



# Contents

---

1. TUTORIAL LEARNING OBJECTIVES	1
2. PREREQUISITES	1
3. QUICK START	2
4. EXPLORING ADVANCED FEATURES	4
5. TROUBLESHOOTING	5
6. SUMMARY	5
7. LEGEND	5

# 1. Tutorial Learning Objectives

튜토리얼의 목표는 Samsung Cloud Platform에서 Kubeflow Mini를 사용하여 머신러닝 Workflow를 실행하는 Pipeline Instance 생성과 Experiment를 구성하는 것입니다.

① Kubeflow Mini는 오픈소스 Kubeflow vanilla version(v1.7)을 제공하는 Samsung Cloud Platform 상품으로, Pipeline과 AutoML의 관련한 튜토리얼은 Kubeflow Mini를 기반으로 진행합니다.

## 2. Prerequisites

튜토리얼 진행에 필요한 사전요건은 아래와 같습니다.

- Kubeflow Mini 또는 AI&MLOps Platform에 대한 이해
- Kubernetes에 대한 이해
- YAML 파일 형식 및 사용법에 대한 이해
- Python 프로그래밍에 대한 이해
- "AI&MLOps Platform에서 Jupyter notebook 생성하기" 튜토리얼 참조

### 2.1 Consult the Kubeflow Documentation

✓ Kubeflow Documentation (공식문서) 참조를 권장합니다.

→ 아래 순서에 따라 이동합니다

① [www.kubeflow.org](http://www.kubeflow.org)로 이동합니다.

② 메인 페이지에서 "Get Started →" 버튼을 클릭합니다.

③ Kubeflow Documentation의 메인 페이지로 이동하였습니다.

→ Kubeflow Documentation에서 아래 순서에 따라 이동합니다.

① "Components"를 선택합니다.

② "Kubeflow Pipelines"을 선택합니다.

③ "v2"를 선택합니다.

④ 관심분야에 따른 상세 이동 위치를 아래에서 선택합니다.

- Jupyter Notebook을 사용하여 Kubeflow Pipelines (kfp) Python SDK를 편집하는 방법은: "Pipelines > Pipeline Basics"로 이동합니다.

- Pipeline을 실행하는 방법은: "Data Types > Run a Pipeline"로 이동합니다.

✓ 튜토리얼은 Kubeflow UI를 통해 Pipeline 생성과 Experiment 구성을 진행합니다. 따라서, "Data Types > Run a Pipeline"의 Documentation을 참조하고 있습니다.

## 2.2 Secure Necessary Permissions

✓ 튜토리얼 진행을 위한 Kubeflow Mini 사용자 권한을 확인합니다.

✓ 필요시 Kubeflow Mini 관리자를 통해 권한을 확보합니다.

## 3. Quick Start

Kubeflow Mini에서 Pipeline Instance를 생성하여 Experiment로 실행하는 과정을 학습합니다.

✓ ML Pipeline은 머신러닝(ML) 모델 개발, 훈련, 배포 과정을 연속으로 구성함으로써 이 과정을 효율화하고, MLOps 환경에서 재현성, 확장성 및 체계적인 모델 관리를 가능하게 합니다.

➤ MLOps(Machine Learning Operations)는 머신러닝의 개발(ML)과 운영(Ops)단계를 결합시키고, 모델 개발, 훈련, 배포, 모니터링 및 유지 보수를 포함하는 머신러닝 프로젝트의 전체 수명 주기동안 워크플로우의 표준화, 자동화 및 간소화를 목표로 합니다.

## 3.1 Preliminary Checks

✓ Pipeline Instance 생성에 충분한 리소스를 확보합니다.

① Kubeflow Mini 리소스 부족 시 Kubernetes Engine의 node pool을 추가하여 리소스를 확보할 수 있습니다.

## 3.2 Creating a New Pipeline Instance

➤ Kubeflow에서 pipeline은 머신러닝(ML) 워크플로우에서 관련 작업을 자동화하기 위해 사용할 수 있는 연속된 단계입니다. Pipeline의 각 단계를 작업(task)이라고 부르며, 작업은 병렬 또는 순차적으로 실행될 수 있습니다. Pipeline은 데이터 전처리, 모델 훈련, 모델 평가와 같은 다양한 ML 작업을 수행하는 데 사용될 수 있습니다.

➤ Experiment(실험)는 pipeline의 다양한 구성을 시도할 수 있는 작업 공간입니다. Experiment을 통해 다양한 pipeline의 결과를 비교하여 특정 요구사항에 가장 적합한 구성을 찾거나, 기존 pipeline을 진단하거나 실행이력을 추적할 수 있습니다.

✓ Kubeflow Mini에서 pipeline instance를 experiment로 실행하는 것은 직관적으로 진행됩니다. 아래의 순서로 진행하겠습니다.

→ "Pipelines" 섹션에 접근하기

Kubeflow Mini의 대시보드 사이드바의 "Pipelines"을 클릭합니다.

→ Pipeline 선택하기

Pipeline 목록에서 "[Tutorial] DSL - Control structures"을 선택합니다.

① 튜토리얼에서 사용하는 pipeline은 Kubeflow Mini가 기본으로 제공하는 pipeline을 재활용합니다. 새로운 pipeline을 생성하기 위해서는 kfp를 Jupyter Notebook에 설치하고, Kubernetes의 PodDefault 설정을 추가해야 합니다. (4. 고급기능 참조)

→ Pipeline 이름을 클릭하여 상세 정보를 확인합니다.

Pipeline 상세 페이지는 그래프 구조, YAML 코드 및 SUMMARY를 제공합니다.

→ 새로운 Experiment 생성하기

① "Pipelines" 페이지 우측 상단의 "+ Create experiment" 버튼을 클릭합니다.

② "Create Experiment" 페이지로 이동하게 됩니다.

→ Experiment Instance 세부 정보 입력하기

① Experiment Name: "MyFirst-Experiment"을 입력합니다.

② Description (Optional): "Tutorial Experiment"을 입력합니다.

→ 생성된 Experiment 실행(Run)하기

① "Next"를 클릭합니다.

② "Start a Run" 페이지로 이동하게 됩니다.

③ Description (Optional): "Tutorial Run"을 입력합니다.

④ "Start" 버튼을 클릭하여 Experiment를 실행(Run)합니다.

→ 실행(Run) 상태 모니터링하기:

- 리다이렉션(redirectation)된 "Runs" 페이지에서 experiment의 실행 상태 및 진행 상황을 추적할 수 있습니다.

- Experiment 리스트에서 "Run name" 필드를 클릭하여 상세정보를 확인합니다.

- 상세정보 페이지에서는 실행 중(Running)인 Pipeline의 워크플로우(DAG) 상태를 시각화하여 제공하고, 실행이 완료된 Experiment를 이력(Archive)으로 저장합니다.

➤ DAG는 Directed Acyclic Graph의 약자로, 각 노드를 연결하는 경로가 순환하지 않도록 배열된 그래프를 의미합니다. DAG에서 각 노드는 작업(task)을 나타내고 노드를 연결하는 경로가 작업 간의 종속성을 나타내어 워크플로우 구성에 활용됩니다.

## 4. Exploring Advanced Features

- ✓ Jupyter Notebook을 Kubeflow Pipelines (kfp) Python SDK와 연계하면 AI&MLOps Platform 또는 Kubeflow Mini 활용 범위를 확대할 수 있습니다. 하지만 kfp Python SDK가 제공하는 kfp.client() 기능 등을 Jupyter Notebook에서 사용하기 위해서는 Kubernetes의 설정을 아래와 같이 변경해야 합니다.

- Jupyter Notebook Instance가 Kubernetes의 pipeline SVC pod에 연결될 수 있도록 Token과 PodDefault 속성을 YAML 파일을 통해 설정해야 합니다.

- 아울러 사용자가 kfp의 고급구성과 설정을 사용할 수 있도록, sudo 권한이 주입된 Jupyter Notebook Instance 활용이 필요할 수 있습니다.

① 이를 해결하기 위해, Jupyter Notebook Image에 sudo 권한주입, Kubernetes의 Pipeline SVC pod설정과 kfp Python SDK 활용을 동영상으로 제작할 예정입니다.

## 5. Troubleshooting

Kubeflow Mini 사용오류를 최소화하는 방안은,

- ✓ 다양한 오픈소스 환경에 따른 컴포넌트 간 버전관리를 철저히 하는 것
- ✓ 사용자와 자원의 권한/인증에 대한 정확한 정보를 유지하는 것
- ✓ 사용자 가이드 등 공식문서를 사전에 참조하여 적용하는 것입니다.

## 6. Summary

- ✓ Kubeflow Mini에서 머신러닝(ML) Pipeline을 생성하여 활용하는 것은 복잡한 머신러닝 작업 흐름을 자동화하고 관리하기 위한 전략적인 방법입니다. Jupyter Notebook과 kfp Python SDK가 결합된 Kubeflow Mini 플랫폼은 머신러닝 프로젝트에서 효율성, 확장성, 재현성을 보장합니다.

## 7. Legend

[✓] Tip

[→] 따라하기

[🚀] Note

[①] Notice

[>] Definition or Terminology

[⚠] Warning

[★] See User Guide or Documentation

[•] 순서가 없는 리스트

[①~⑮] 순서가 있는 리스트