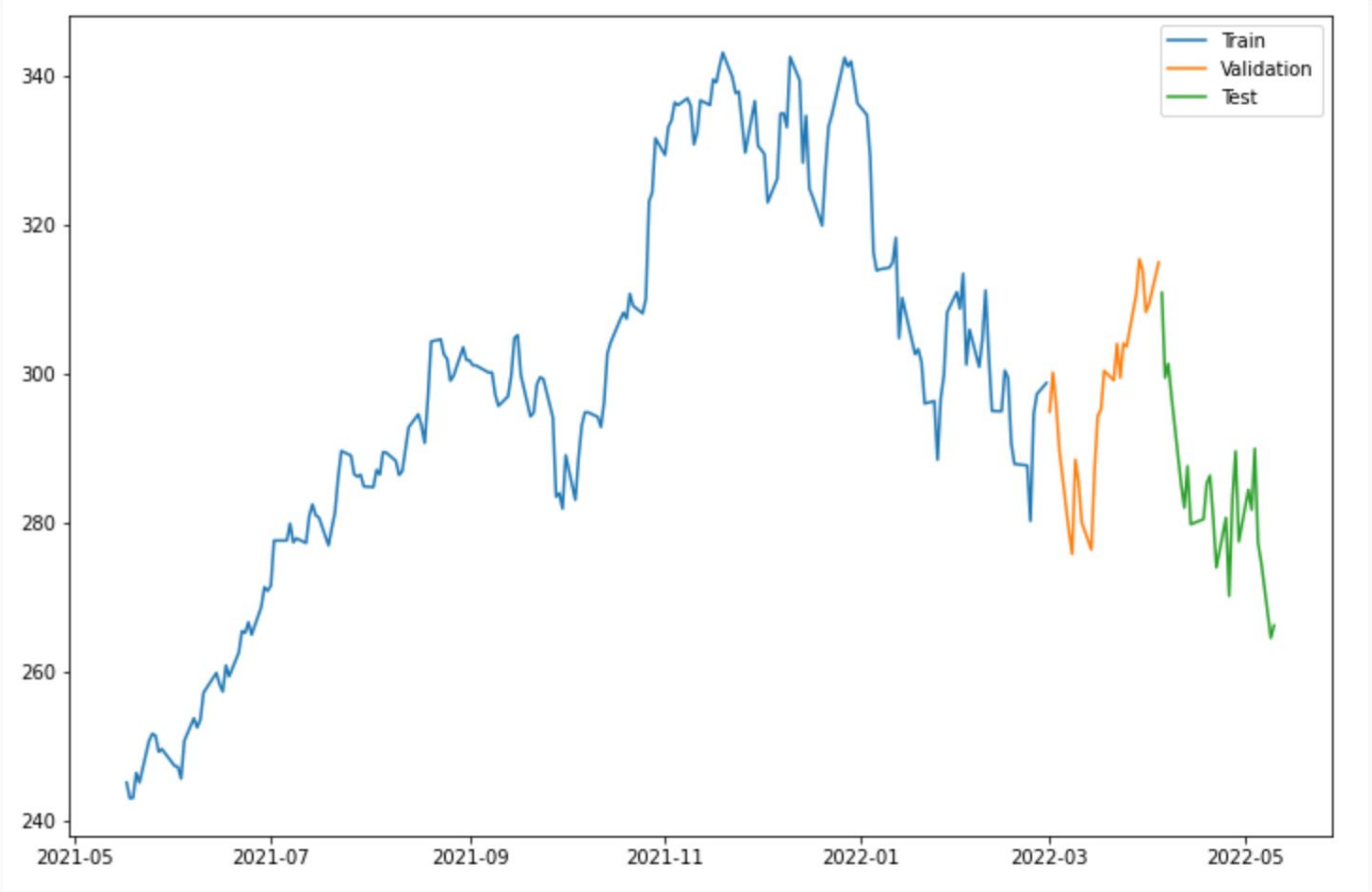
Kubeflow Architecture



Kubeflow In Financial Fields

Can LSTM model be used to predict the stock price?

How to predict stock price for the next day?



What sequences means in a series?

Input: [300, 320, 310, 350, 390]

t0	300
t1	320
t2	310
t3	350
t4	390
t5	?

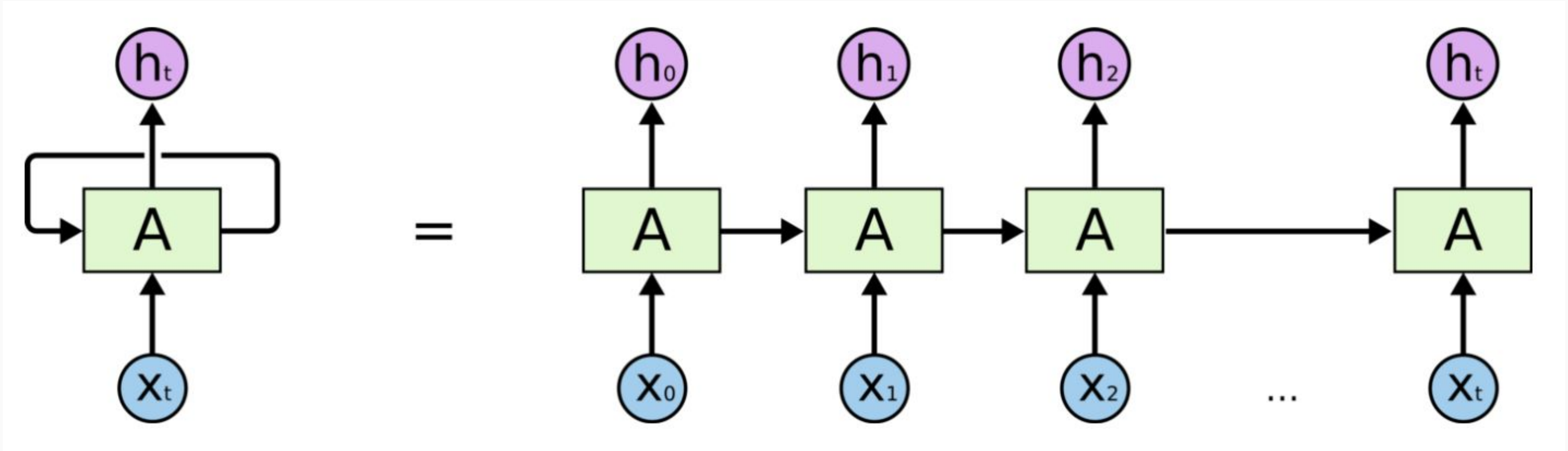
What sequences means in a sentence?

Input: I am a good guy

t0	I
t1	am
t2	a
t3	good
t4	guy
t5	?

What Is RNN (Recurrent Neural Network)?

Recurrent Neural Network (RNN) takes decisions on CURRENT (Xt) and PREVIOUS (Xt-1) inputs. Especially useful in topics including machine translation, speech recognition.



Recurrent Neural Network

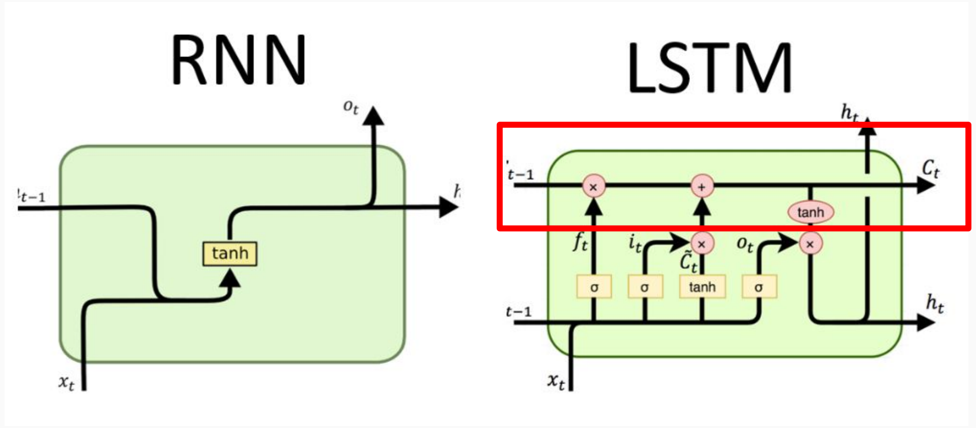
https://towardsdatascience.com/introduction-to-recurrent-neural-network-27202c3945f3

What Is LSTM (Long-short term memory)?

RNN only remember the latest things from X and it didn't remember(no memory) anything before at the beginning.

LSTM provides an information highway to let the neuron to selectively choose

1. forget from its memory (focus on the current inputs)
2. Listens to what information it added into memory (though information highway)



https://towardsdatascience.com/introduction-to-recurrent-neural-network-27202c3945f3
https://www.quora.com/How-is-LSTM-different-from-RNN-In-a-layman-explanation

How to formulate a sequence into a trainable dataset?

t0	t1	t2	t3	t4
300	320	310	350	390

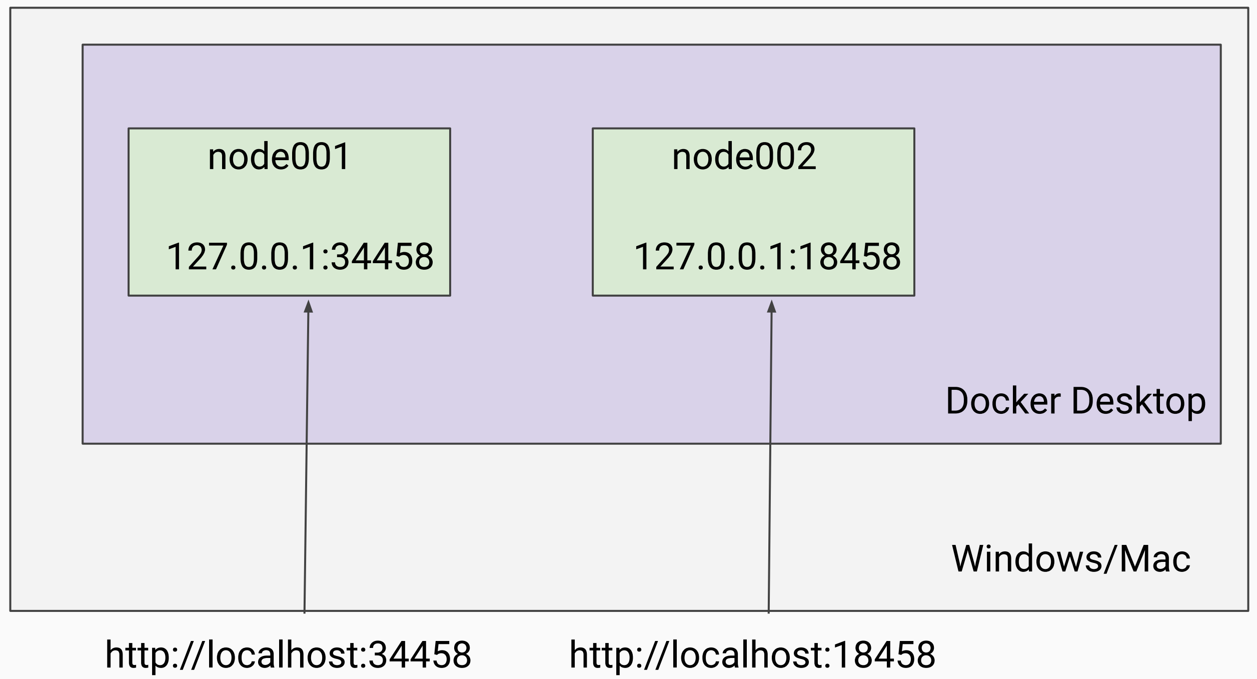
Windows size 3, we looks last three days data

Feature 1	Feature 2	Feature 3	Target
300	320	310	350
320	310	350	390

Environment Setup

Deployment Overview

Provisioning with multikf https://github.com/FootprintAI/multikf/tree/main/docs

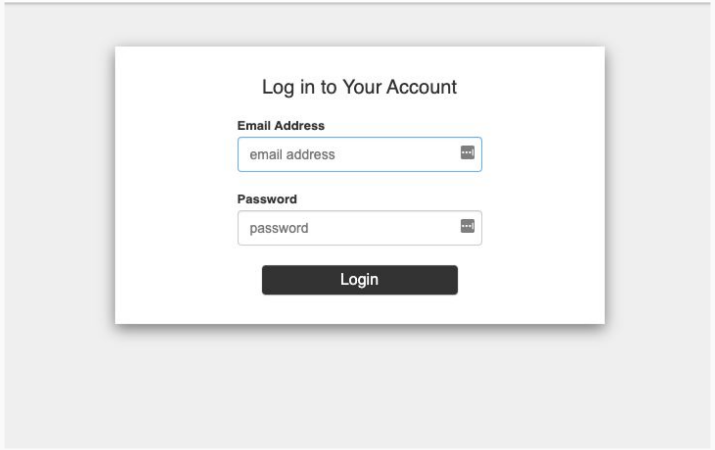


Deployment Overview: few steps setup

```
// install dockerd
wget https://raw.githubusercontent.com/FootprintAI/multikf/main/docker/linux.sh
chmod +x linux.sh
sudo ./linux.sh

// install multikf
wget https://github.com/FootprintAI/multikf/raw/main/build/multikf.linux
chmod +x multikf.linux

// add an instances with port 80/443 exported
./multikf.linux add node002 --export_ports 80:80,443:443

// connect kubeflow
./multikf.linux connect kubefloe node002
```

Wait! 所以我說那個帳號密碼呢?

Account: user@example.com
Password: 12341234



Log in to Your Account

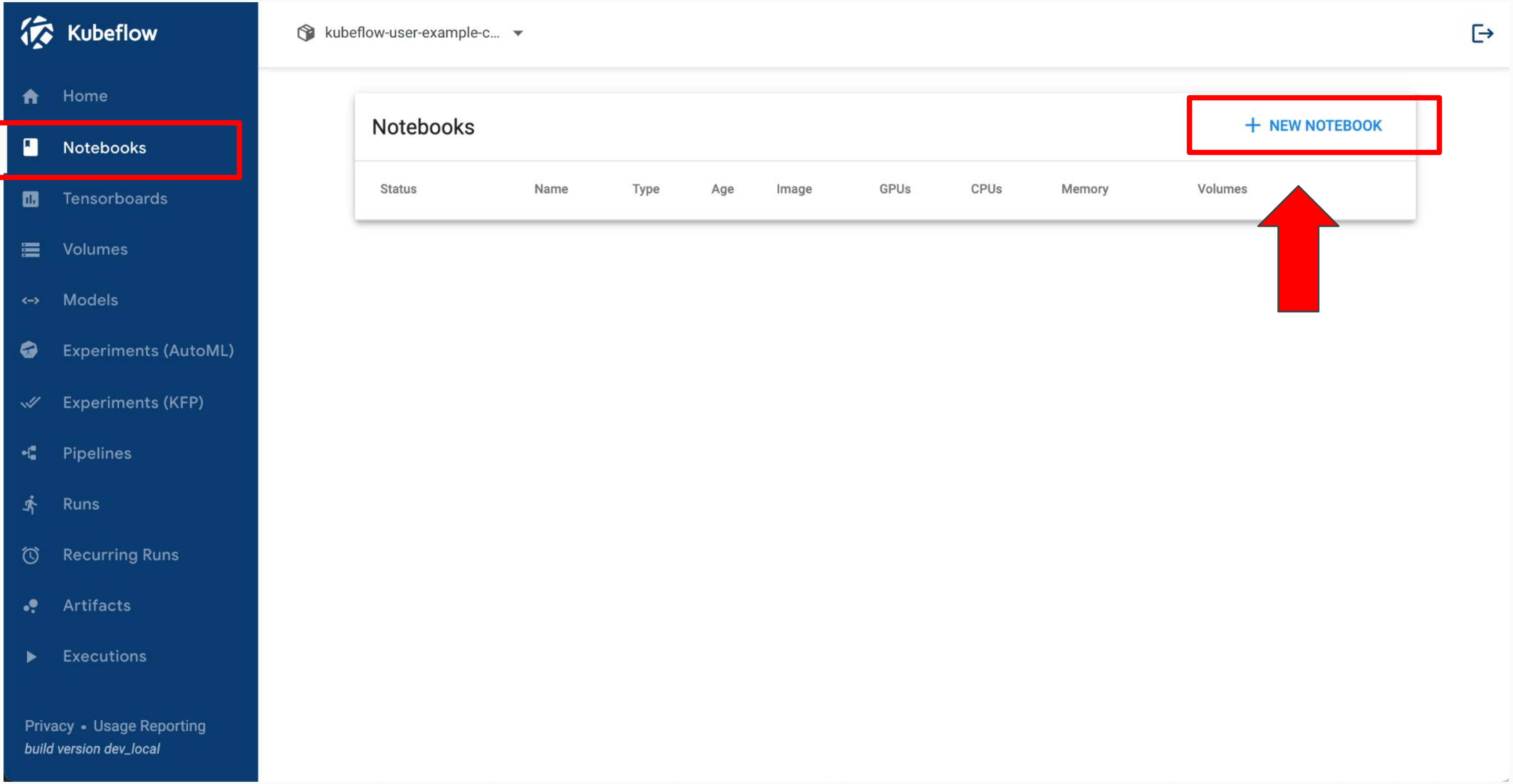
Email Address

email address

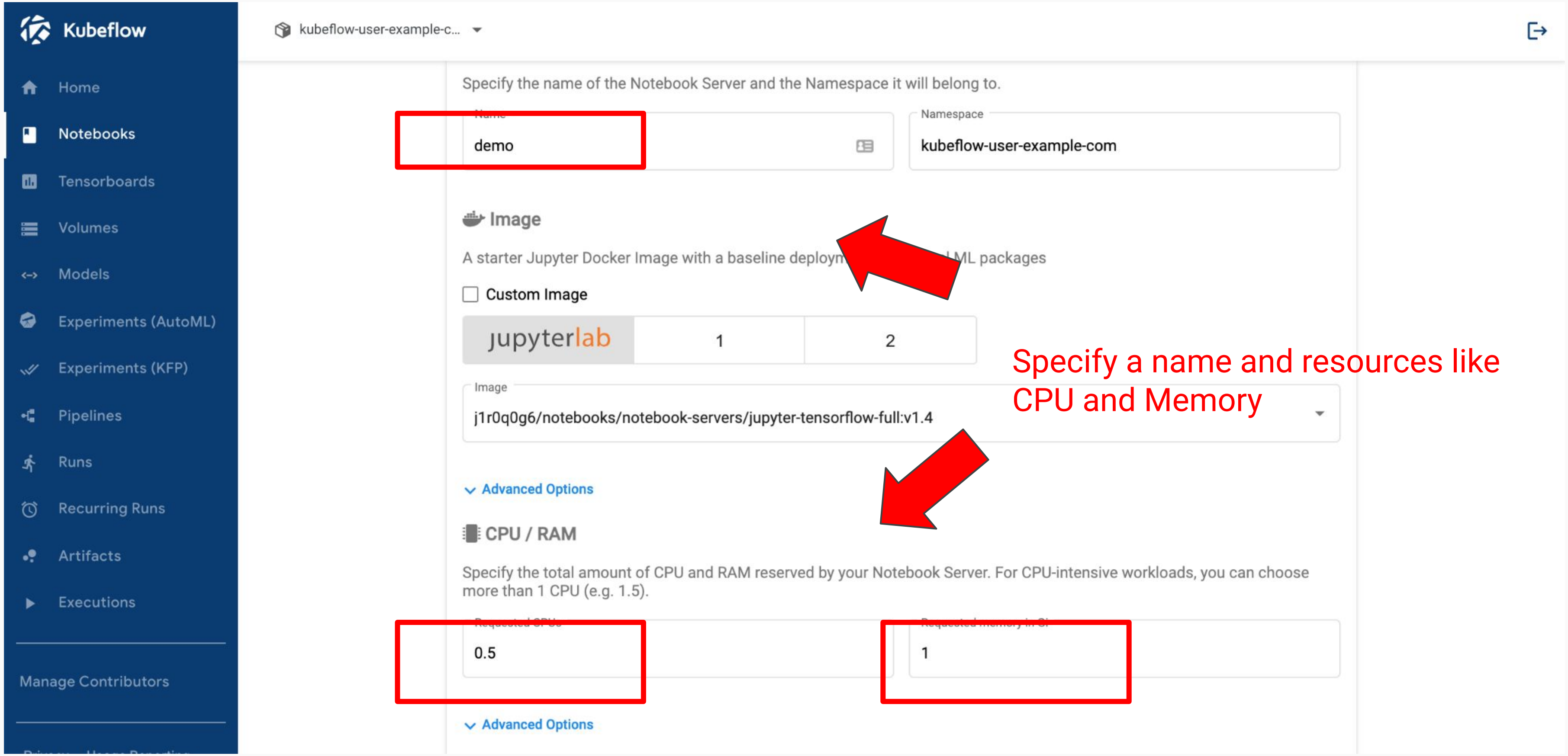
Password

password

Step1: Use Notebook as Online IDE (1/3)



Step1: Use Notebook as Online IDE (2/3)



Specify the name of the Notebook Server and the Namespace it will belong to.

Name: demo

Namespace: kubeflow-user-example-com

Image

A starter Jupyter Docker Image with a baseline deploym... ML packages

Custom Image

Image: j1r0q0g6/notebooks/notebook-servers/jupyter-tensorflow-full:v1.4

CPU / RAM

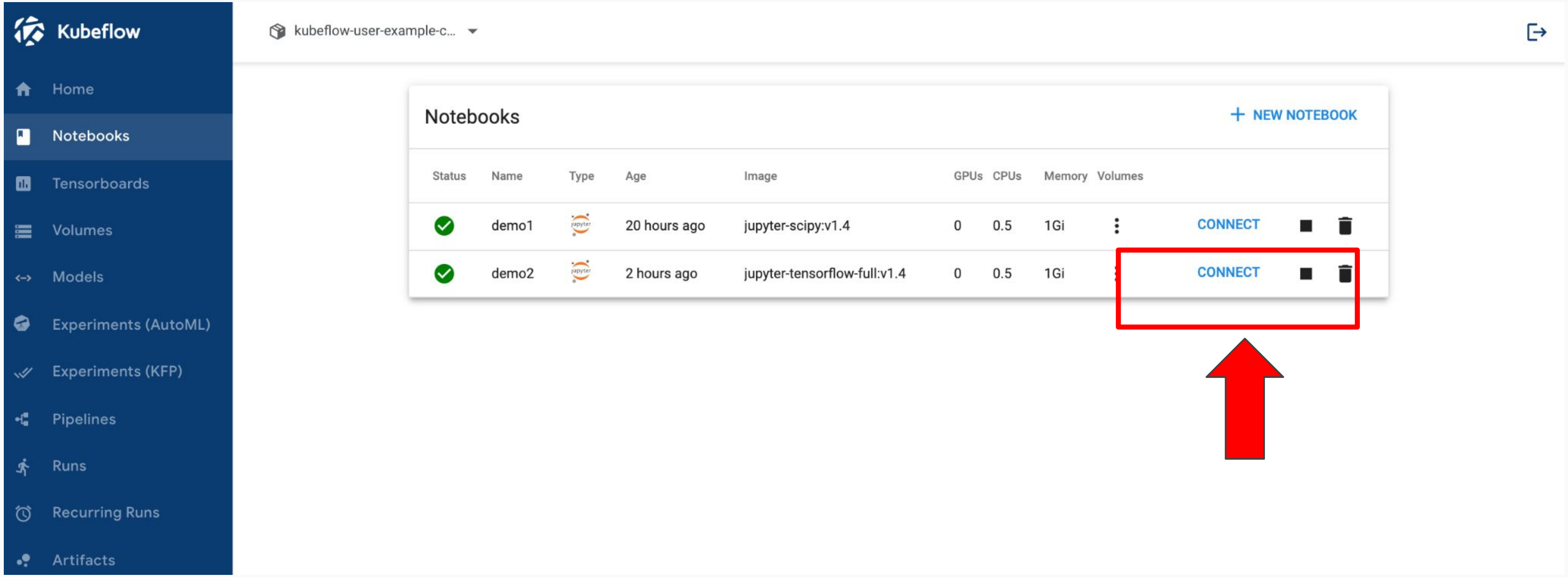
Specify the total amount of CPU and RAM reserved by your Notebook Server. For CPU-intensive workloads, you can choose more than 1 CPU (e.g. 1.5).

Requested CPUs: 0.5

Requested memory in Gi: 1

Specify a name and resources like CPU and Memory

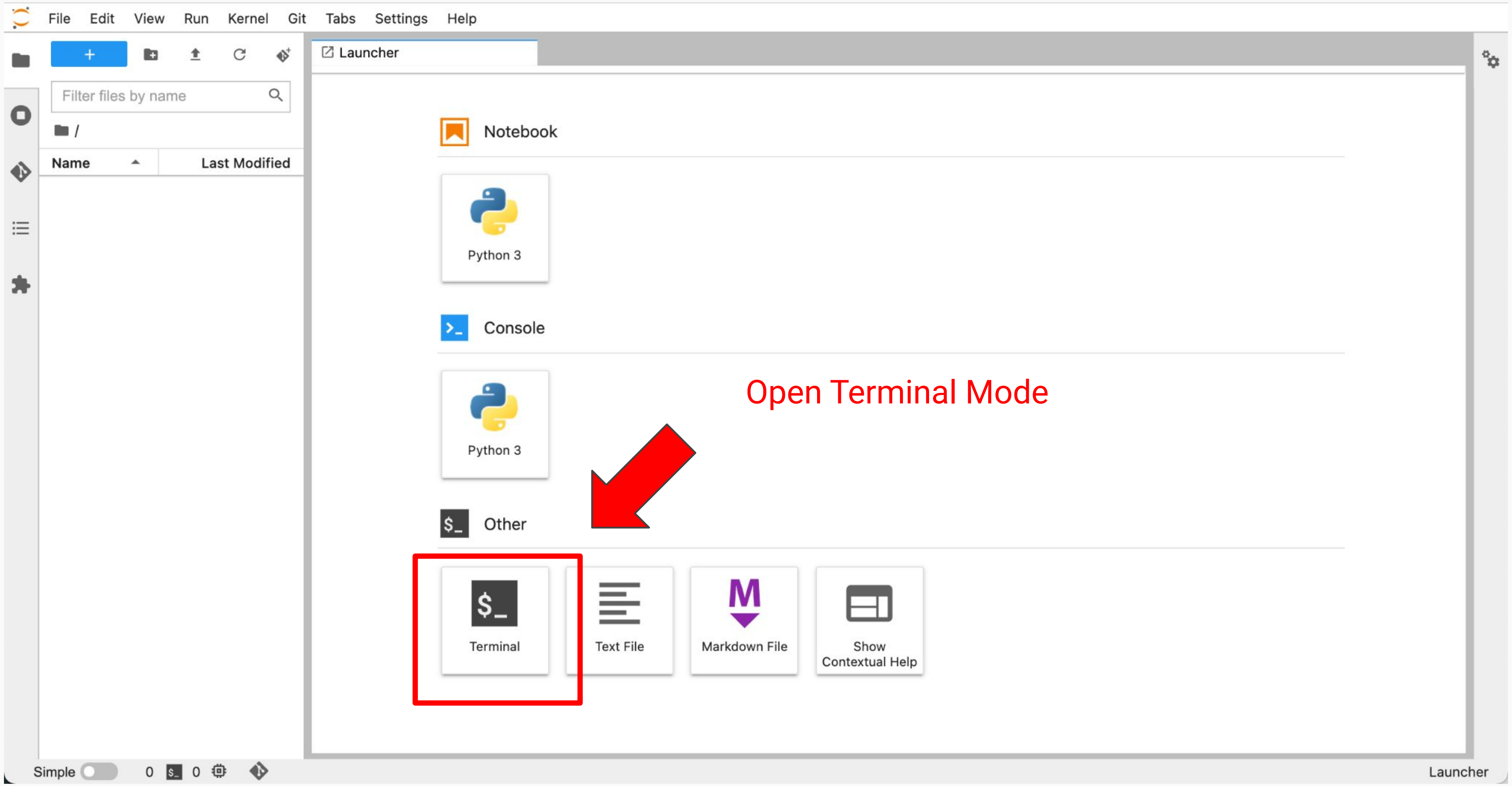
Step1: Use Notebook as Online IDE (3/3)



Notebooks

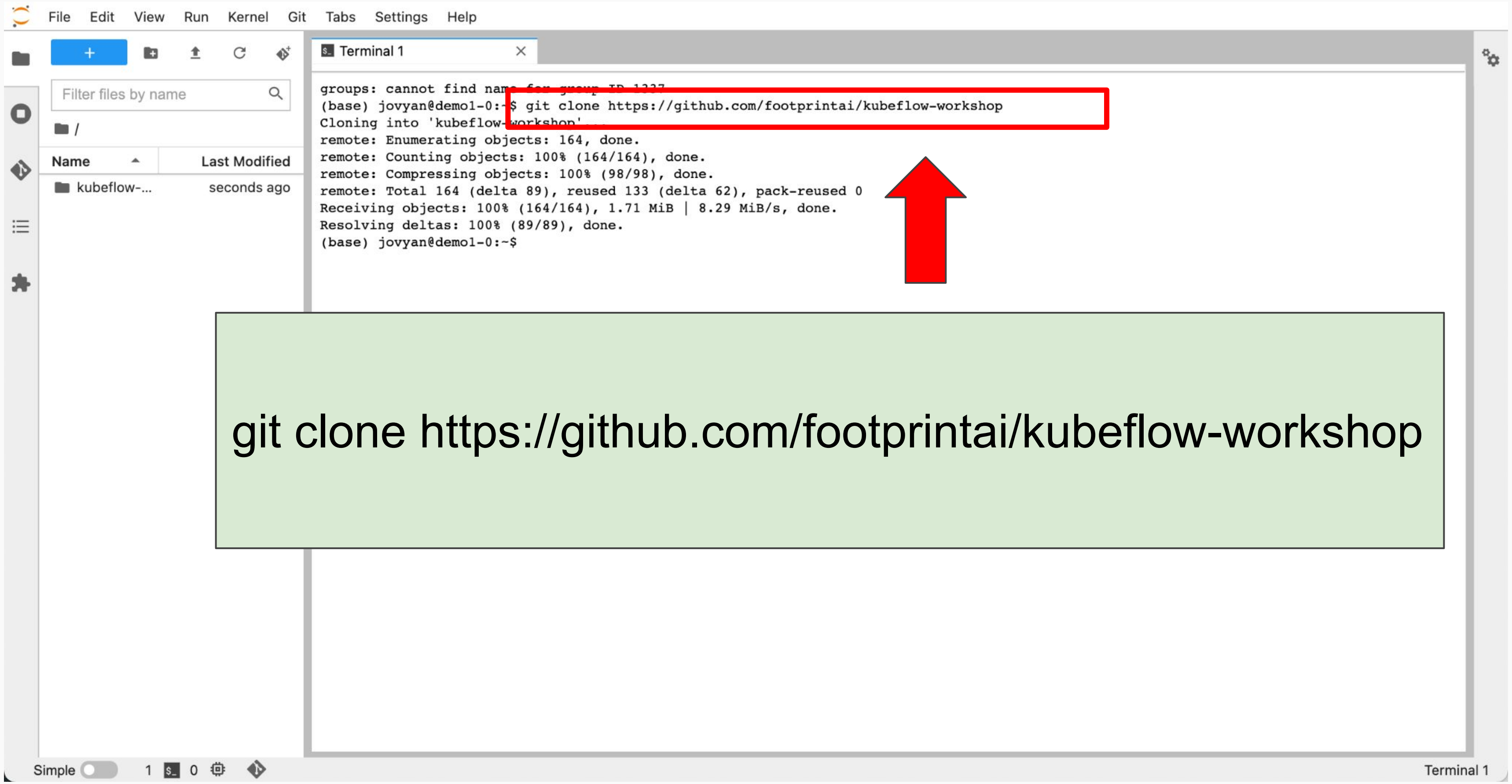
Status	Name	Type	Age	Image	GPUs	CPUs	Memory	Volumes			
✓	demo1	jupyter	20 hours ago	jupyter-scipy:v1.4	0	0.5	1Gi	⋮	CONNECT	■	🗑
✓	demo2	jupyter	2 hours ago	jupyter-tensorflow-full:v1.4	0	0.5	1Gi	⋮	CONNECT	■	🗑

Step2: Use terminal to download the materials (1/3)



Open Terminal Mode

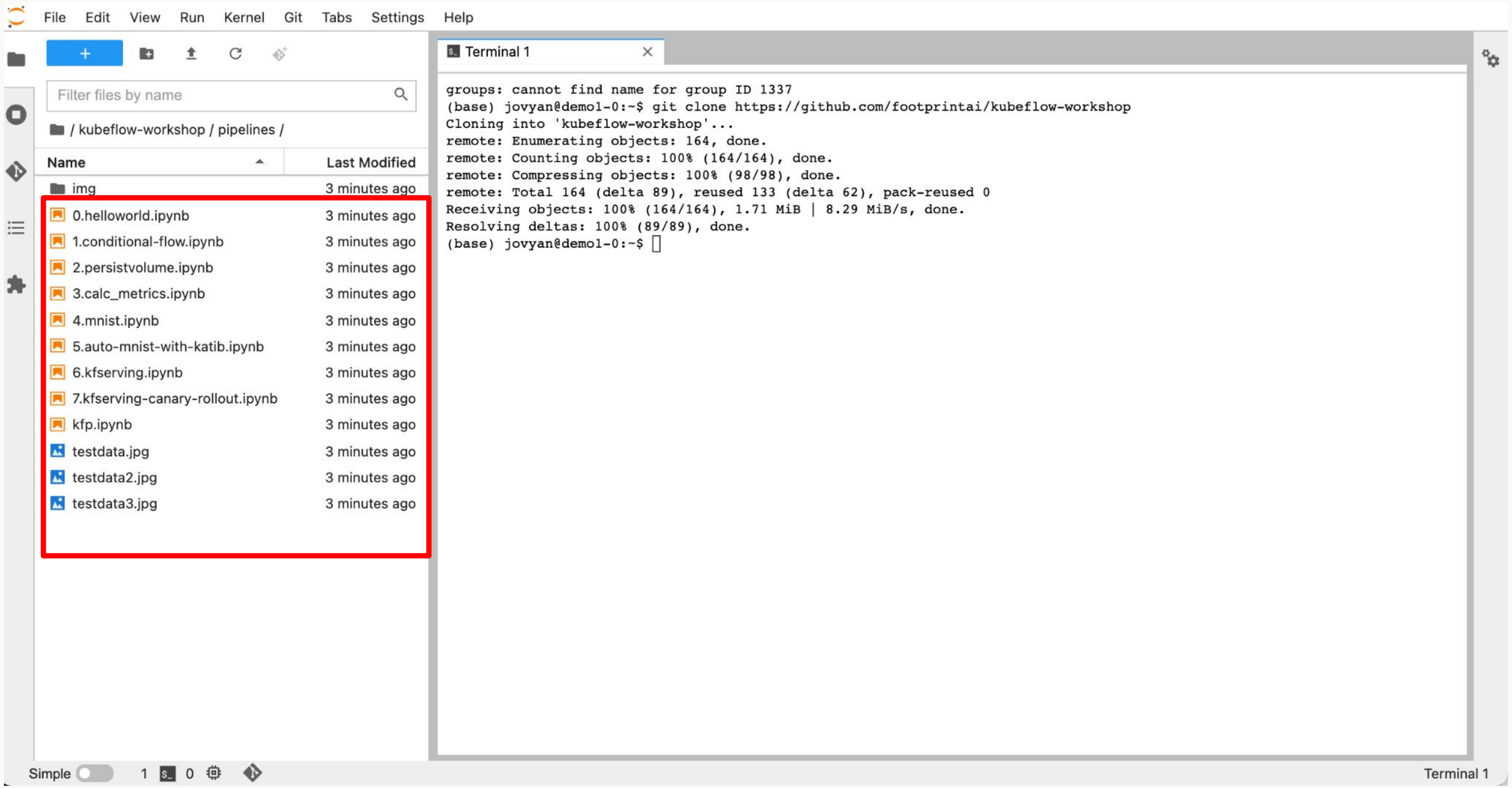
Step2: Use terminal to download the materials (2/3)



```
groups: cannot find name for group ID 1337
(base) jovyan@demo1-0:~$ git clone https://github.com/footprintai/kubeflow-workshop
Cloning into 'kubeflow-workshop'...
remote: Enumerating objects: 164, done.
remote: Counting objects: 100% (164/164), done.
remote: Compressing objects: 100% (98/98), done.
remote: Total 164 (delta 89), reused 133 (delta 62), pack-reused 0
Receiving objects: 100% (164/164), 1.71 MiB | 8.29 MiB/s, done.
Resolving deltas: 100% (89/89), done.
(base) jovyan@demo1-0:~$
```

```
git clone https://github.com/footprintai/kubeflow-workshop
```

Step2: Use terminal to download the materials (3/3)

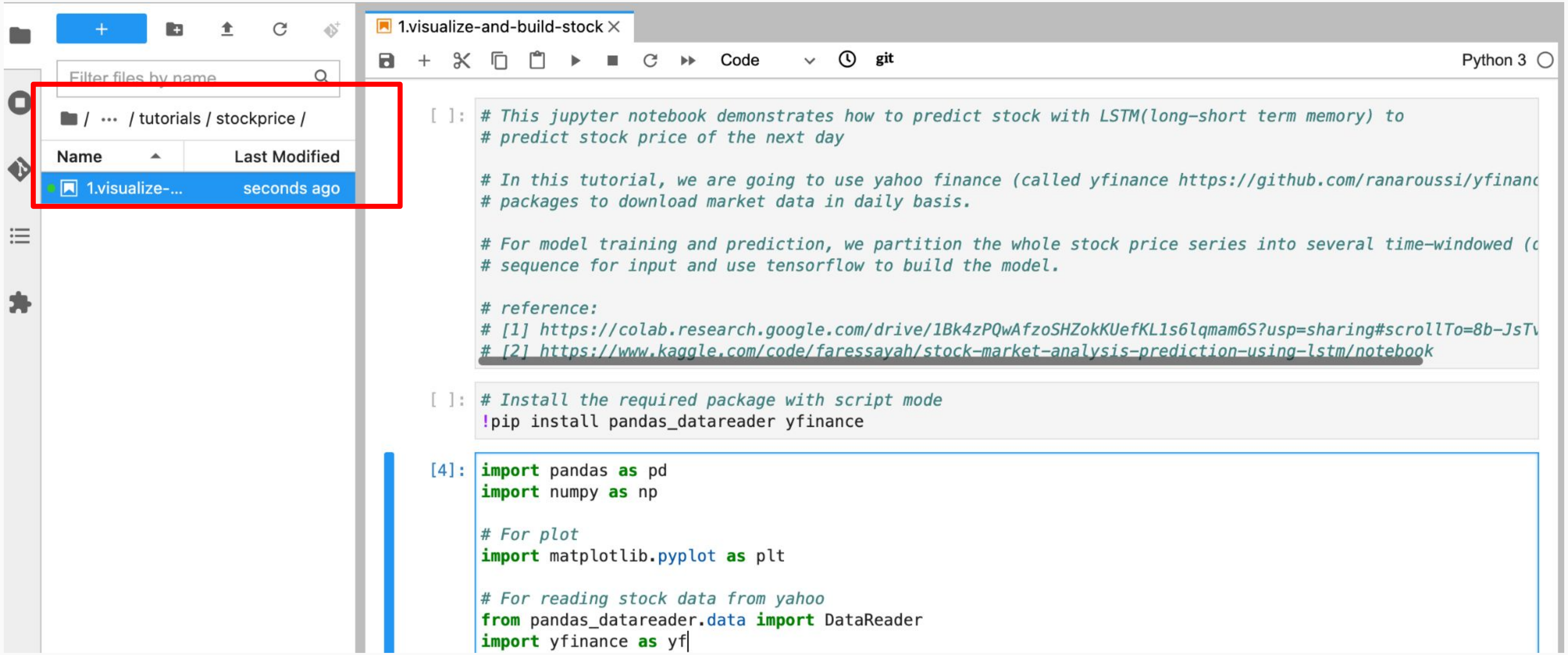


Name	Last Modified
img	3 minutes ago
0.helloworld.ipynb	3 minutes ago
1.conditional-flow.ipynb	3 minutes ago
2.persistvolume.ipynb	3 minutes ago
3.calc_metrics.ipynb	3 minutes ago
4.mnist.ipynb	3 minutes ago
5.auto-mnist-with-katib.ipynb	3 minutes ago
6.kfserving.ipynb	3 minutes ago
7.kfserving-canary-rollout.ipynb	3 minutes ago
kfp.ipynb	3 minutes ago
testdata.jpg	3 minutes ago
testdata2.jpg	3 minutes ago
testdata3.jpg	3 minutes ago

```
groups: cannot find name for group ID 1337
(base) jovyan@demo1-0:~$ git clone https://github.com/footprintai/kubeflow-workshop
Cloning into 'kubeflow-workshop'...
remote: Enumerating objects: 164, done.
remote: Counting objects: 100% (164/164), done.
remote: Compressing objects: 100% (98/98), done.
remote: Total 164 (delta 89), reused 133 (delta 62), pack-reused 0
Receiving objects: 100% (164/164), 1.71 MiB | 8.29 MiB/s, done.
Resolving deltas: 100% (89/89), done.
(base) jovyan@demo1-0:~$ 
```

Financial Model Builder

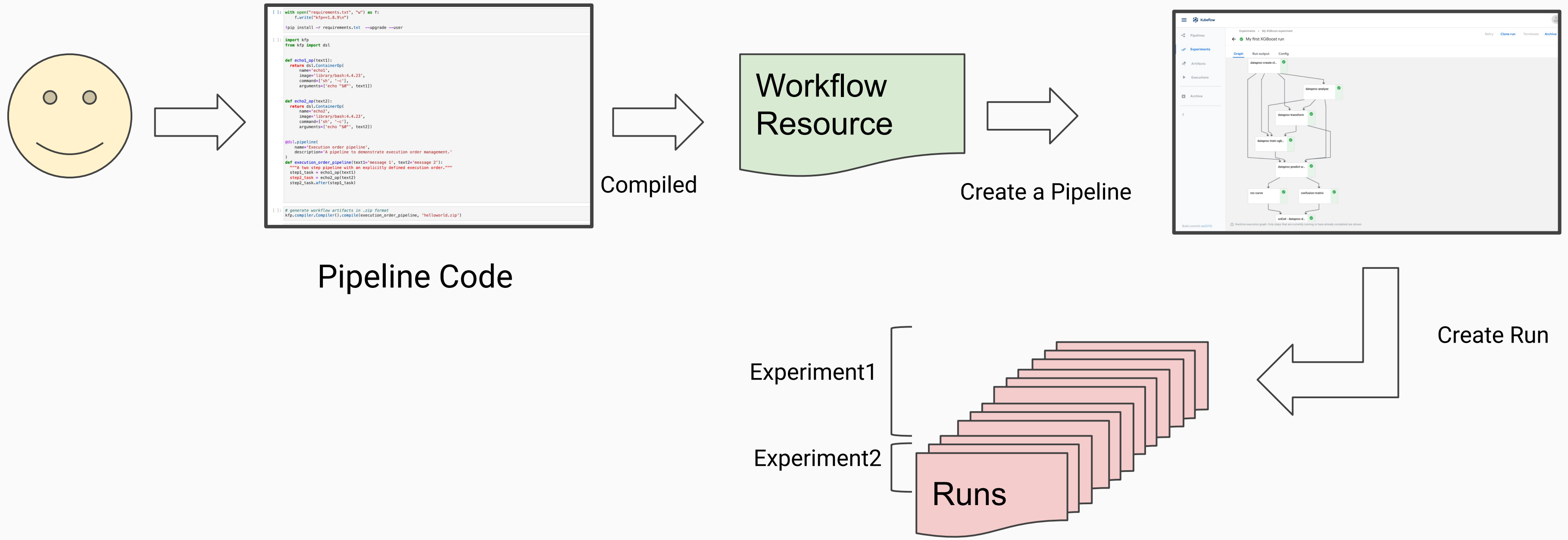
Step3: Open financial model ipynb under tutorials/stockprice



```
[ ]: # This jupyter notebook demonstrates how to predict stock with LSTM(long-short term memory) to
     # predict stock price of the next day

     # In this tutorial, we are going to use yahoo finance (called yfinance https://github.com/ranaroussi/yfinanc
     # packages to download market data in daily basis.

     # For model training and prediction, we partition the whole stock price series into several time-windowed (c
     # sequence for input and use tensorflow to build the model.

     # reference:
     # [1] https://colab.research.google.com/drive/1Bk4zPQwAfzoSHZokKUefKL1s6lqmam6S?usp=sharing#scrollTo=8b-JsTv
     # [2] https://www.kaggle.com/code/faressayah/stock-market-analysis-prediction-using-lstm/notebook

[ ]: # Install the required package with script mode
     !pip install pandas_datareader yfinance

[4]: import pandas as pd
     import numpy as np

     # For plot
     import matplotlib.pyplot as plt

     # For reading stock data from yahoo
     from pandas_datareader.data import DataReader
     import yfinance as yf
```

Kubeflow Terms

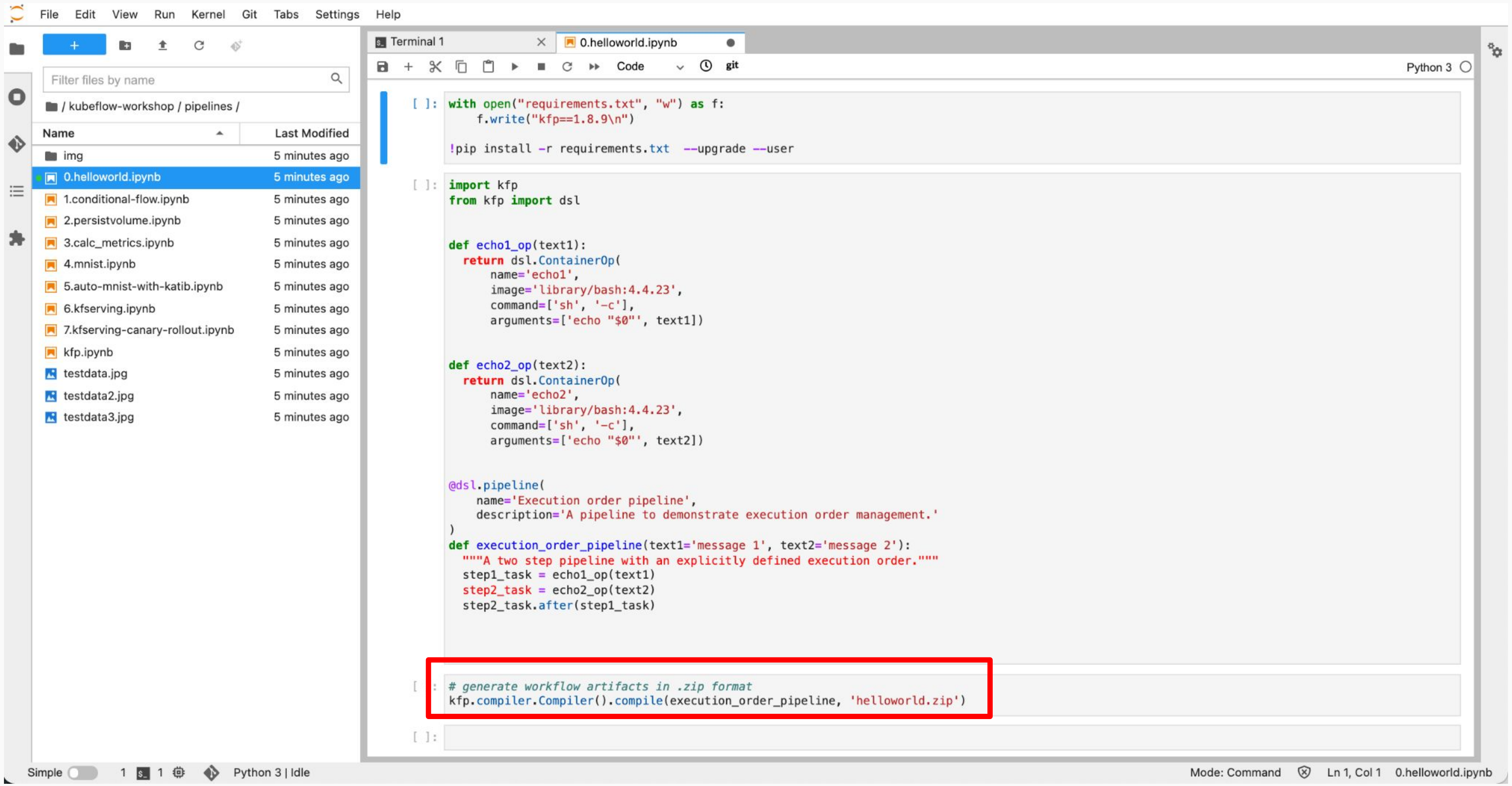
詞彙說明



```
[ ]: with open("requirements.txt", "w") as f:
         f.write("kfp==1.8.9\n")

     !pip install -r requirements.txt  --upgrade --user

[ ]: import kfp
     from kfp import dsl


     def echo1_op(text1):
       return dsl.ContainerOp(
           name='echo1',
           image='library/bash:4.4.23',
           command=['sh', '-c'],
           arguments=['echo "$0"', text1])


     def echo2_op(text2):
       return dsl.ContainerOp(
           name='echo2',
           image='library/bash:4.4.23',
           command=['sh', '-c'],
           arguments=['echo "$0"', text2])


     @dsl.pipeline(
         name='Execution order pipeline',
         description='A pipeline to demonstrate execution order management.'
     )
     def execution_order_pipeline(text1='message 1', text2='message 2'):
       """A two step pipeline with an explicitly defined execution order."""
       step1_task = echo1_op(text1)
       step2_task = echo2_op(text2)
       step2_task.after(step1_task)

[ ]: # generate workflow artifacts in .zip format
     kfp.compiler.Compiler().compile(execution_order_pipeline, 'helloworld.zip')
```

Hello World Example

Step4: Compile helloworld.ipynb (1/2)


File Edit View Run Kernel Git Tabs Settings Help
Terminal 1
0.helloworld.ipynb
Code
git
Python 3
Filter files by name
/ kubeflow-workshop / pipelines /

Name	Last Modified
img	5 minutes ago
0.helloworld.ipynb	5 minutes ago
1.conditional-flow.ipynb	5 minutes ago
2.persistvolume.ipynb	5 minutes ago
3.calc_metrics.ipynb	5 minutes ago
4.mnist.ipynb	5 minutes ago
5.auto-mnist-with-katib.ipynb	5 minutes ago
6.kfserving.ipynb	5 minutes ago
7.kfserving-canary-rollout.ipynb	5 minutes ago
kfp.ipynb	5 minutes ago
testdata.jpg	5 minutes ago
testdata2.jpg	5 minutes ago
testdata3.jpg	5 minutes ago

```
[ ]: with open("requirements.txt", "w") as f:
         f.write("kfp==1.8.9\n")

     !pip install -r requirements.txt  --upgrade --user
```

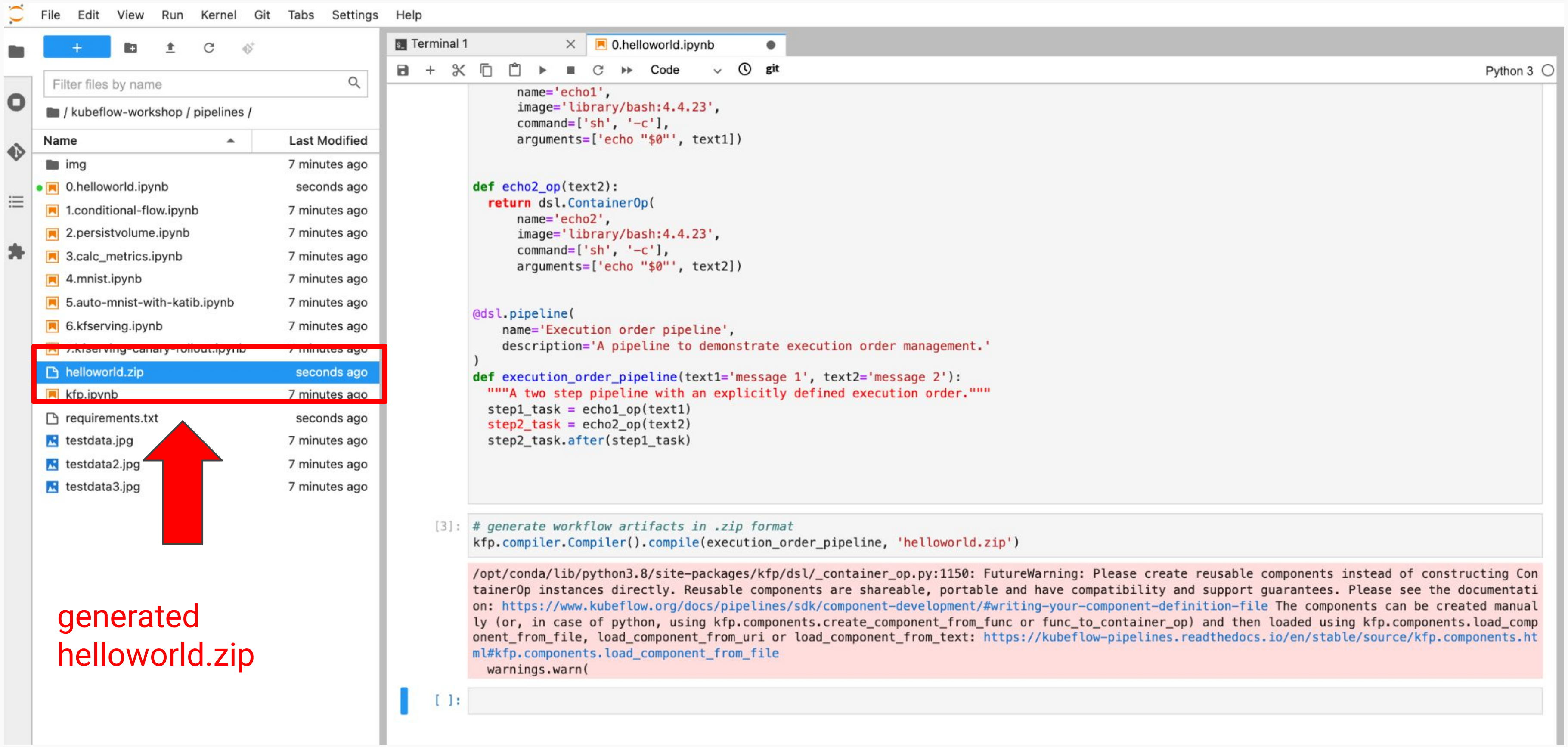
```
[ ]: import kfp
     from kfp import dsl


     def echo1_op(text1):
       return dsl.ContainerOp(
           name='echo1',
           image='library/bash:4.4.23',
           command=['sh', '-c'],
           arguments=['echo "$0"', text1])


     def echo2_op(text2):
       return dsl.ContainerOp(
           name='echo2',
           image='library/bash:4.4.23',
           command=['sh', '-c'],
           arguments=['echo "$0"', text2])


     @dsl.pipeline(
         name='Execution order pipeline',
         description='A pipeline to demonstrate execution order management.'
     )
     def execution_order_pipeline(text1='message 1', text2='message 2'):
       """A two step pipeline with an explicitly defined execution order."""
       step1_task = echo1_op(text1)
       step2_task = echo2_op(text2)
       step2_task.after(step1_task)
```

```
[ ]: # generate workflow artifacts in .zip format
     kfp.compiler.Compiler().compile(execution_order_pipeline, 'helloworld.zip')
```

```
[ ]:
```

Simple 1 1 Python 3 | Idle
Mode: Command Ln 1, Col 1 0.helloworld.ipynb

Step4: Compile helloworld.ipynb (2/2)



Name	Last Modified
img	7 minutes ago
0.helloworld.ipynb	seconds ago
1.conditional-flow.ipynb	7 minutes ago
2.persistvolume.ipynb	7 minutes ago
3.calc_metrics.ipynb	7 minutes ago
4.mnist.ipynb	7 minutes ago
5.auto-mnist-with-katib.ipynb	7 minutes ago
6.kfserving.ipynb	7 minutes ago
7.kfserving-canary-rollout.ipynb	7 minutes ago
helloworld.zip	seconds ago
kfp.ipynb	7 minutes ago
requirements.txt	seconds ago
testdata.jpg	7 minutes ago
testdata2.jpg	7 minutes ago
testdata3.jpg	7 minutes ago

```
      name='echo1',
      image='library/bash:4.4.23',
      command=['sh', '-c'],
      arguments=['echo "$0"', text1])


def echo2_op(text2):
  return dsl.ContainerOp(
      name='echo2',
      image='library/bash:4.4.23',
      command=['sh', '-c'],
      arguments=['echo "$0"', text2])


@dsl.pipeline(
    name='Execution order pipeline',
    description='A pipeline to demonstrate execution order management.'
)
def execution_order_pipeline(text1='message 1', text2='message 2'):
  """A two step pipeline with an explicitly defined execution order."""
  step1_task = echo1_op(text1)
  step2_task = echo2_op(text2)
  step2_task.after(step1_task)
```

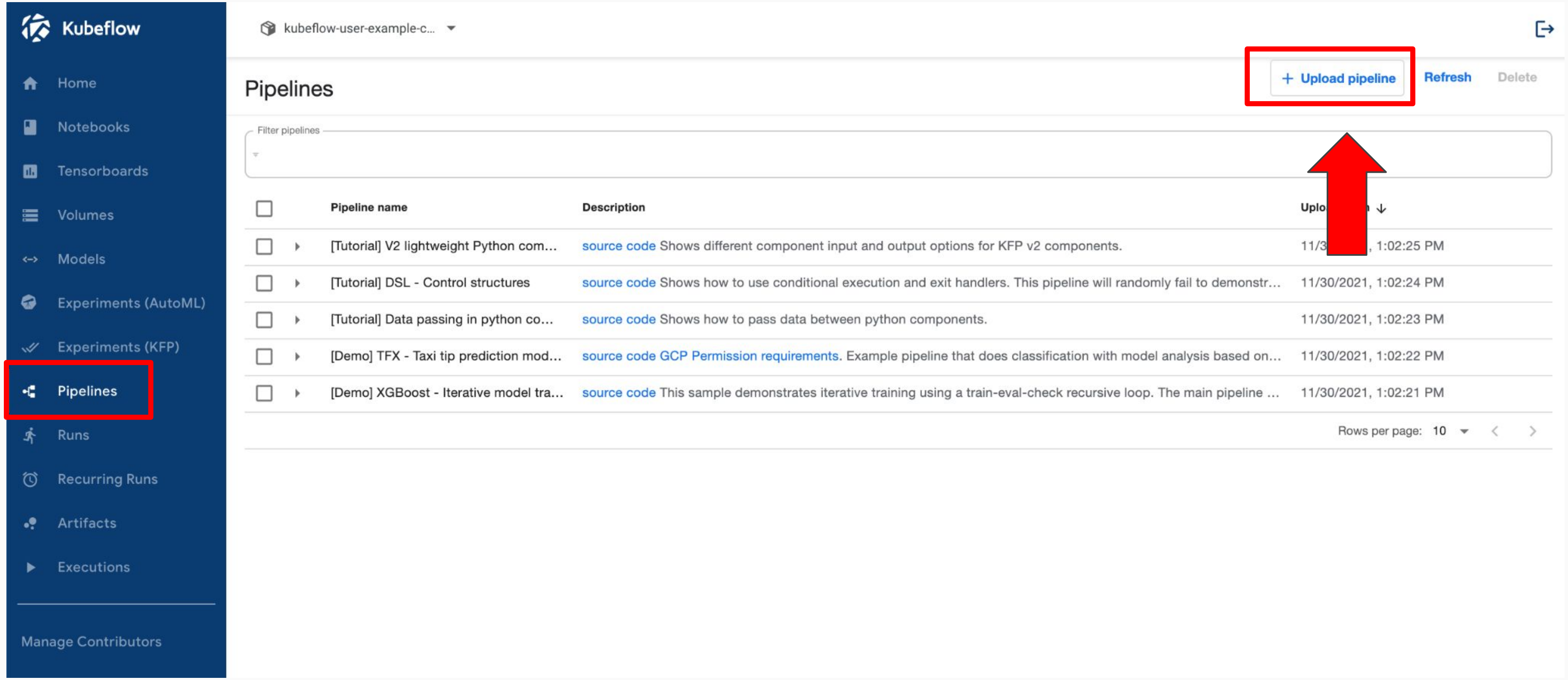
```
[3]: # generate workflow artifacts in .zip format
     kfp.compiler.Compiler().compile(execution_order_pipeline, 'helloworld.zip')
```

```
/opt/conda/lib/python3.8/site-packages/kfp/dsl/_container_op.py:1150: FutureWarning: Please create reusable components instead of constructing Con
tainerOp instances directly. Reusable components are shareable, portable and have compatibility and support guarantees. Please see the documentati
on: https://www.kubeflow.org/docs/pipelines/sdk/component-development/#writing-your-component-definition-file The components can be created manual
ly (or, in case of python, using kfp.components.create_component_from_func or func_to_container_op) and then loaded using kfp.components.load_comp
onent_from_file, load_component_from_uri or load_component_from_text: https://kubeflow-pipelines.readthedocs.io/en/stable/source/kfp.components.ht
ml#kfp.components.load_component_from_file
  warnings.warn(
```

```
[ ]:
```

generated helloworld.zip

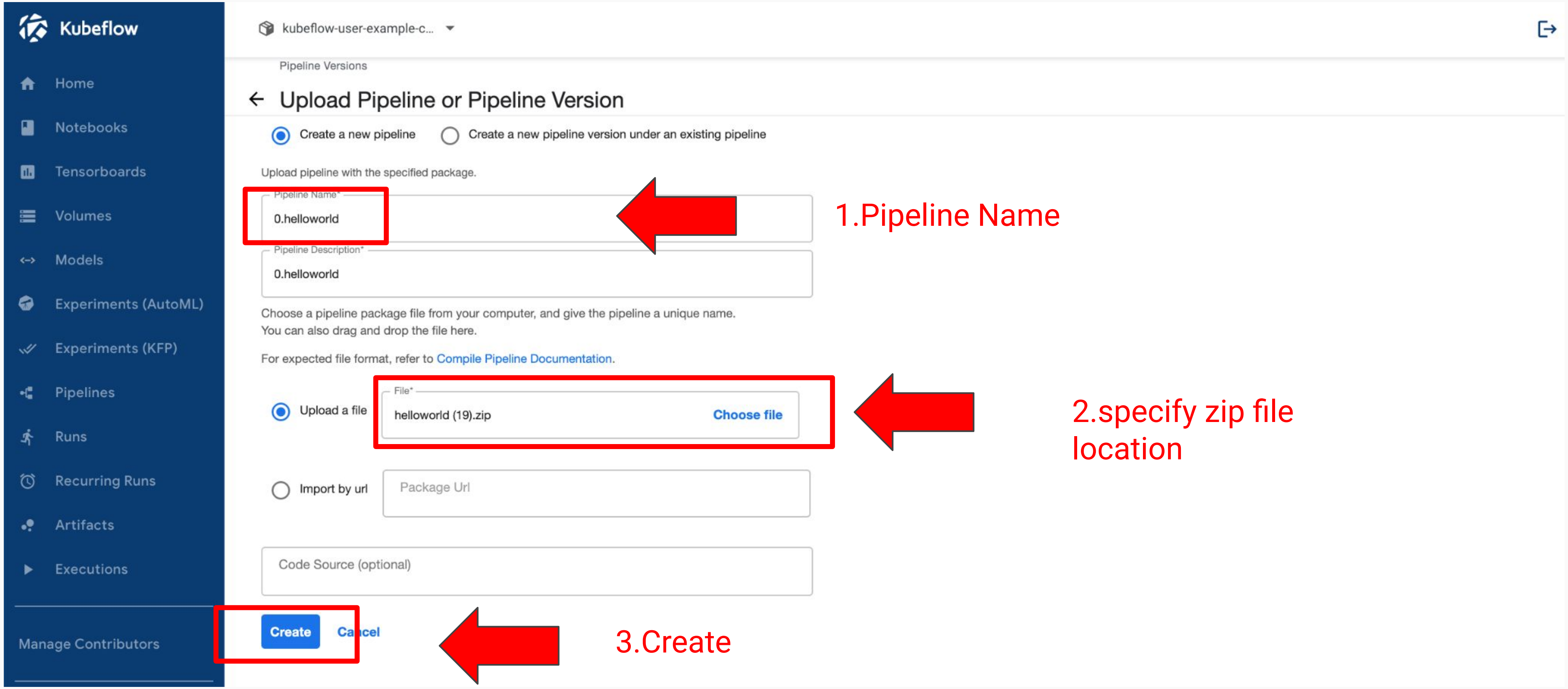
Step4: Create a Pipeline (1/7)



Pipelines

Pipeline name	Description	Uplo...
[Tutorial] V2 lightweight Python com…	source code Shows different component input and output options for KFP v2 components.	11/3..., 1:02:25 PM
[Tutorial] DSL - Control structures	source code Shows how to use conditional execution and exit handlers. This pipeline will randomly fail to demonstr…	11/30/2021, 1:02:24 PM
[Tutorial] Data passing in python co…	source code Shows how to pass data between python components.	11/30/2021, 1:02:23 PM
[Demo] TFX - Taxi tip prediction mod…	source code GCP Permission requirements. Example pipeline that does classification with model analysis based on…	11/30/2021, 1:02:22 PM
[Demo] XGBoost - Iterative model tra…	source code This sample demonstrates iterative training using a train-eval-check recursive loop. The main pipeline …	11/30/2021, 1:02:21 PM

Step4: Create a Pipeline (2/7)



Upload Pipeline or Pipeline Version

[✓ Create a new pipeline] Create a new pipeline version under an existing pipeline

Upload pipeline with the specified package.

Pipeline Name*: 0.helloworld

Pipeline Description*: 0.helloworld

Choose a pipeline package file from your computer, and give the pipeline a unique name.
You can also drag and drop the file here.

For expected file format, refer to Compile Pipeline Documentation.

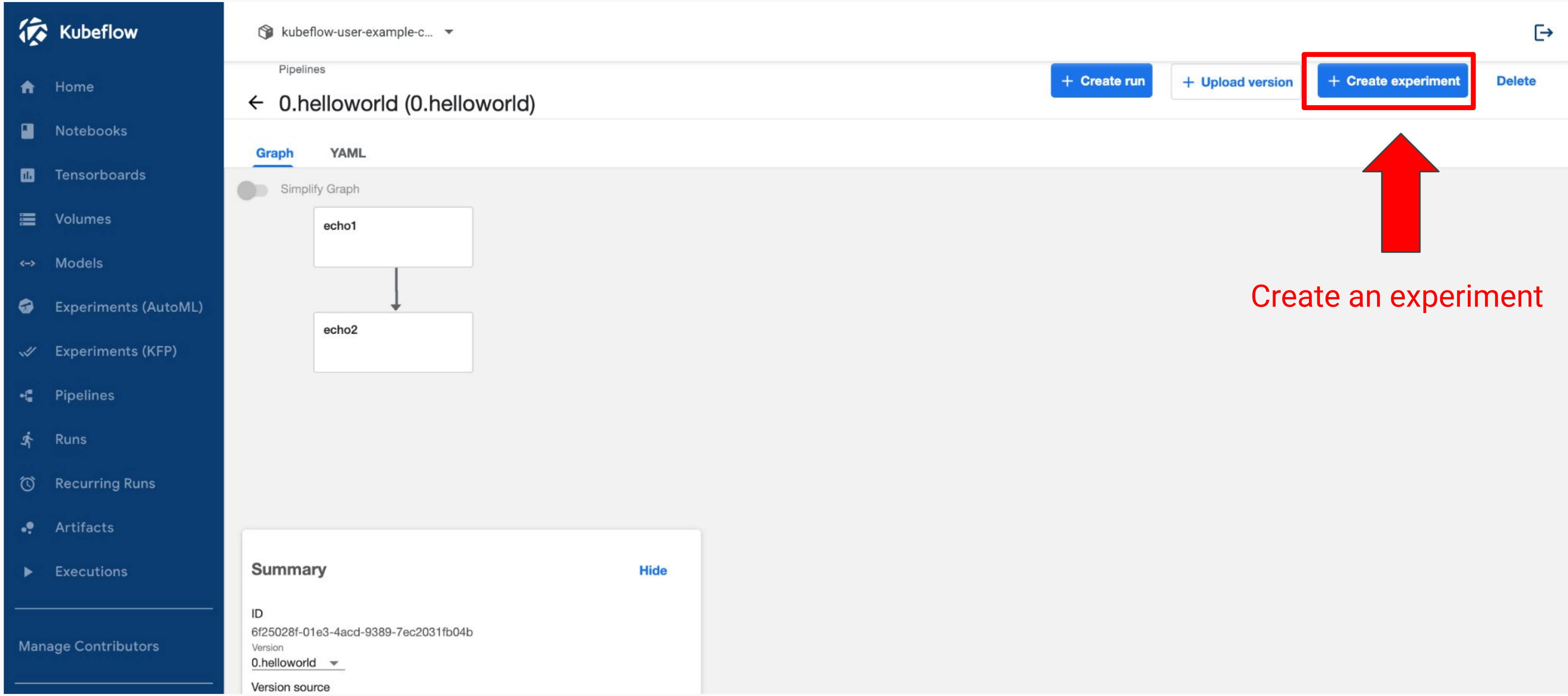
[✓ Upload a file] File*: helloworld (19).zip

1.Pipeline Name

2.specify zip file location

3.Create

Step4: Create a Pipeline (3/7)



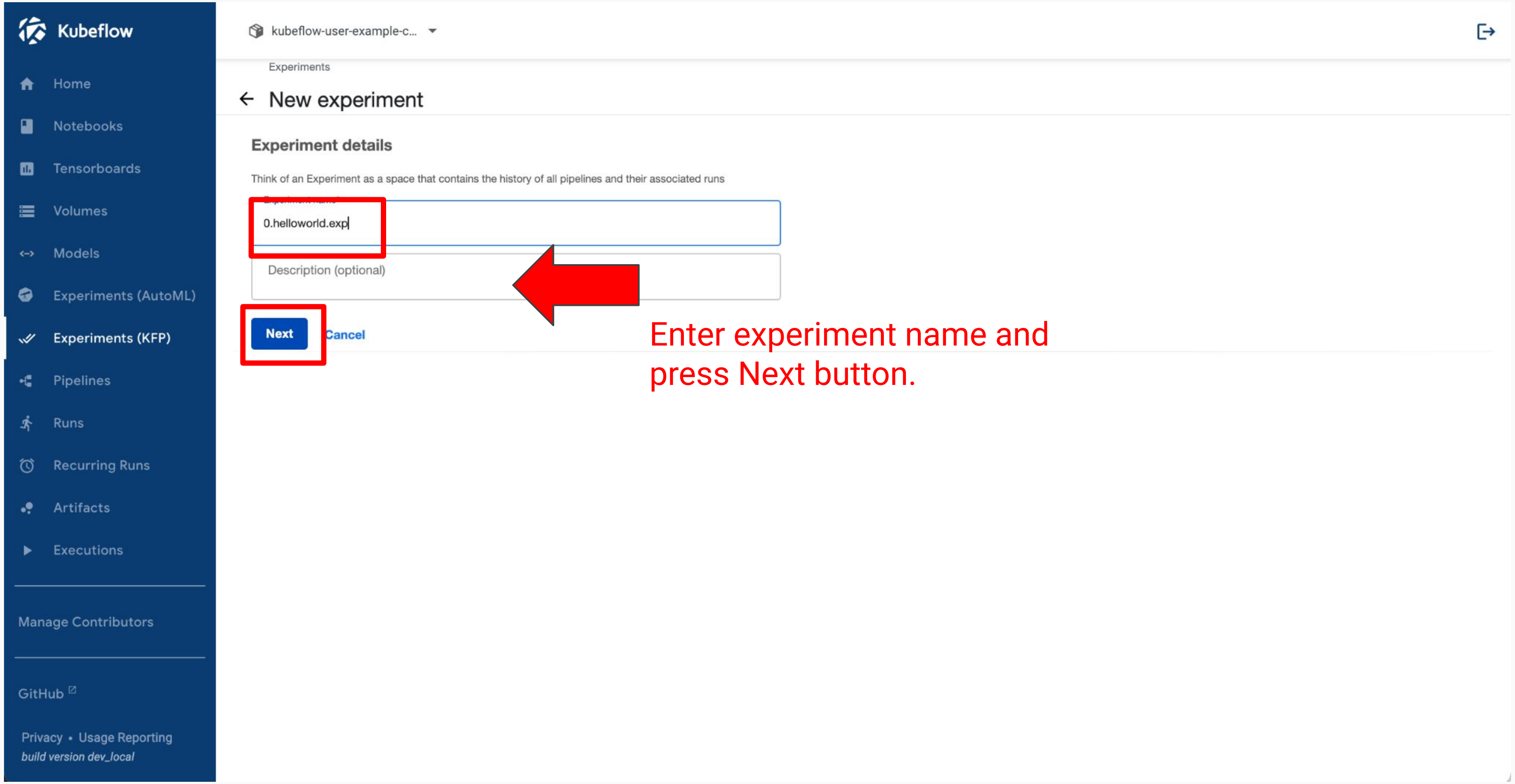
0.helloworld (0.helloworld)

Summary

ID
6f25028f-01e3-4acd-9389-7ec2031fb04b
Version
0.helloworld
Version source

Create an experiment

Step4: Create a Pipeline (4/7)

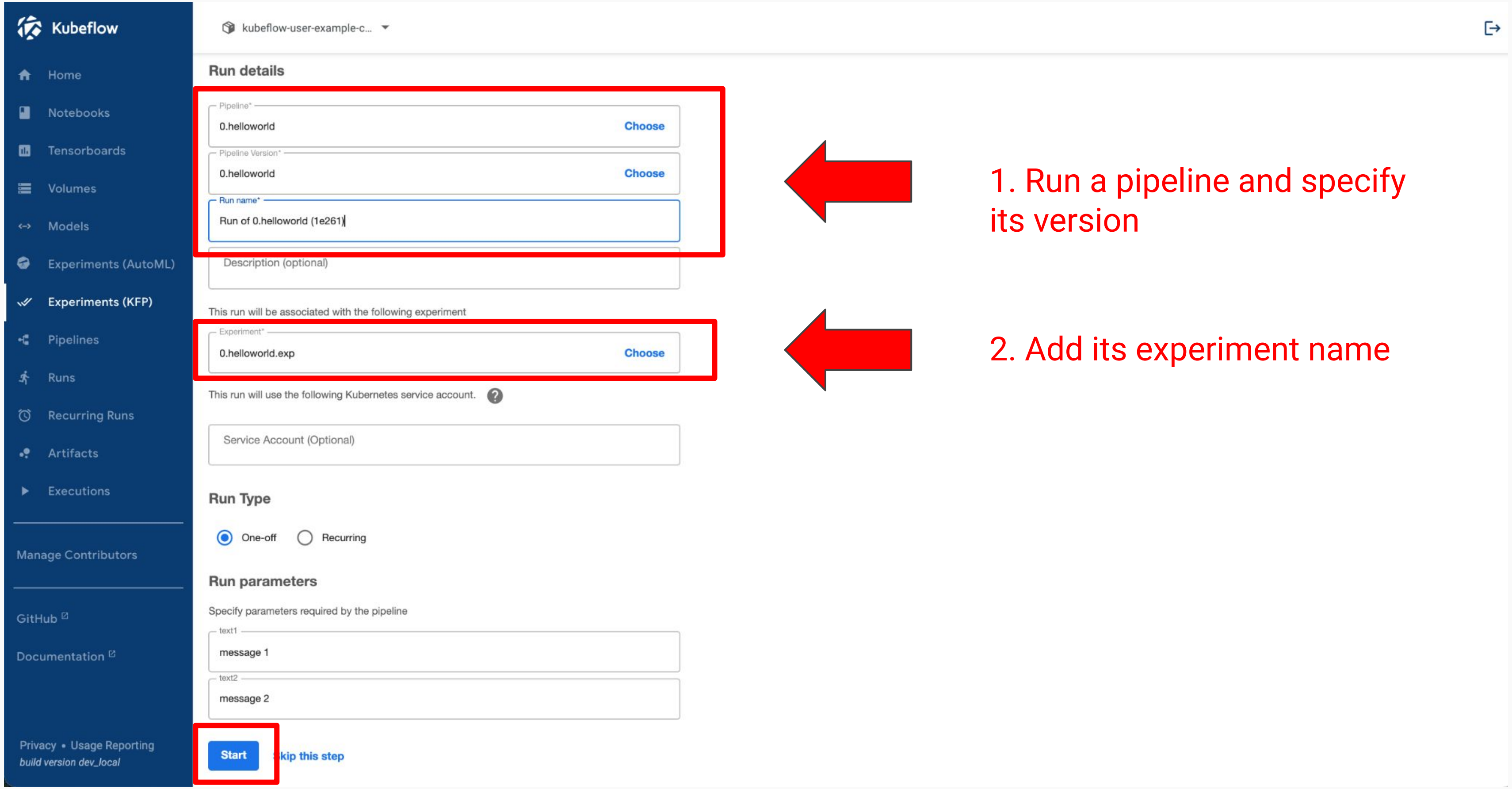


Think of an Experiment as a space that contains the history of all pipelines and their associated runs

Experiment name: 0.helloworld.exp

Enter experiment name and press Next button.

Step4: Create a Pipeline (5/7)



Run details

Pipeline*: 0.helloworld — Choose

Pipeline Version*: 0.helloworld — Choose

Run name*: Run of 0.helloworld (1e261)

Description (optional)

This run will be associated with the following experiment

Experiment*: 0.helloworld.exp — Choose

This run will use the following Kubernetes service account.

Service Account (Optional)

Run Type

[✓ One-off] Recurring

Run parameters

Specify parameters required by the pipeline

text1: message 1

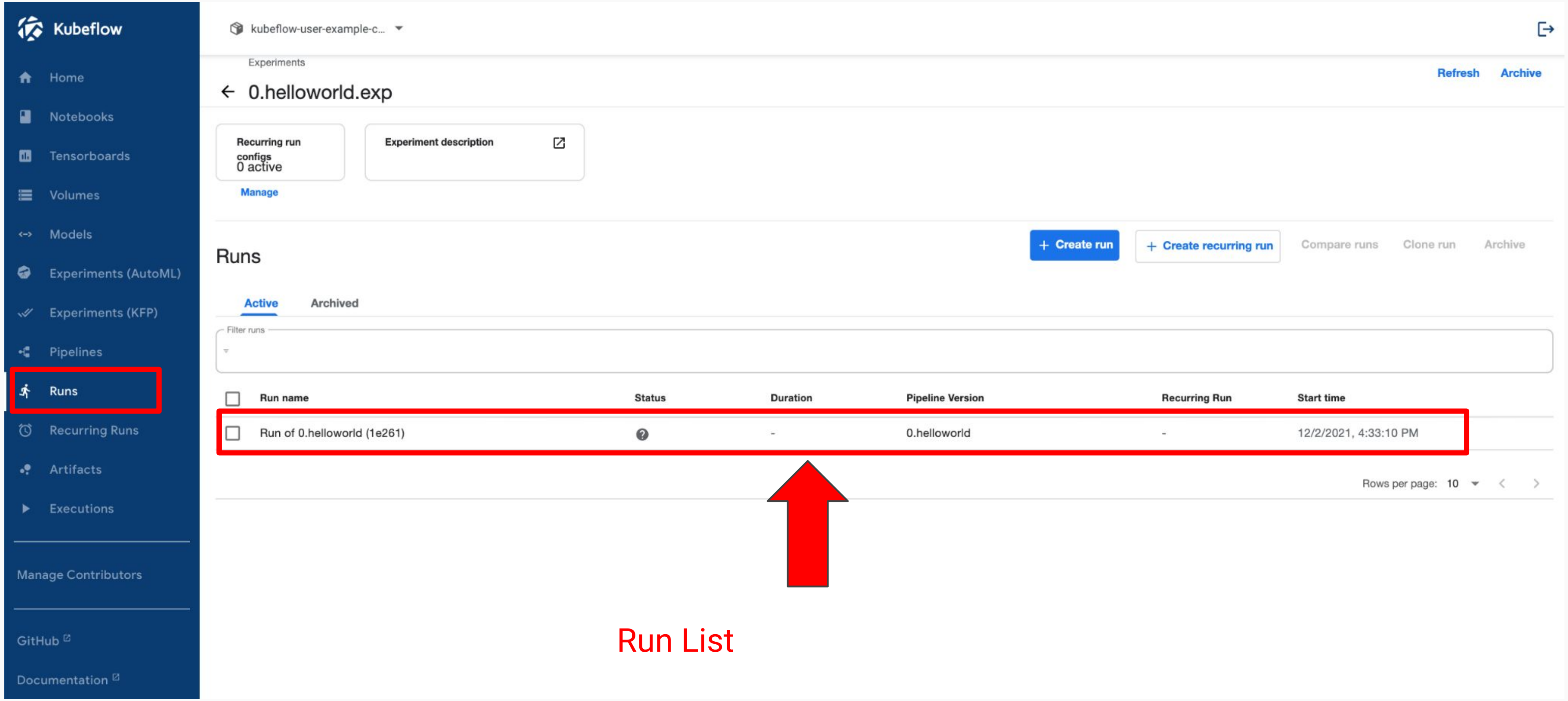
text2: message 2

Start   Skip this step

1. Run a pipeline and specify its version

2. Add its experiment name

Step4: Create a Pipeline (6/7)



0.helloworld.exp

Recurring run configs
0 active

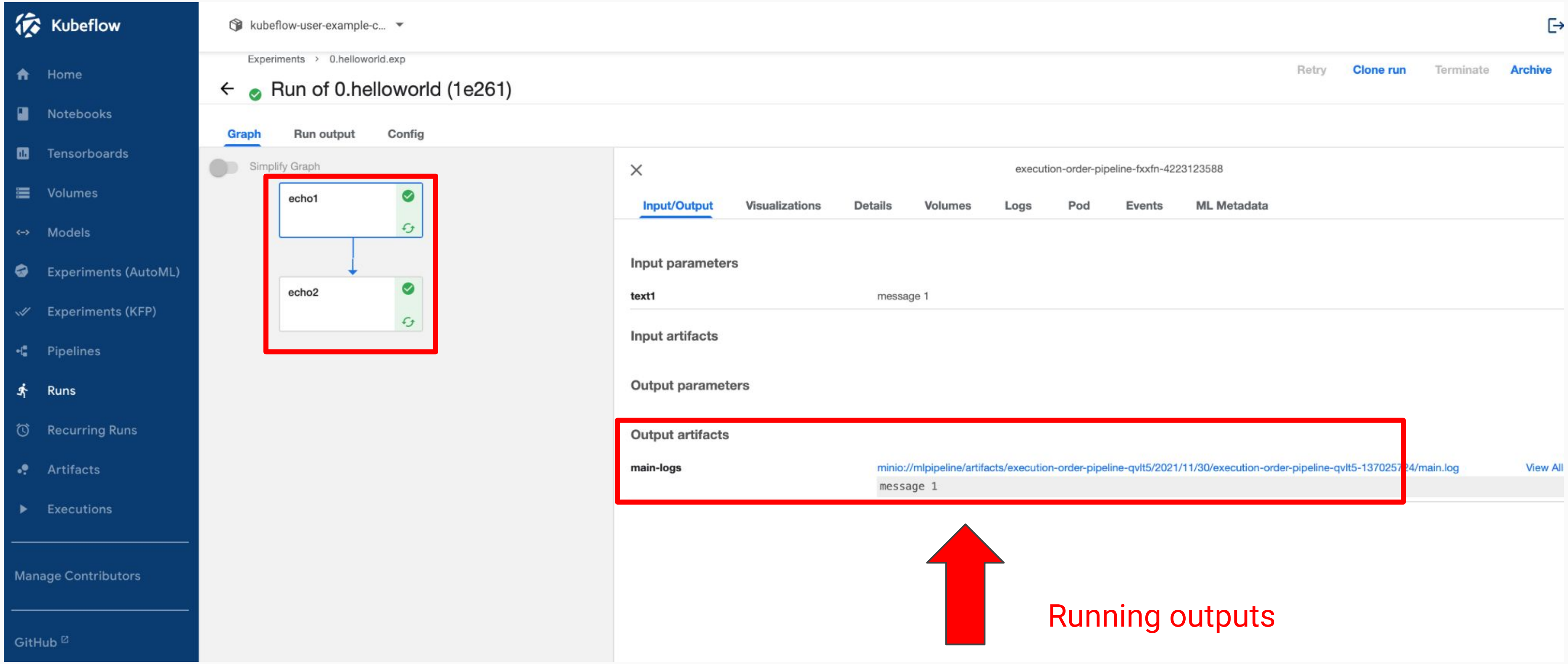
Experiment description

Runs

Run name	Status	Duration	Pipeline Version	Recurring Run	Start time
Run of 0.helloworld (1e261)	?	-	0.helloworld	-	12/2/2021, 4:33:10 PM

Run List

Step4: Create a Pipeline (7/7)



Run of 0.helloworld (1e261)

execution-order-pipeline-fxxfn-4223123588

Input parameters

text1	message 1

Input artifacts

Output parameters

Output artifacts

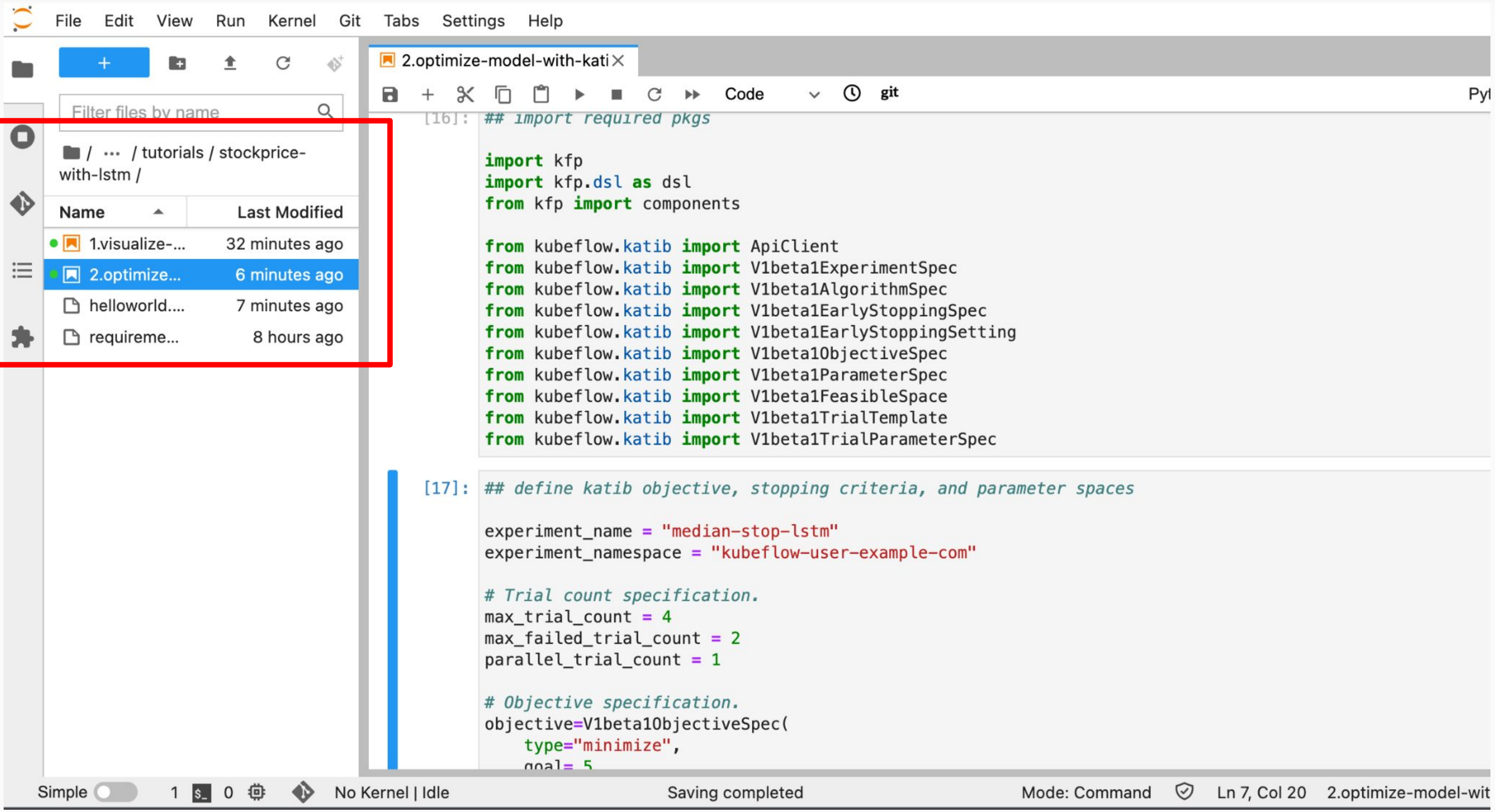
main-logs	minio://mlpipeline/artifacts/execution-order-pipeline-qvlt5/2021/11/30/execution-order-pipeline-qvlt5-1370257?4/main.log

```
message 1
```

Running outputs

Hyperparameter Example

Step5: Hyperparameter tuning with katib (1/4)


File Edit View Run Kernel Git Tabs Settings Help
Filter files by name
/ ··· / tutorials / stockprice-with-lstm /
2.optimize-model-with-kati
Code
git

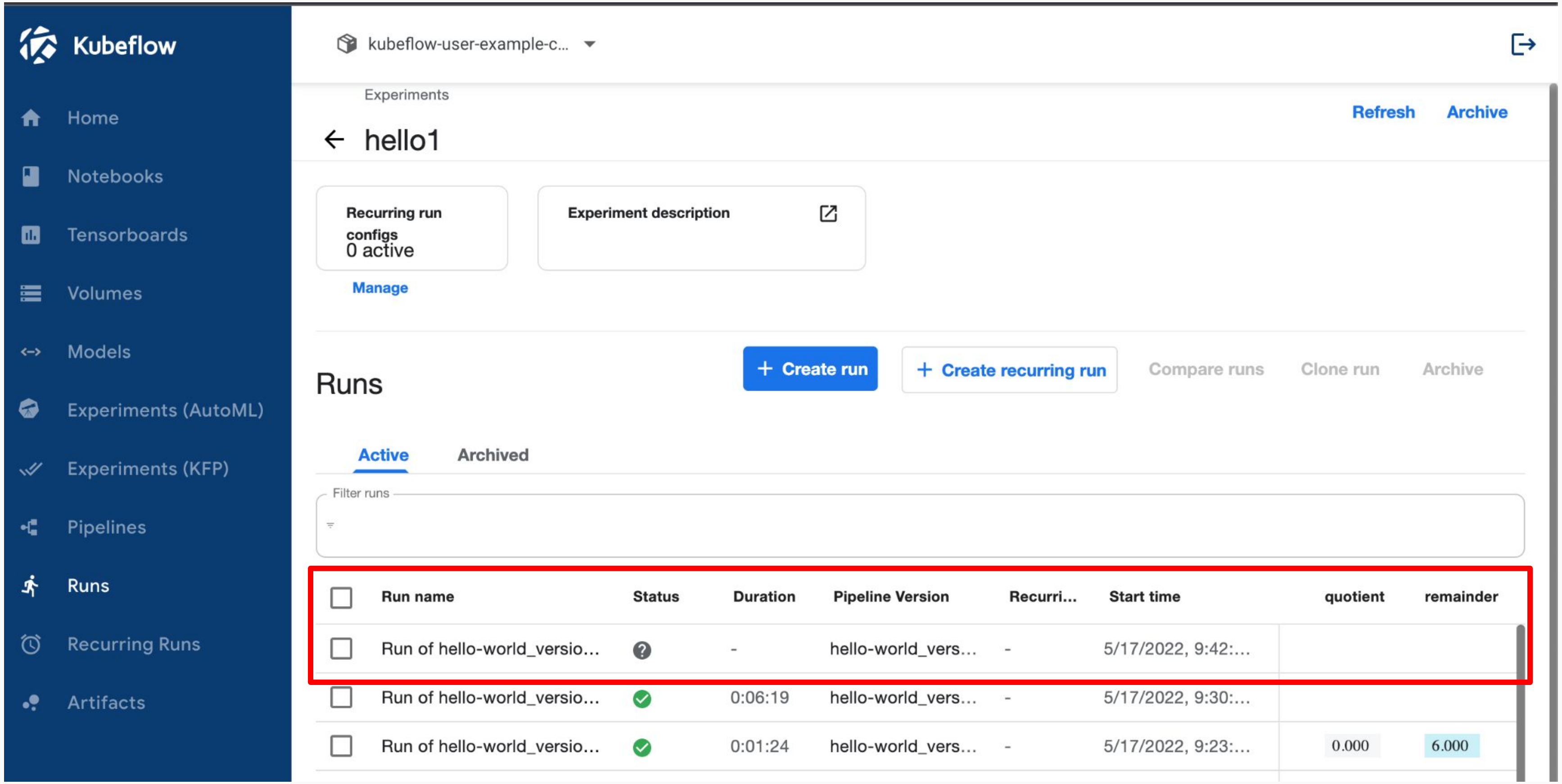
Name	Last Modified
1.visualize-...	32 minutes ago
2.optimize...	6 minutes ago
helloworld....	7 minutes ago
requireme...	8 hours ago

```
[16]: ## import required pkgs

      import kfp
      import kfp.dsl as dsl
      from kfp import components

      from kubeflow.katib import ApiClient
      from kubeflow.katib import V1beta1ExperimentSpec
      from kubeflow.katib import V1beta1AlgorithmSpec
      from kubeflow.katib import V1beta1EarlyStoppingSpec
      from kubeflow.katib import V1beta1EarlyStoppingSetting
      from kubeflow.katib import V1beta1ObjectiveSpec
      from kubeflow.katib import V1beta1ParameterSpec
      from kubeflow.katib import V1beta1FeasibleSpace
      from kubeflow.katib import V1beta1TrialTemplate
      from kubeflow.katib import V1beta1TrialParameterSpec
```

```
[17]: ## define katib objective, stopping criteria, and parameter spaces

      experiment_name = "median-stop-lstm"
      experiment_namespace = "kubeflow-user-example-com"

      # Trial count specification.
      max_trial_count = 4
      max_failed_trial_count = 2
      parallel_trial_count = 1

      # Objective specification.
      objective=V1beta1ObjectiveSpec(
          type="minimize",
          goal= 5
```

Simple 1 0 No Kernel | Idle Saving completed Mode: Command Ln 7, Col 20 2.optimize-model-wit

Step5: Hyperparameter tuning with katib (2/4)



hello1

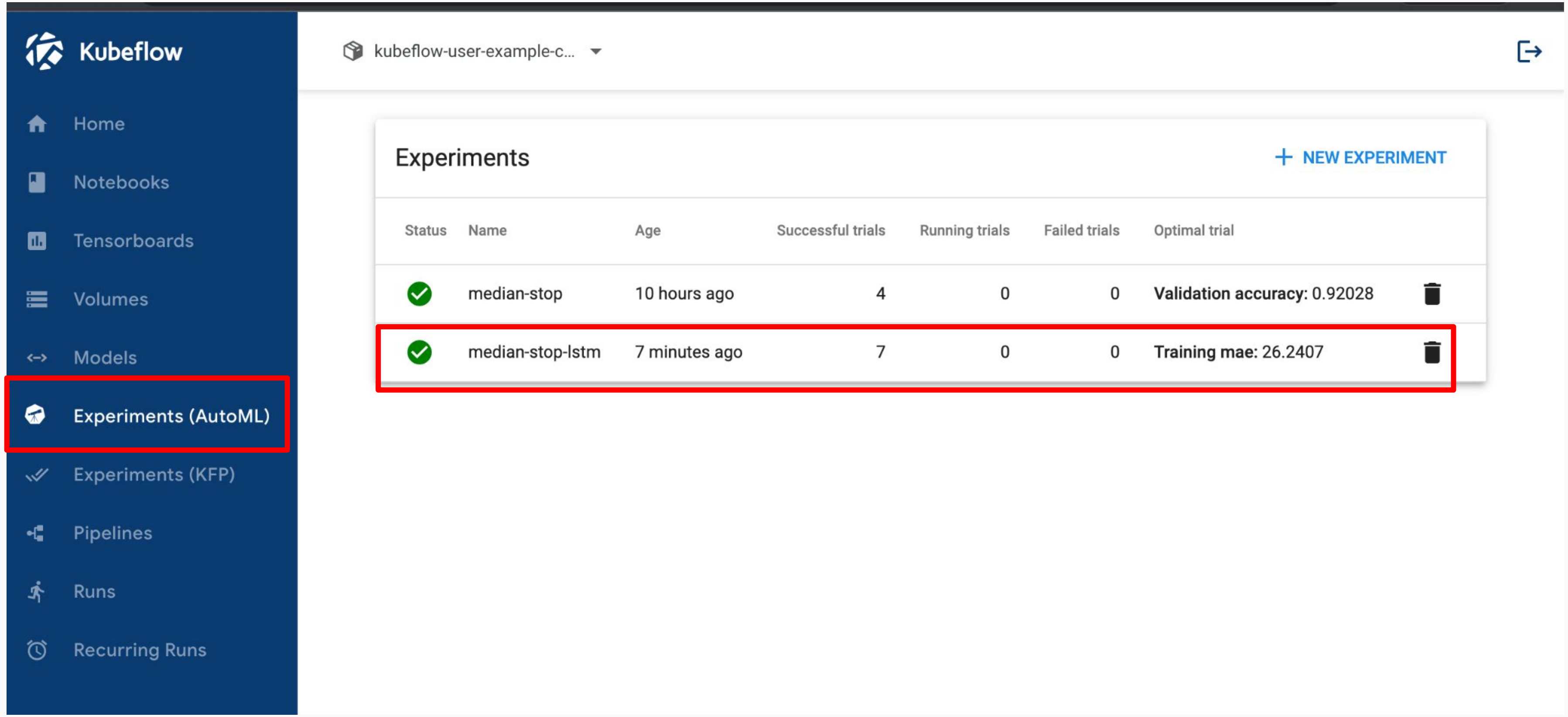
Recurring run configs
0 active

Experiment description

Runs

	Run name	Status	Duration	Pipeline Version	Recurri...	Start time	quotient	remainder
☐	Run of hello-world_versio...	?	-	hello-world_vers...	-	5/17/2022, 9:42:...		
☐	Run of hello-world_versio...	✓	0:06:19	hello-world_vers...	-	5/17/2022, 9:30:...		
☐	Run of hello-world_versio...	✓	0:01:24	hello-world_vers...	-	5/17/2022, 9:23:...	0.000	6.000

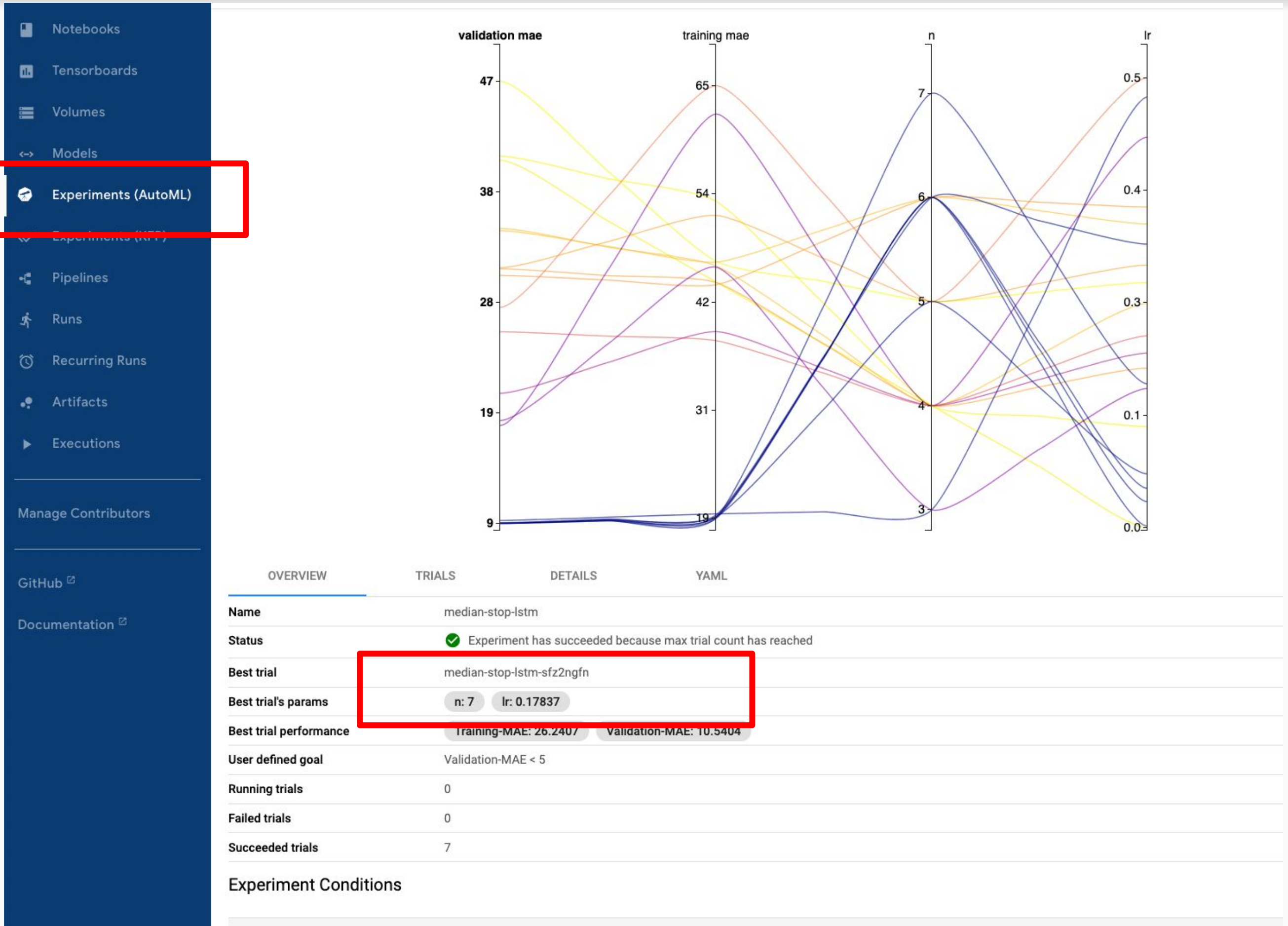
Step5: Hyperparameter tuning with katib (3/4)



Experiments

Status	Name	Age	Successful trials	Running trials	Failed trials	Optimal trial
✓	median-stop	10 hours ago	4	0	0	**Validation accuracy:** 0.92028
✓	median-stop-lstm	7 minutes ago	7	0	0	**Training mae:** 26.2407

Step5: Hyperparameter tuning with katib (4/4)

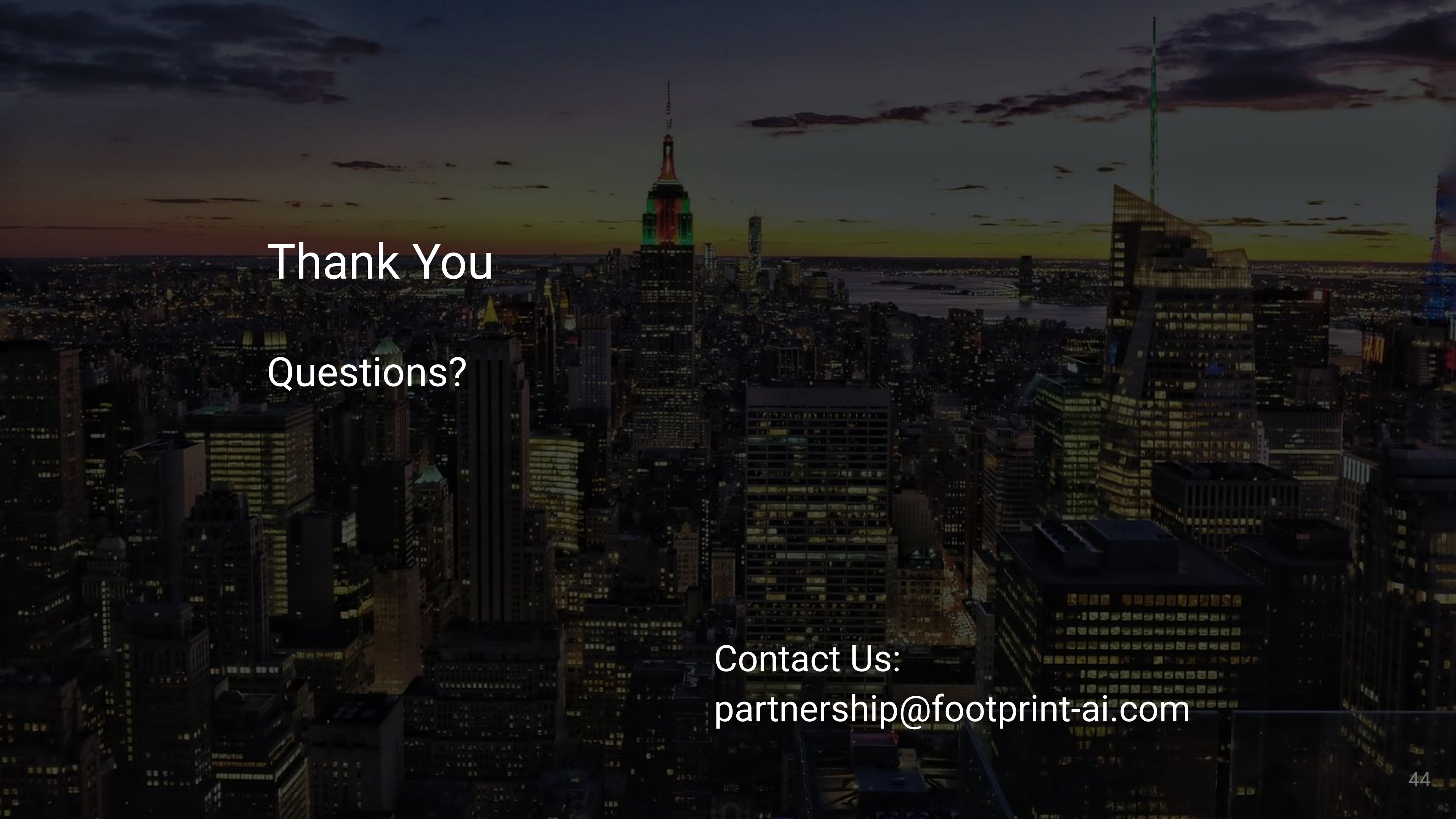


Name	median-stop-lstm
Status	Experiment has succeeded because max trial count has reached
Best trial	median-stop-lstm-sfz2ngfn
Best trial's params	n: 7 lr: 0.17837
Best trial performance	Training-MAE: 26.2407 Validation-MAE: 10.5404
User defined goal	Validation-MAE < 5
Running trials	0
Failed trials	0
Succeeded trials	7

Experiment Conditions

One minute takeaway

- Implemented remote development, training, and deployment.
- Pipeline-based development model
- Distributed model training and optimization



Thank You

Questions?

Contact Us:
partnership@footprint-ai.com