This shows how to create a model with Keras but customize the training loop. This way you get the benefit of writing a model in the simple Keras API, but still retain the flexibility by allowing you to train the model with a custom loop.
(https://colab.research.google.com/github/tensorflow/tpu/blob/master/tools/colab/custom_training.ipynb)

This is a canonical end-to-end TPU sample in Keras, featuring data loading with tf.data.Dataset, the Keras model, TPU training, TPU inference and also trained model export to the Tensorflow standard "saved model" format, model deployment to ML Engine, and predictions from the cloud-deployed model.
(https://colab.research.google.com/github/tensorflow/tpu/blob/master/tools/colab/keras_mnist_tpu.ipynb)

This notebook demonstrates an end-to-end image classification sample with data loading, TPU training, model export, and deployment.
(https://colab.research.google.com/github/tensorflow/tpu/blob/master/tools/colab/fashion_mnist.ipynb)

This notebook demonstrates how to port a MNIST Estimator model to TPUEstimator. Every line that needed changing is commented. The baseline Estimator model is in the "MNIST with Estimator" notebook.
(https://colab.research.google.com/github/tensorflow/tpu/blob/master/tools/colab/mnist_tpuestimator.ipynb)

This is a baseline before porting to TPUEstimator. The ported version is in the "MNIST Estimator to TPUEstimator" notebook.
(https://colab.research.google.com/github/tensorflow/tpu/blob/master/tools/colab/mnist_estimator.ipynb)

This colab explores how to train autoencoders on a TPU device.
(http://colab.research.google.com/github/tensorflow/tpu/blob/master/tools/colab/autoencoder_embeddings.ipynb)

This notebook demonstrates how to train, evaluate, and generate predictions using Keras with Cloud TPUs. It uses the iris dataset to predict the species of the flower and also shows how to use your own data instead of using pre-loaded data.
(https://colab.research.google.com/github/tensorflow/tpu/blob/master/tools/colab/classification_iris_data_with_keras.ipynb)

This notebook demonstrates how to train, evaluate, and generate predictions using TPUEstimator with Cloud TPUs. It uses the iris dataset to predict the species of the flower and also shows how to use your own data instead of using pre-loaded data.
(https://colab.research.google.com/github/tensorflow/tpu/blob/master/tools/colab/classification_iris_data_with_tpuestimator.ipynb)

This notebook uses Keras to build a language model and train it on a Cloud TPU. This language model predicts the next character of text given the text so far. The trained model can generate new snippets of text that read in a similar style to the text training data.
(https://colab.research.google.com/github/tensorflow/tpu/blob/master/tools/colab/shakespeare_with_tpu_and_keras.ipynb)

This notebook uses TPUEstimator to build a language model and train it on a Cloud TPU. This language model predicts the next character of text given the text so far. The trained model can generate new snippets of text

that read in a similar style to the text training data.
(https://colab.research.google.com/github/tensorflow/tpu/blob/master/tools/colab/shakespeare_with_tpuestimator.ipynb)

This notebook demonstrates using Cloud TPUs to build a simple regression model using **y = sin(x)** to predict y for given x.
(https://colab.research.google.com/github/tensorflow/tpu/blob/master/tools/colab/regression_sine_data_with_keras.ipynb)

This notebook demonstrates using a free Colab Cloud TPU to fine-tune sentence and sentence-pair classification tasks built on top of pretrained BERT models.
(https://colab.research.google.com/github/tensorflow/tpu/blob/master/tools/colab/bert_finetuning_with_cloud_tpus.ipynb)

This notebook uses TPUs to train a GAN on the CIFAR10 dataset. The model learns to generate images of airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, and trucks.
(https://colab.sandbox.google.com/github/tensorflow/gan/blob/master/tensorflow_gan/examples/colab_notebooks/tfgan_on_tpus.ipynb)

This colab example corresponds to the implementation under test_train_mnist.py and is TF/XRT 1.15 compatible.
(https://colab.sandbox.google.com/github/pytorch/xla/blob/master/contrib/colab/mnist-training-xrt-1-15.ipynb)

This colab example corresponds to the implementation under test_train_cifar.py and is TF/XRT 1.15 compatible.

(https://colab.sandbox.google.com/github/pytorch/xla/blob/master/contrib/colab/resnet18-training-xrt-1-15.ipynb)

This colab example is TF/XRT 1.15 compatible.
(https://colab.sandbox.google.com/github/pytorch/xla/blob/master/contrib/colab/resnet50-inference-xrt-1-15.ipynb)