Executing a Sample Pipeline

1. Click the name **[Sample] Basic - Condition** (see Figure 46-7).

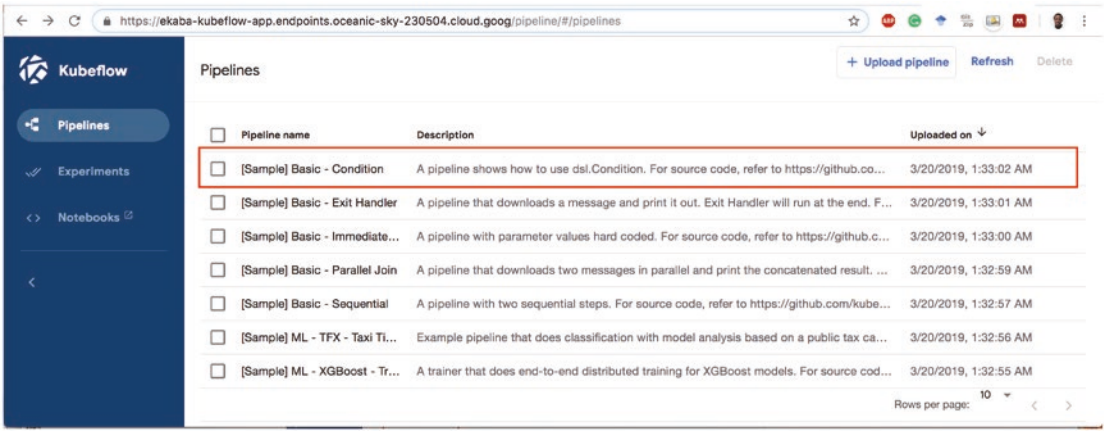

Figure 46-7. *Select a Pipeline*

2. Click **Start an Experiment** (see Figure 46-8).

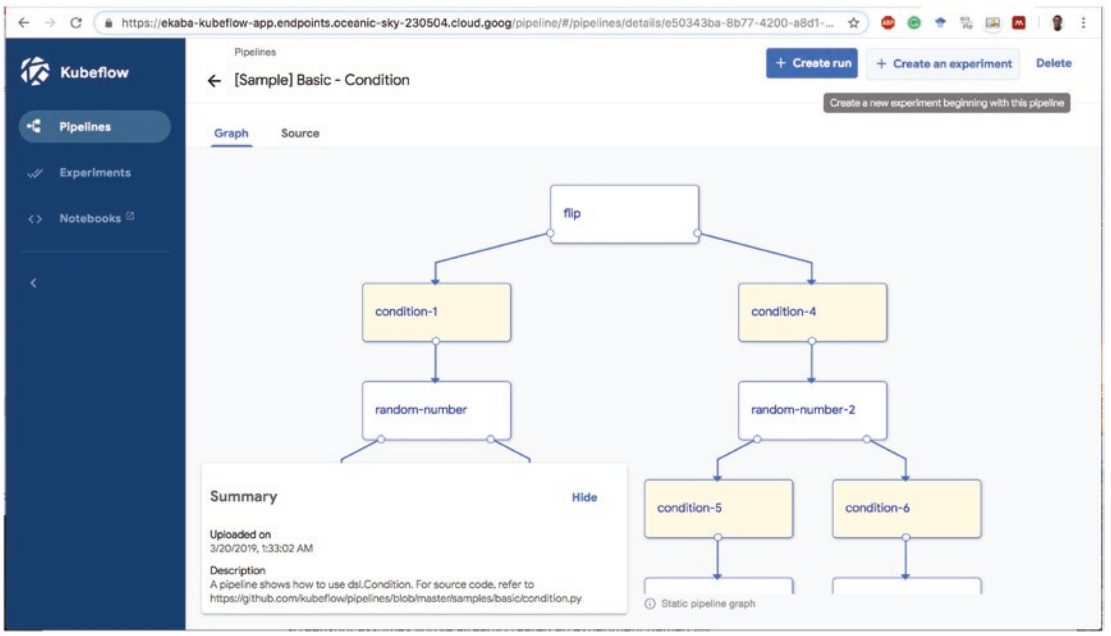

Figure 46-8. *Create a new Experiment*

3. Give the Experiment a name (see Figure 46-9).

Figure 46-9. *Assign a name to the Experiment*

4. Give the run a name (see Figure 46-10).

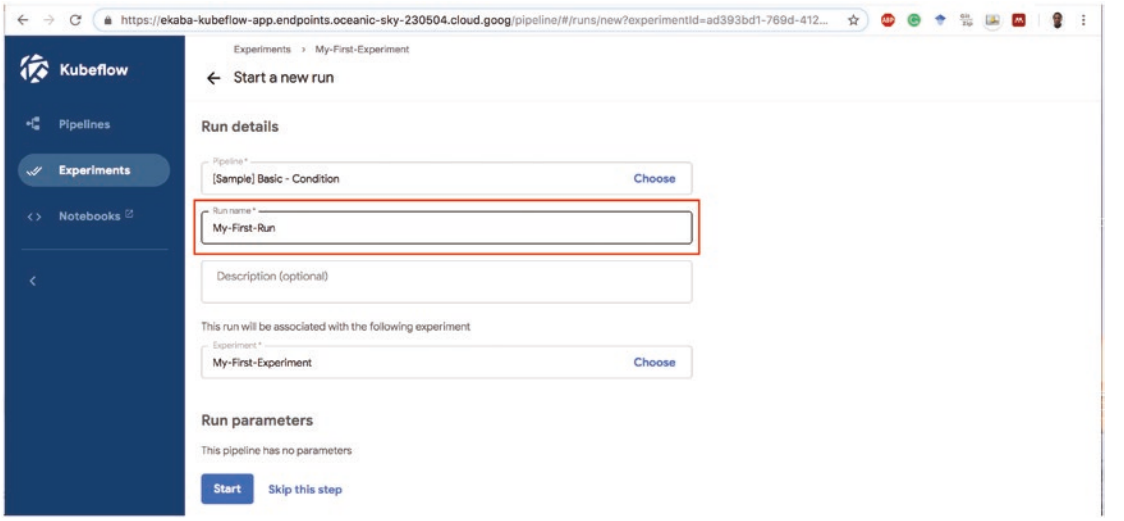

Figure 46-10. *Assign a name to the run*

5. Click the **Run Name** to start the run (see Figure 46-11).

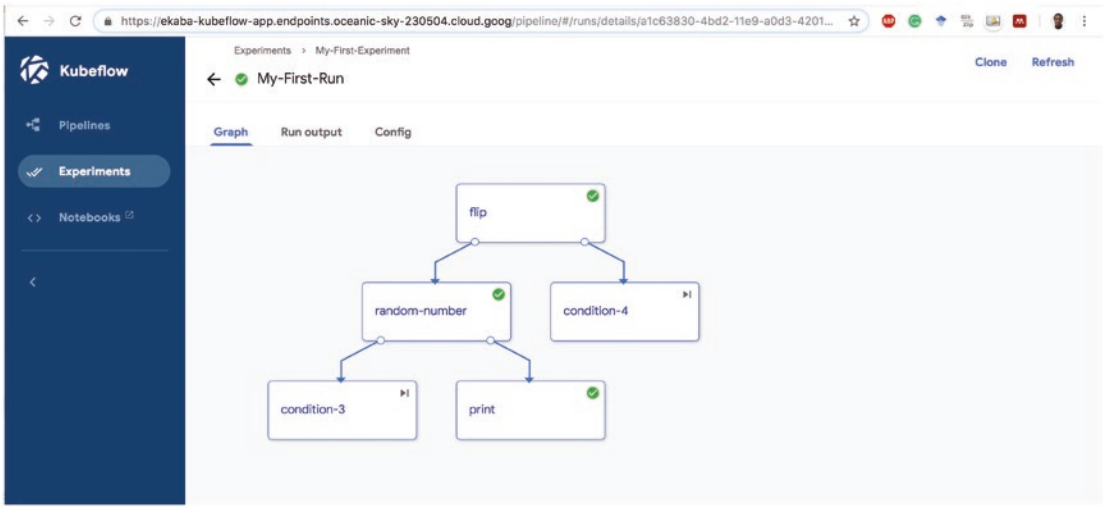

Figure 46-11. *Run the Pipeline*

Note Always remember to clean up cloud resources when they are no longer needed.

This chapter covered setting up Kubeflow on Kubernetes and introduced working with Kubeflow Pipelines to manage containerized machine learning workflows. The next chapter will deploy an end-to-end machine learning solution with Kubeflow Pipelines.

CHAPTER 47

Deploying an End-to-End Machine Learning Solution on Kubeflow Pipelines

A Kubeflow pipeline component is an implementation of a pipeline task. A component is a step in the workflow. Each task takes one or more artifacts as input and may produce one or more artifacts as output.

Each component usually includes two parts:

- Client code: The code that talks to endpoints to submit jobs, for example, code to connect with the Google Cloud Machine Learning Engine.
- Runtime code: The code that does the actual job and usually runs in the cluster, for example, the code that prepares the model for training on Cloud MLE.

A component consists of an interface (inputs/outputs), the implementation (a Docker container image and command-line arguments), and metadata (name, description).

E. Bisong, *Building Machine Learning and Deep Learning Models on Google Cloud Platform*,
https://doi.org/10.1007/978-1-4842-4470-8_47