Week 7 경현

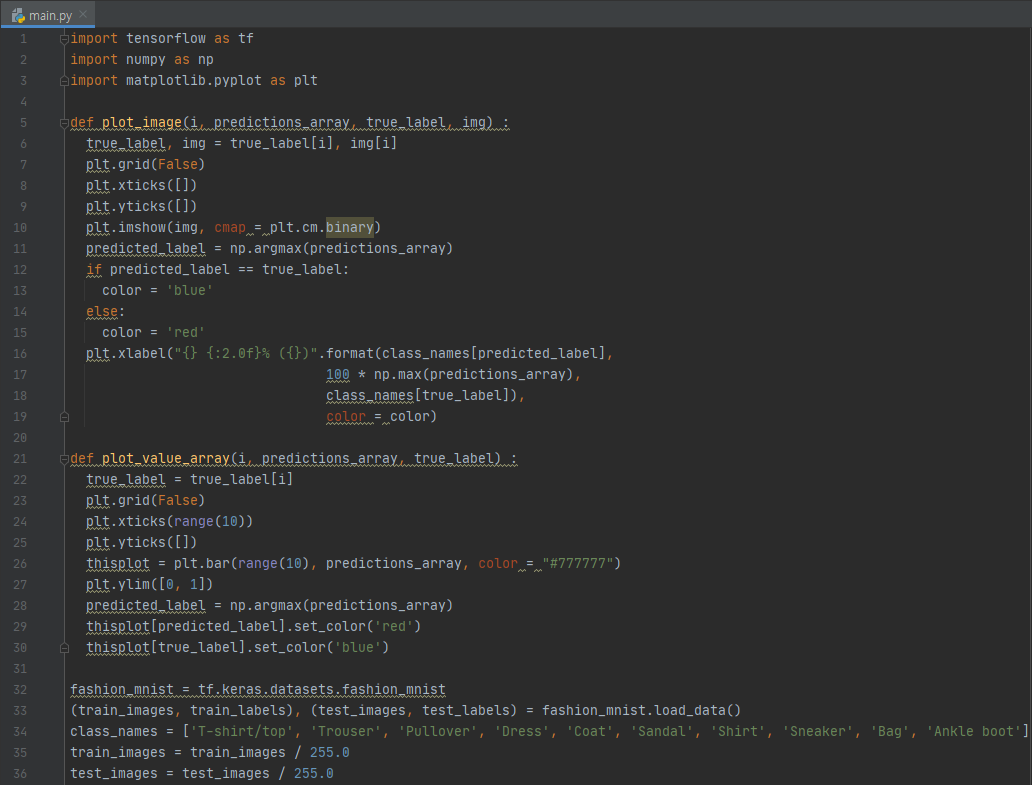
This week’s task:

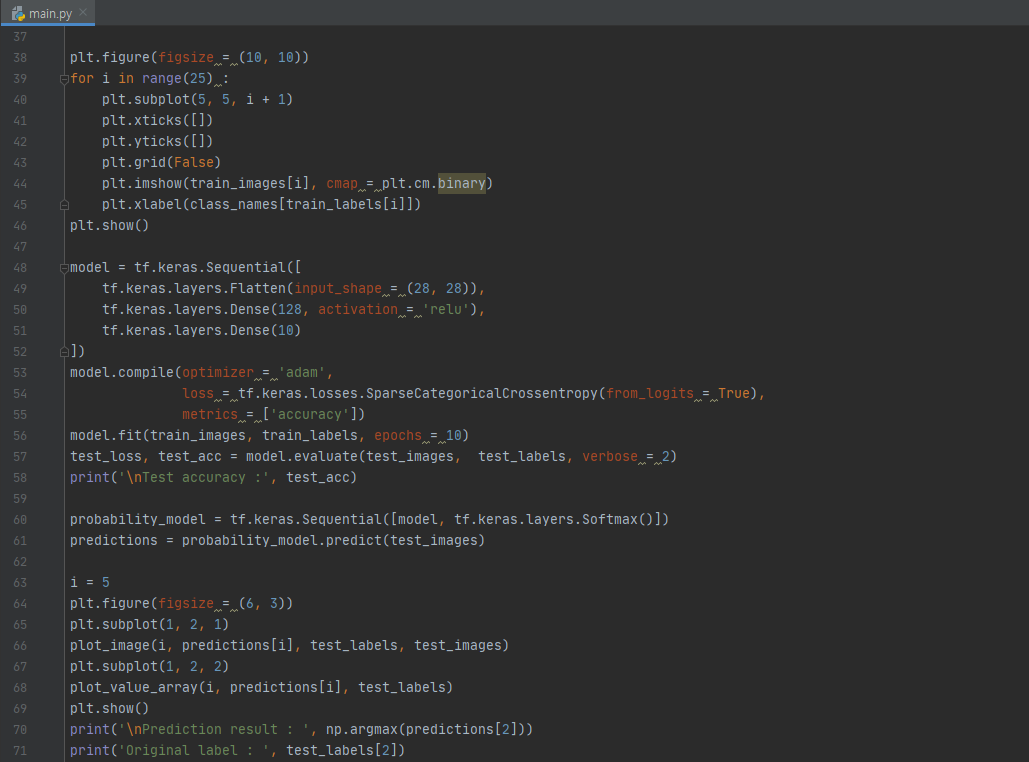
**[Task 1] Basic image classification neural network model training using python TensorFlow keras API**

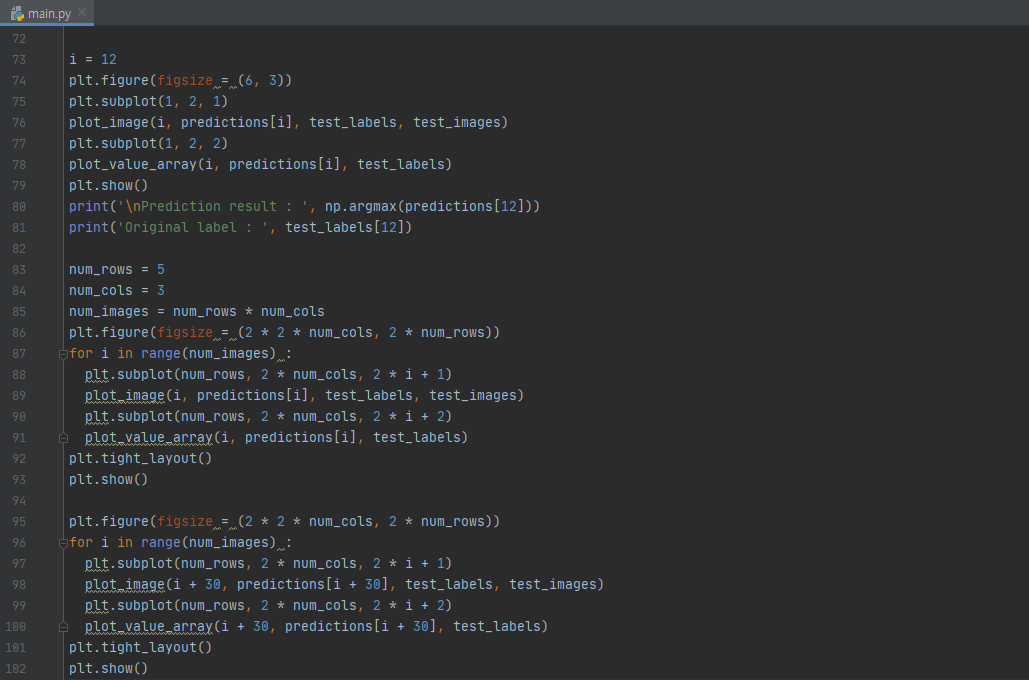
1. Overview

To make a fashion recommendation application, we trained a neural network to classify basic images using TensorFlow keras API. Through this, we will implement a function for the application to identify clothes as accurately as possible. However, since this is done with a basic image, we will develop a process for detecting specific clothes in a colored image in a later task.

2. Application code







3. Code detail & Interpretation of result

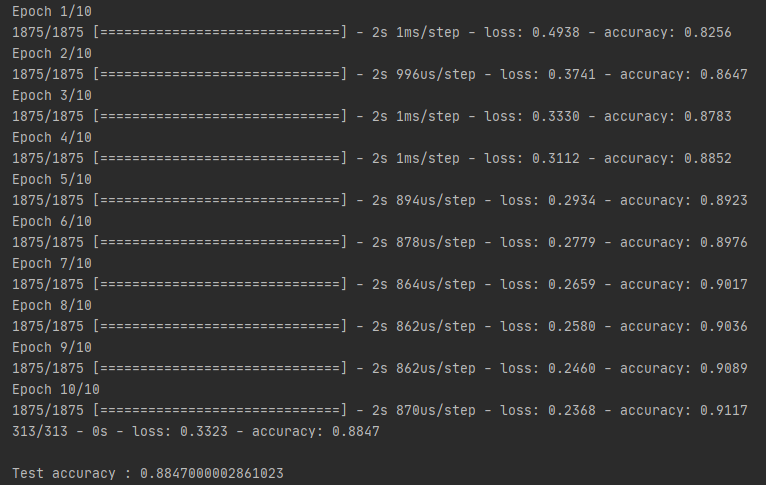
**Line 32~36, 38~46**

To understand how tf.keras works, we use 60000 images of fashion MNIST. The figure below shows the first 25 images of the train set.

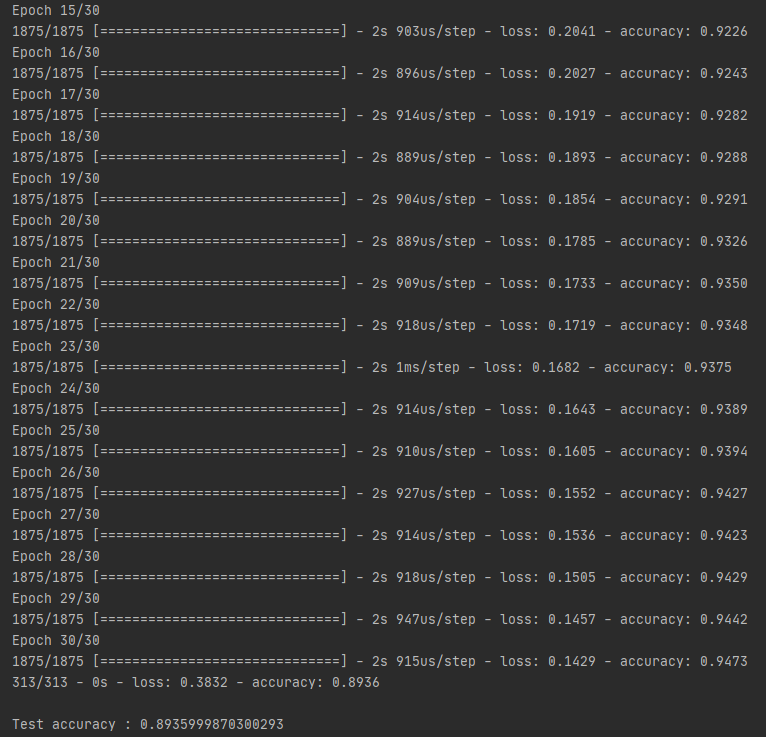


**Line 48~61**

Layer configuration, model compile, and model training were carried out to create a neural network model. As the epoch progressed, the accuracy increased.

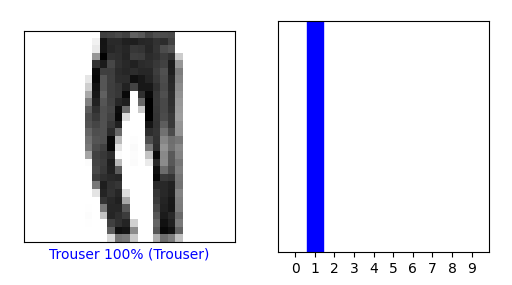


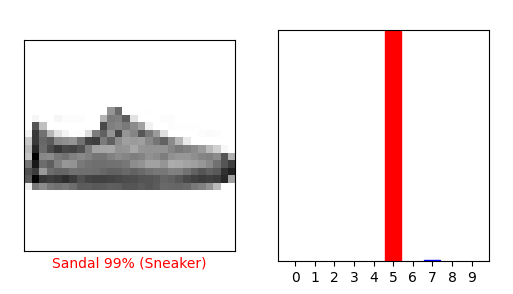
The accuracy of 30 runs was higher than the accuracy of 10 runs.

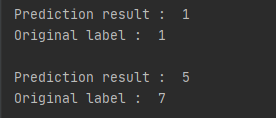


**Line 63~71, 73~81**

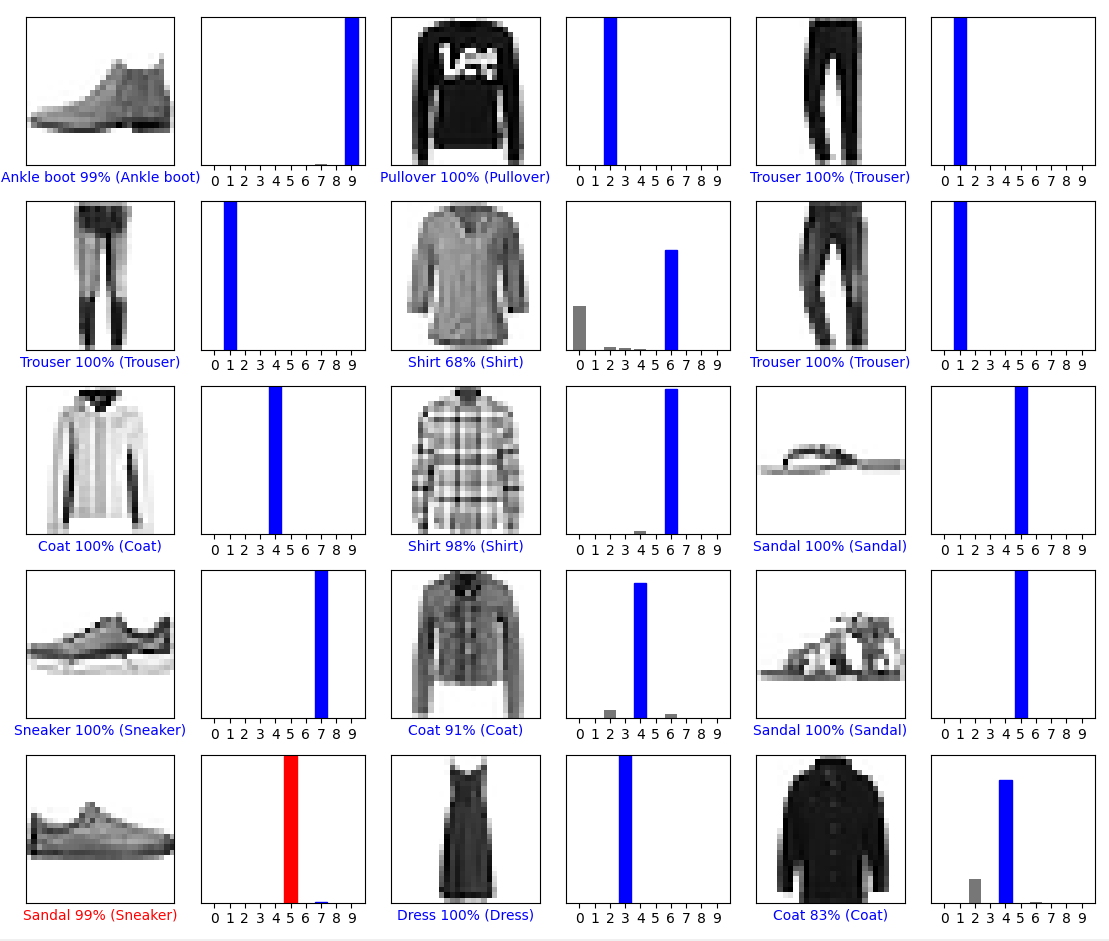
Through the trained model, the results of image 5 and 12 were derived. In image 5, the predicted result and the original label were the same, but in image 12, the result was different. Numbers 0 to 9 in the chart are cloth types that were set as initial labels.

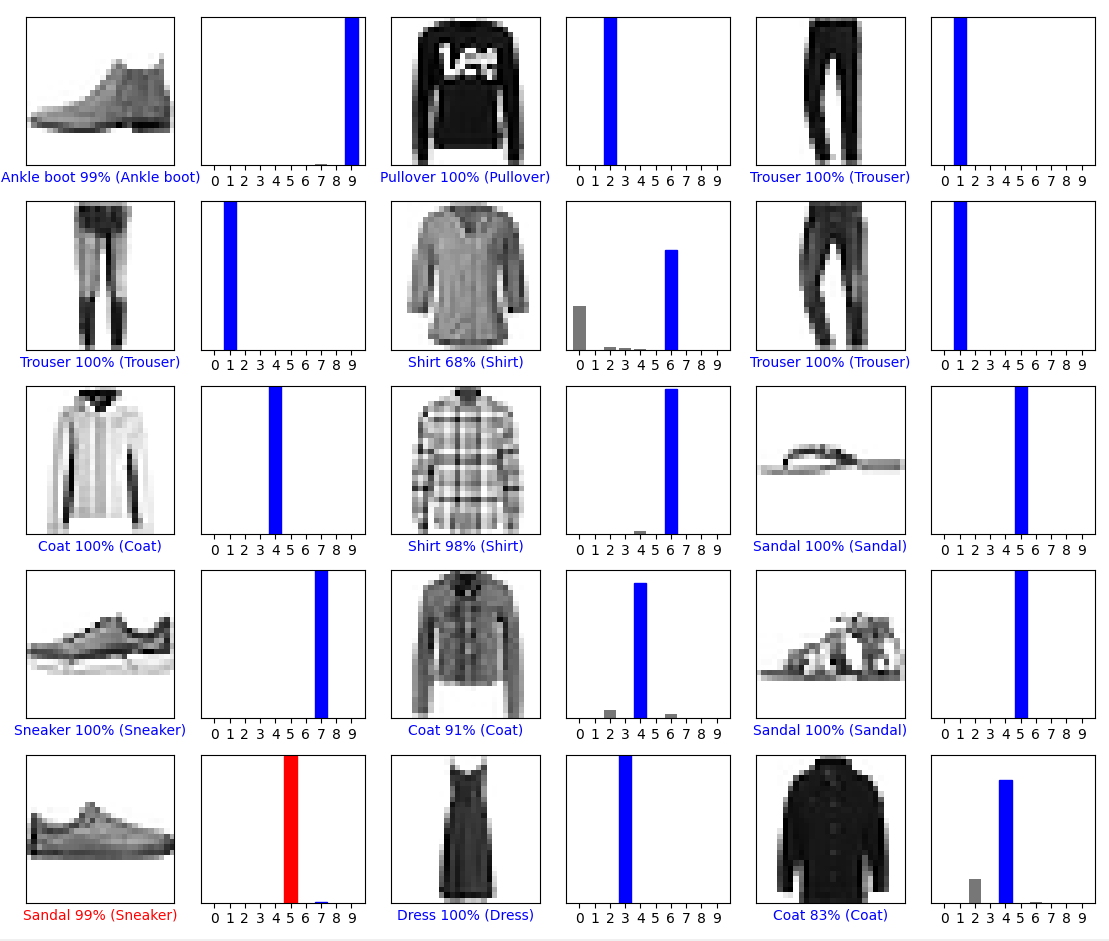






**Line 83~93, 95~102**

The figures below are the results from image 1 to 15 and image 30 to 45, respectively. 

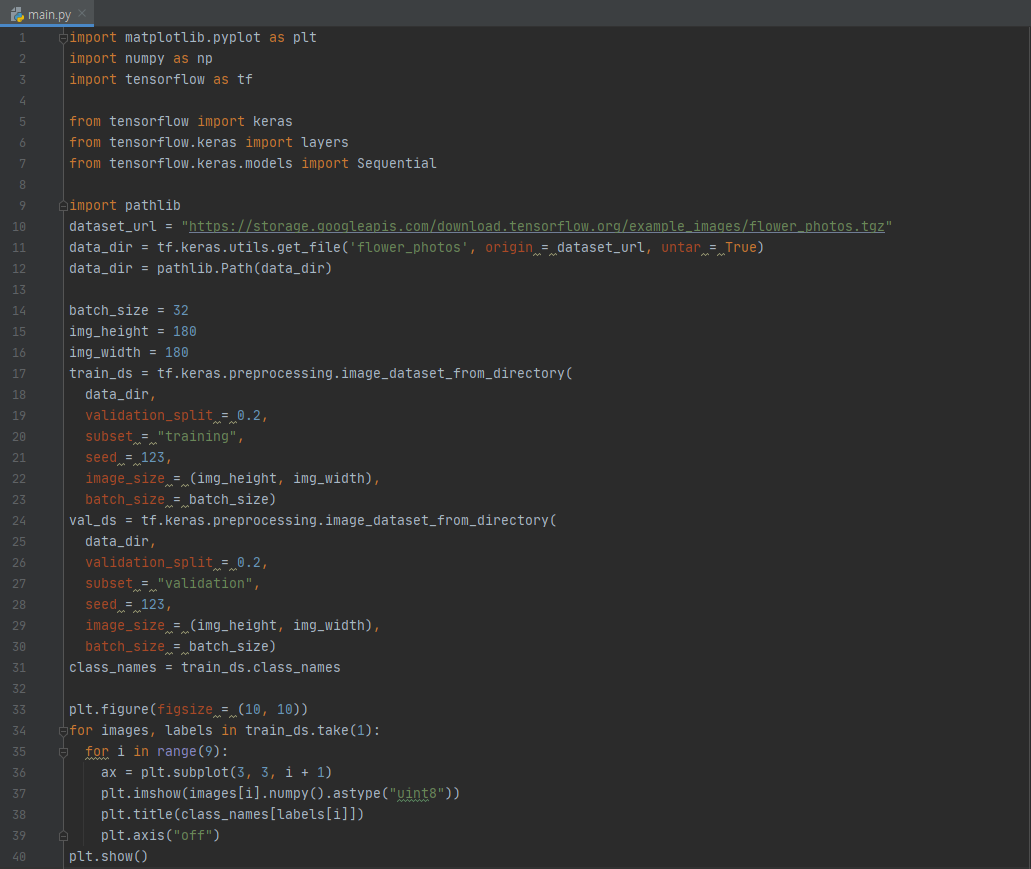


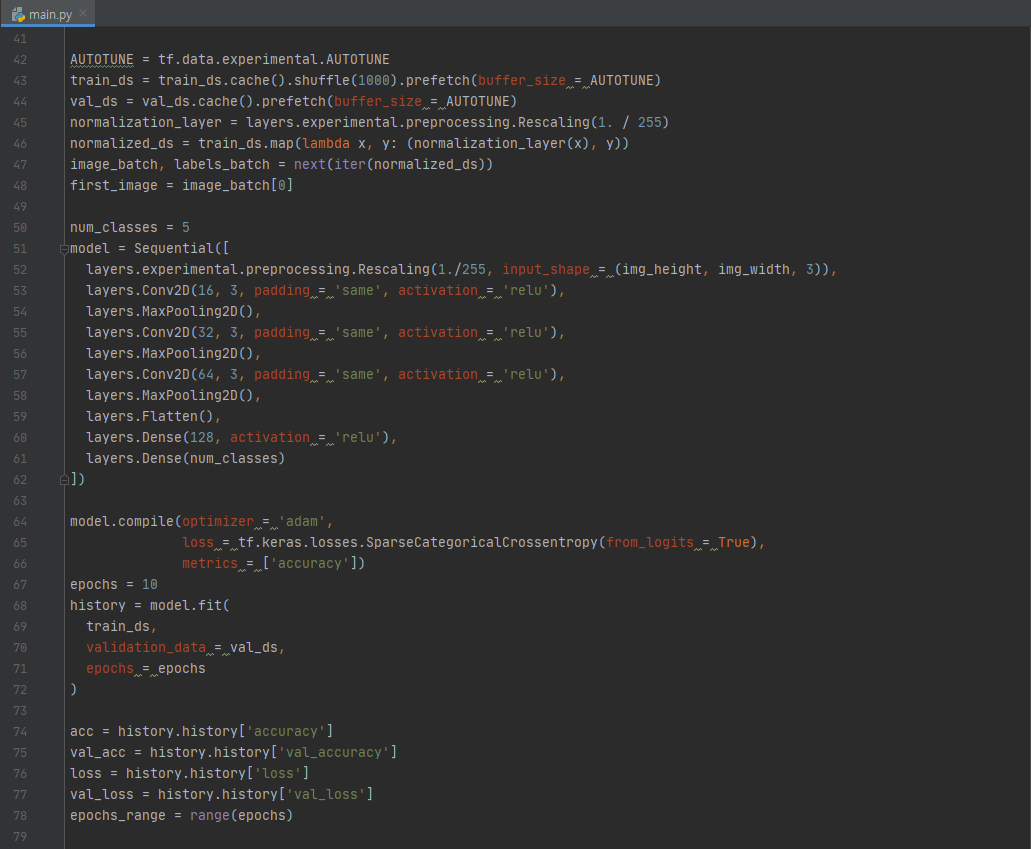
**[Task 2] Training a neural network model through data augmentation**

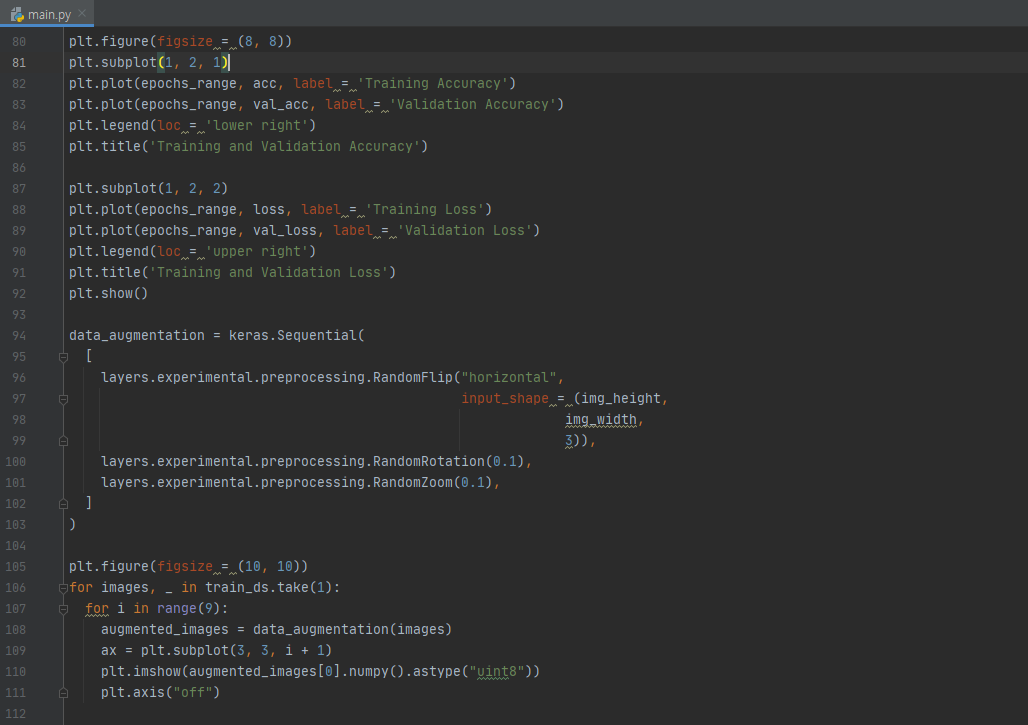
1. Overview

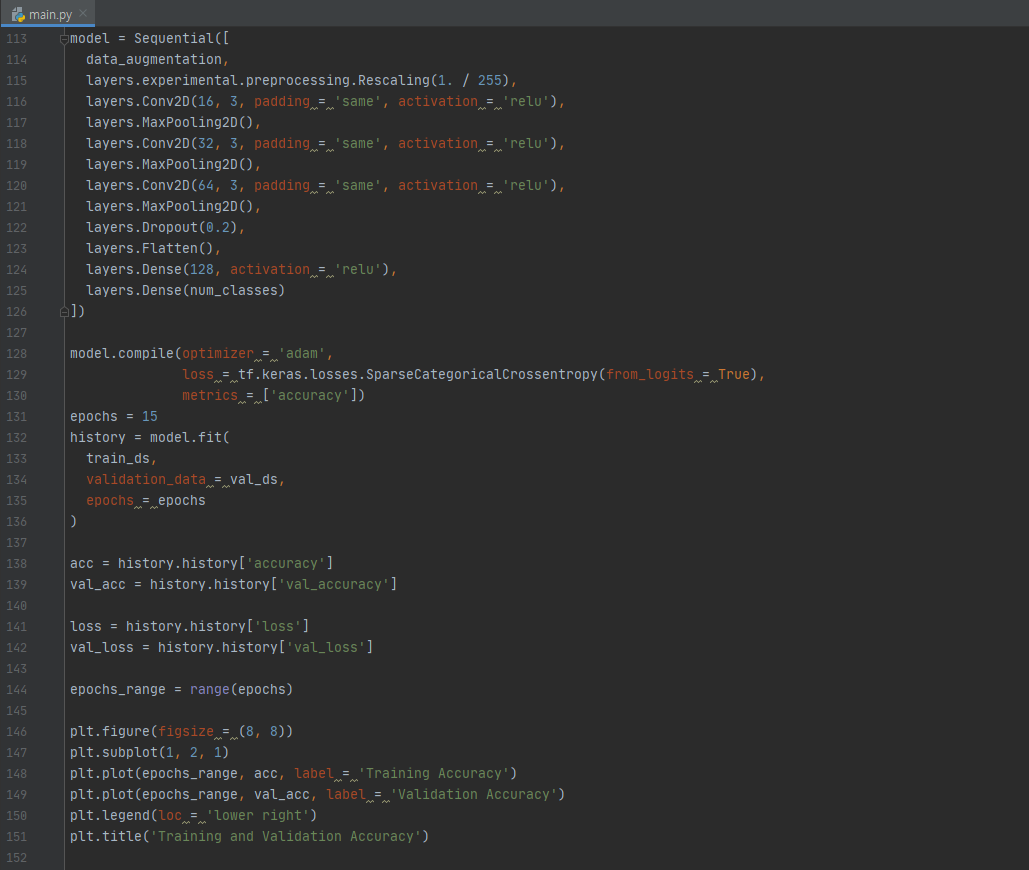
In task 1, we created a data classification neural network model using a basic image. In this task, we will create a neural network model using data augmentation of colored flower images and check how much the accuracy increases.

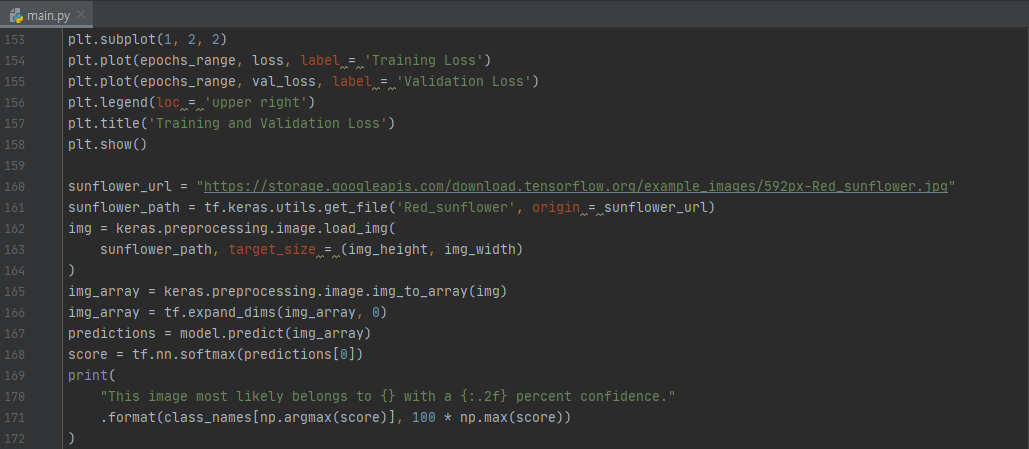
2. Application code











3. Code detail & Interpretation of result

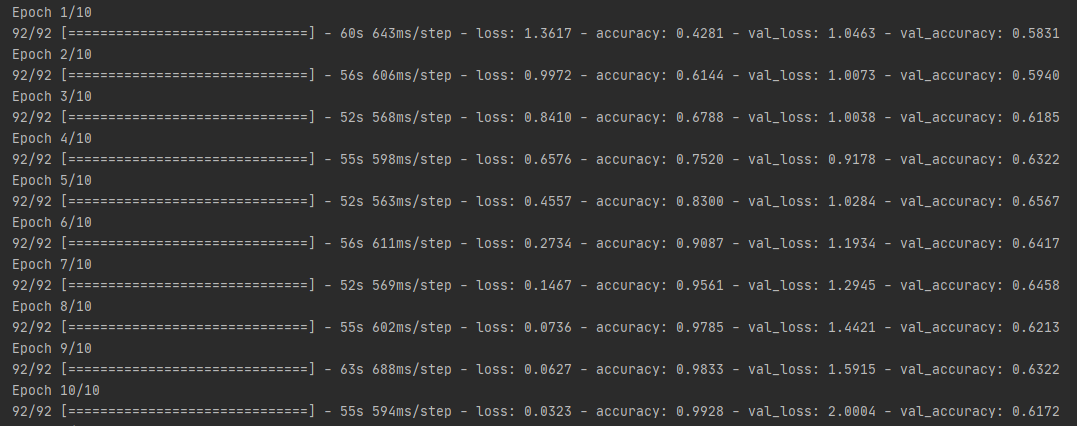
**Line 14~31, 33~40**

We define some parameters for the loader, and use 80% of the images for training and the remaining 20% ​​for validation to use the validation split. The figure below shows the first 9 images of the training dataset.

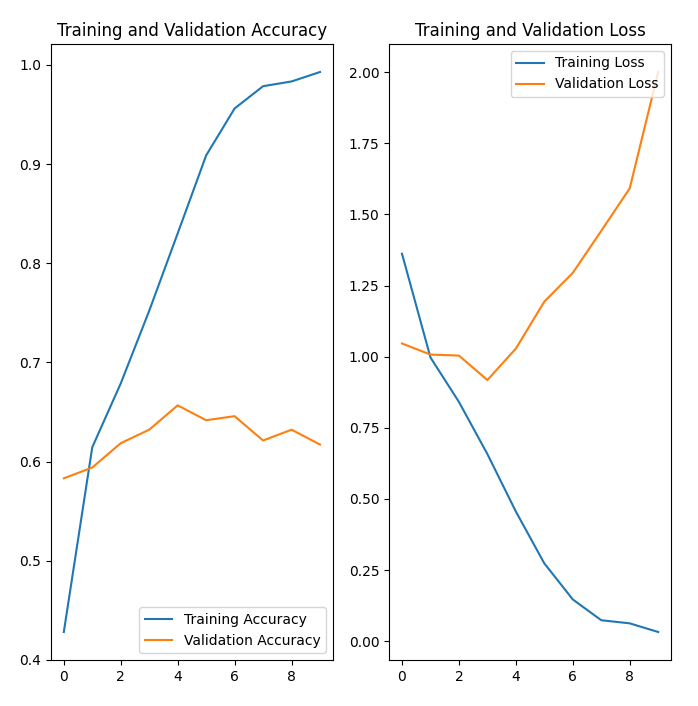


**Line 50~62, 64~72, 74~78, 80~85, 87~92**

The model was trained through 10 epochs without date augmentation.

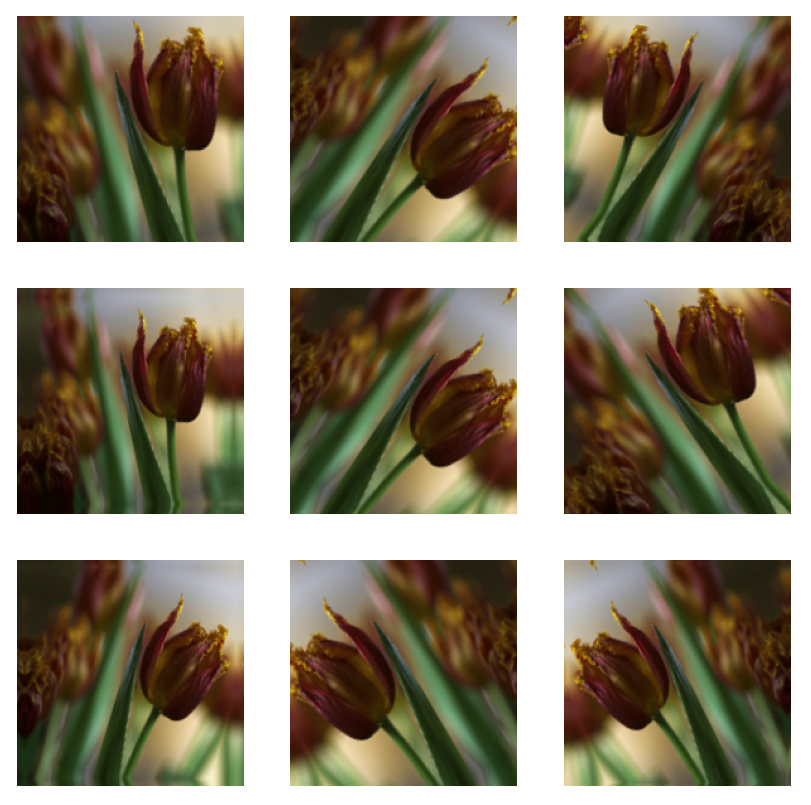


It only reaches 60% accuracy on the validation set of the model. Also, the difference in accuracy between training and validation accuracy is significant, which is a sign of overfitting.



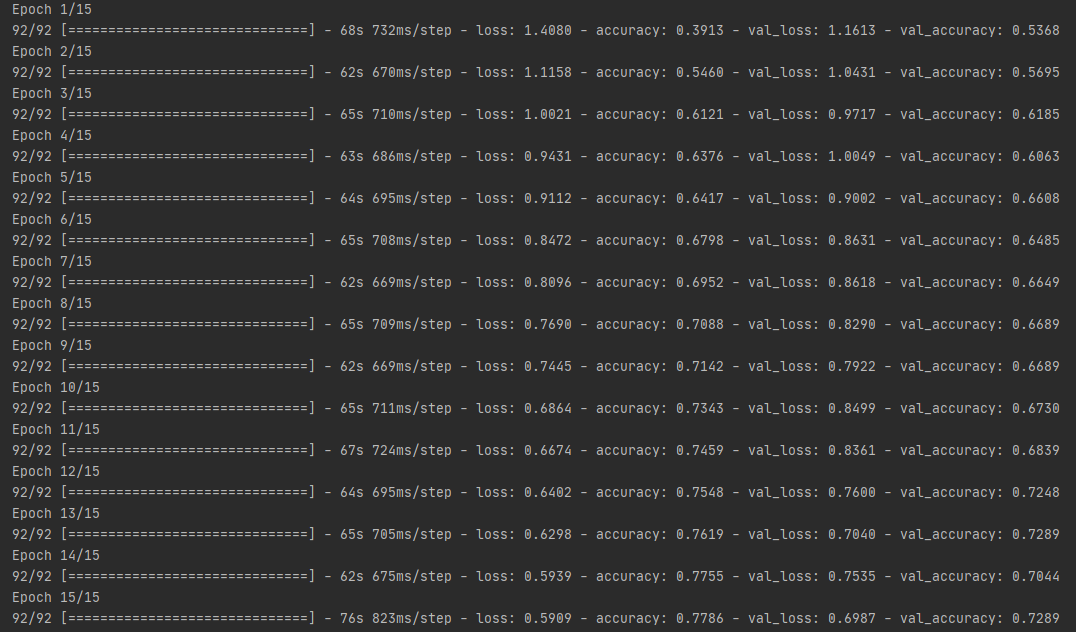
**Line 94~103, 105~111**

Therefore, data augmentation was implemented using the keras preprocessing layer. The figure below visualizes an example where data augmentation is applied multiple times.

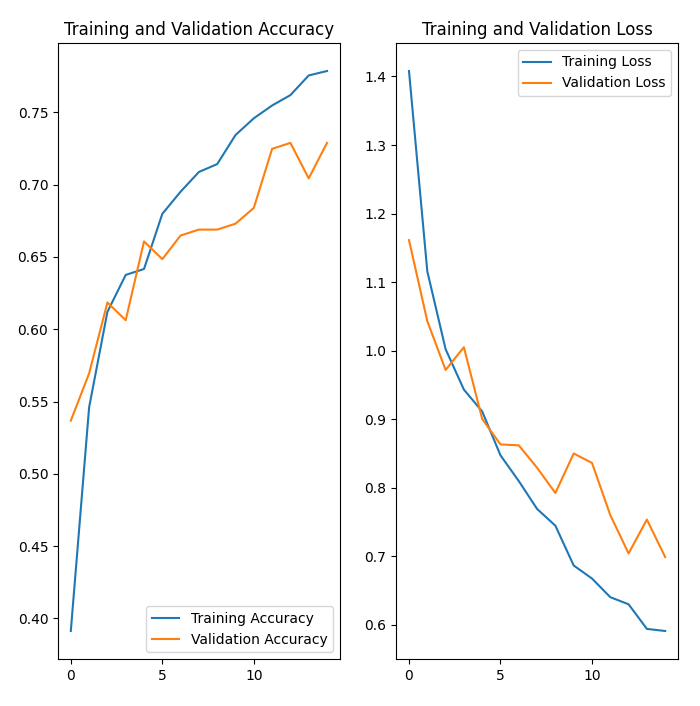


**Line 113~126, 128~136, 138~144, 146~151, 153~158**

The model was trained through 15 epochs with date augmentation.

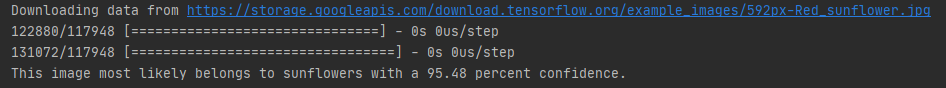


As a result of applying data augmentation, the accuracy on the validation set of the model increased to 70%, and the difference between training and validation accuracy was significantly reduced.



**Line 160~172**

A specific image was input to the model to which data augmentation was applied, and the results were derived. As shown in the figure below, results were obtained with high accuracy.



Next week’s task:

This week, we had a course to check how a model to which data augmentation is applied classifies images. Next week, we will proceed with the process of decomposing and extracting components from specific fashion images to create a model that trains them in future tasks.