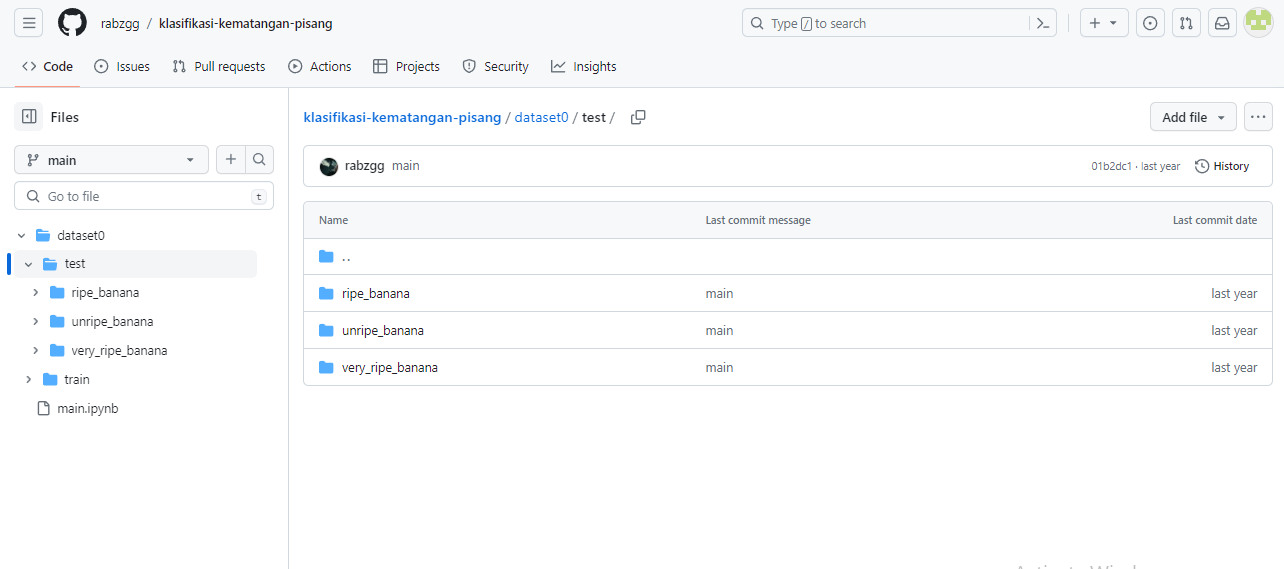
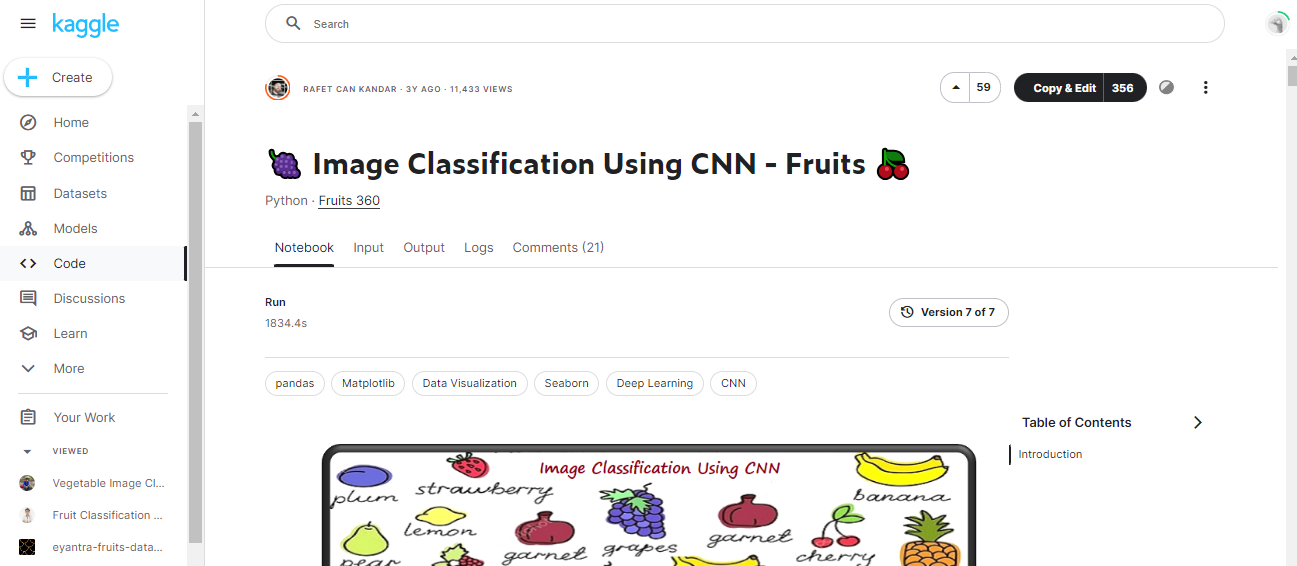
**Sumber**

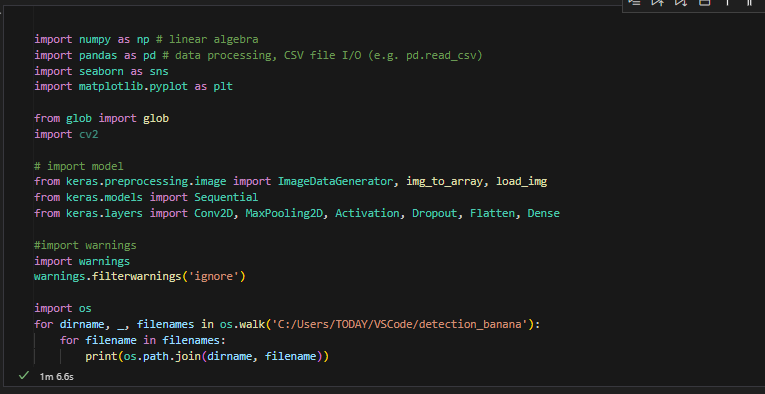
Dataset : <https://github.com/rabzgg/klasifikasi-kematangan-pisang/tree/main/dataset0/test>

****

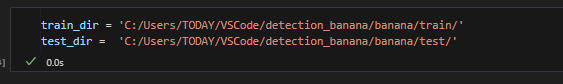
Souce Code : <https://www.kaggle.com/code/rafetcan/image-classification-using-cnn-fruits/notebook>



**Import Library**



**Data Understanding**

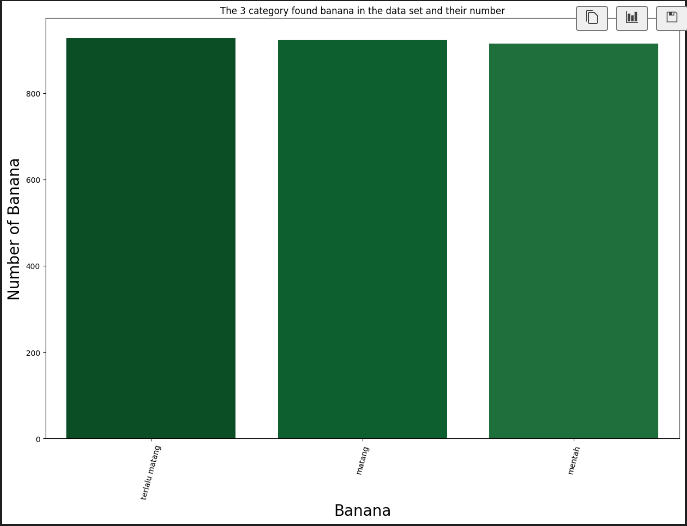


menentukan path atau direktori dari data pelatihan (**train\_dir**) dan data pengujian (**test\_dir**)

|  |  |
| --- | --- |
|  | setiap gambar pisang dan kategori pisang direpresentasikan oleh dua list terpisah: satu untuk nama kategori pisang(**banana**) dan satu untuk path gambar pisang (**banana\_image**) |

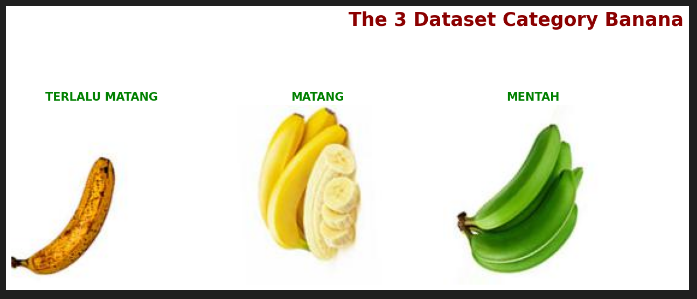
**Visualisasi**

Visualisasi Grafik

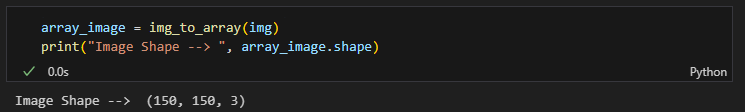


Visualisasi ini menampilkan ada 3 kategori kematangan pisang dalam dataset ini, yaitu terlalu matang, matang, dan mentah.

Visualisasi Gambar



**Data Preparation**



1. **img\_to\_array(img):**

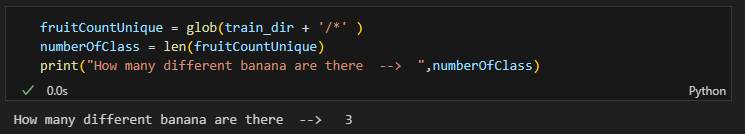
* Fungsi img\_to\_array dari Keras digunakan untuk mengonversi objek gambar (img) menjadi bentuk array numerik.
* Setiap pixel dalam gambar direpresentasikan sebagai nilai numerik dalam array.

1. **array\_image = img\_to\_array(img):**

* Hasil dari konversi gambar ke array disimpan dalam variabel array\_image.

1. **print("Image Shape --> ", array\_image.shape):**

* Baris ini mencetak bentuk (shape) dari array gambar.
* Bentuk ini menunjukkan dimensi dari array, yang mencakup tinggi, lebar, dan jumlah saluran warna (untuk gambar berwarna) atau hanya tinggi dan lebar (untuk gambar grayscale).
* Ini memberikan informasi penting tentang struktur data gambar yang akan digunakan dalam pelatihan model.



1. **fruitCountUnique = glob(train\_dir + '/\*' ):**

* Fungsi glob digunakan untuk mencocokkan semua path file atau direktori yang sesuai dengan pola tertentu.
* Dalam hal ini, train\_dir + '/\*' akan mencocokkan semua file atau direktori di dalam direktori train\_dir.
* Hasilnya akan berupa list yang berisi path dari setiap kategori buah pisang di dalam direktori pelatihan.

1. **numberOfClass = len(fruitCountUnique):**

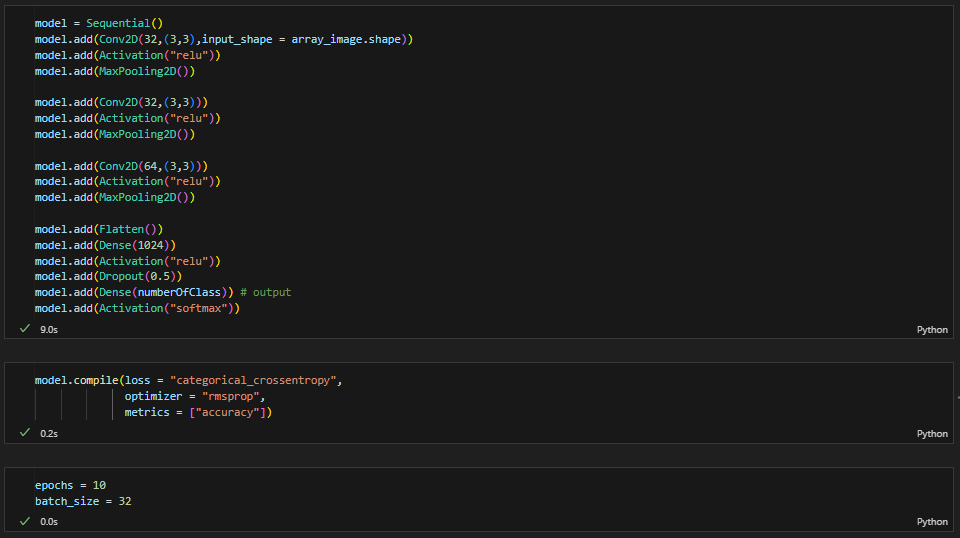
* Menggunakan fungsi len untuk menghitung berapa banyak elemen (path kategori buah pisang) yang ada dalam list fruitCountUnique.
* Jumlah ini merepresentasikan jumlah kategori buah pisang yang berbeda dalam data pelatihan.

1. **print("How many different banana are there --> ",numberOfClass):**

* Mencetak jumlah kategori buah pisang yang berbeda ke konsol.

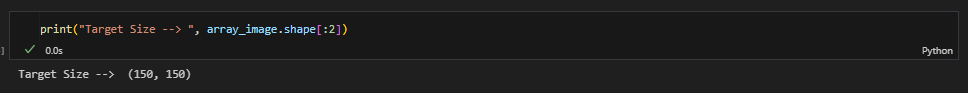
Jadi dalam data preparation ini memberikan informasi tentang dimensi atau bentuk array yang mewakili gambar yang akan digunakan dalam pelatihan model. Lalu memberikan informasi tentang berapa banyak kategori atau kelas buah pisang yang berbeda dalam data pelatihan.

**Modelling**

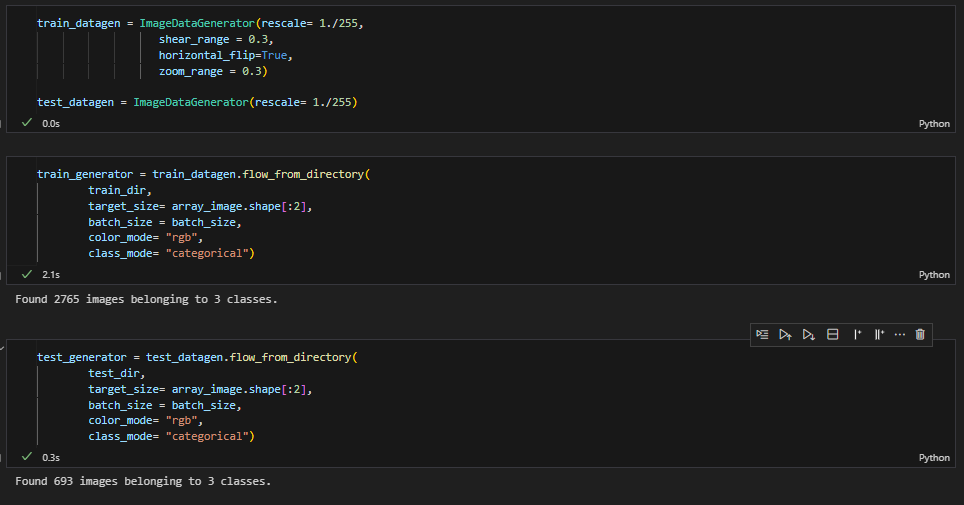
****

Model ini adalah Convolutional Neural Network (CNN) untuk klasifikasi gambar. Lalu **epochs=10** dan **batch\_size=32**, model Anda akan melalui seluruh dataset sebanyak 10 kali, dan dalam setiap iterasi (epoch), akan menggunakan batch sebanyak 32 sampel data.

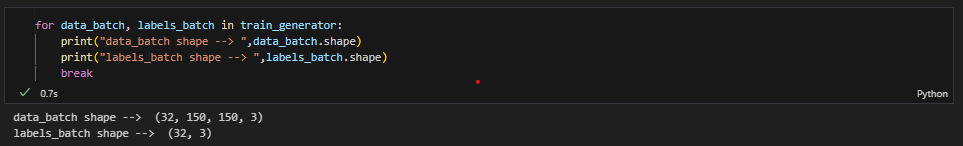
**Data Augmentation**

****

Menentukan ukuran dari gambar yang akan digunakan sebagai target pada generator data selama pelatihan model. Dimensi disini untuk mengatur parameter **target\_size** pada generator data, sehingga gambar dapat diubah ukurannya sesuai dengan yang diharapkan oleh model. Target size pada data ini yaitu (150,150).



* melakukan normalisasi
* **train\_generator**, **test\_generator** akan menghasilkan batch-batch gambar dan labelnya pada setiap iterasi pengujian. Generator ini kemudian dapat digunakan untuk mengevaluasi performa model.
* **Jumlah Data Training 2765**
* **Jumlah Data Testing 693**

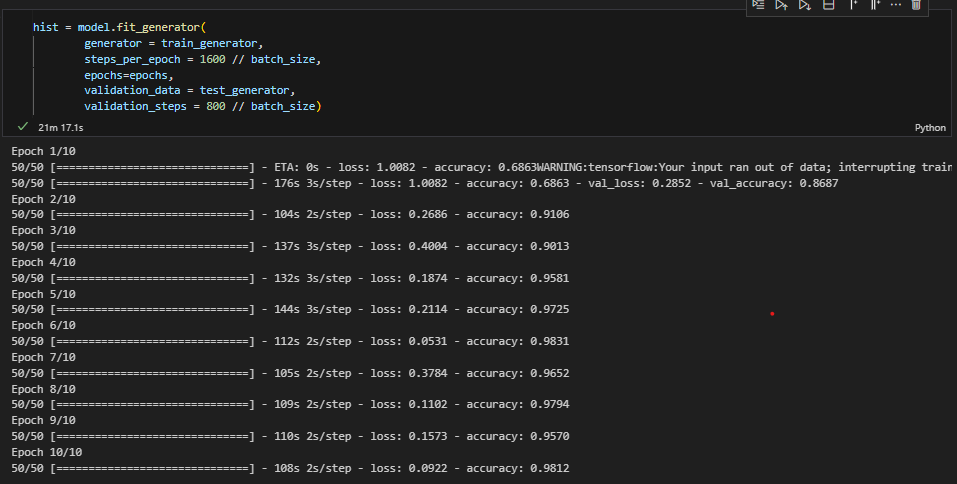


bentuk batch gambar dan label yang dihasilkan oleh generator data pada set pelatihan

**Evaluation**

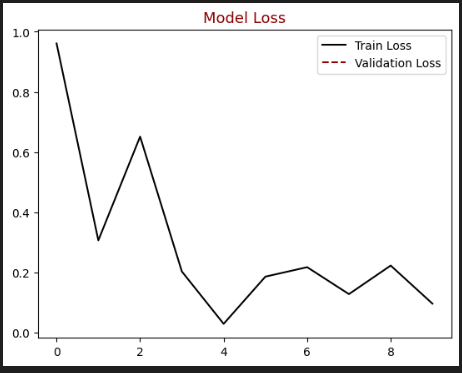
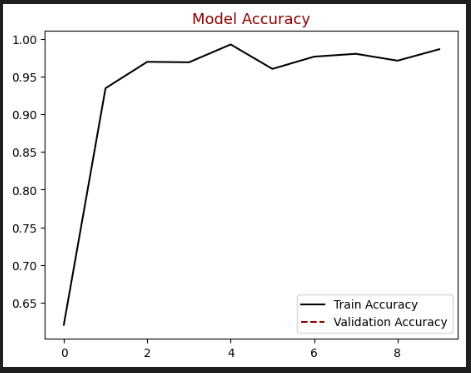
Fit Model

Fungsinya untuk melatih (train) model pada data pelatihan. Fungsi utama dari **fit** adalah mengoptimalkan parameter-model agar sesuai dengan data pelatihan yang diberikan.



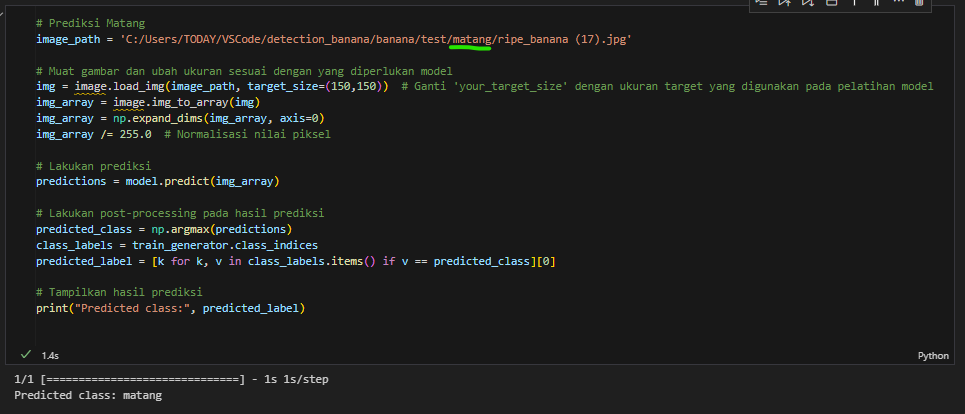
Akurasi yang didapatkan dari model CNN adalah sebesar 98%.

Visualisasi model loss dan acurracy yang didapatkan dari fit model :

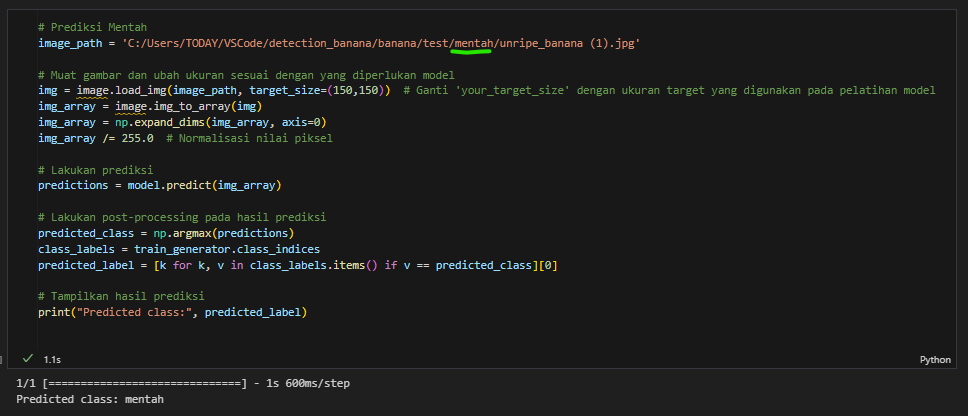
 

**Prediksi**

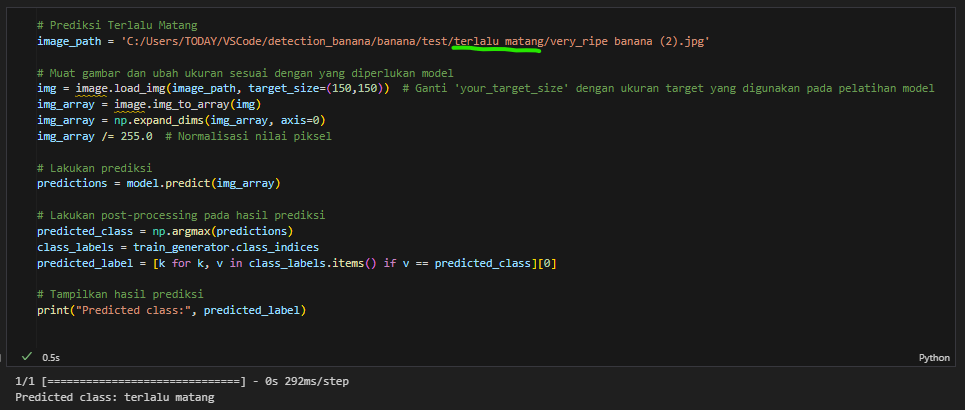
Matang

****

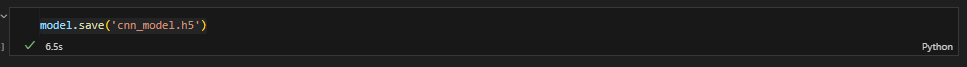
Mentah



Terlalu Matang

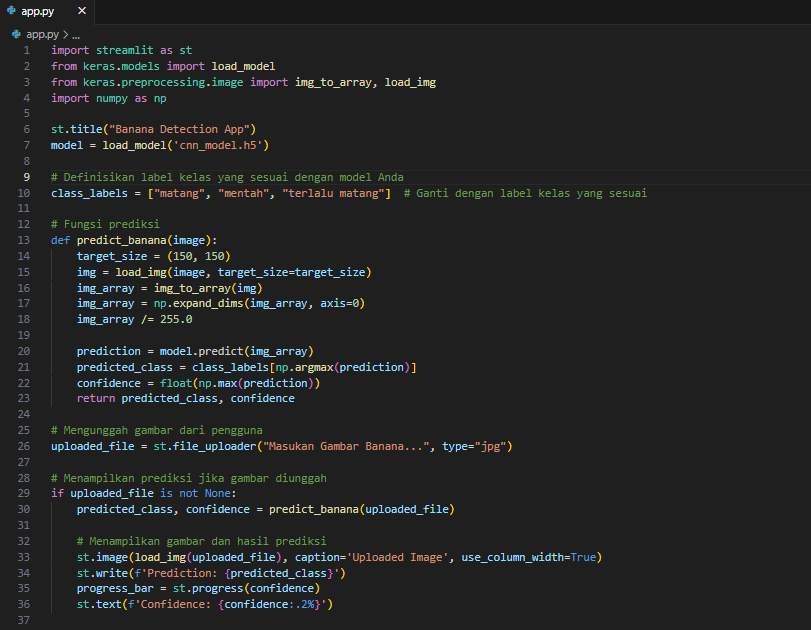


**Save Model**



Fungsinya digunakan untuk menyimpan arsitektur model, konfigurasi pelatihan (termasuk fungsi loss, optimizer, dan metrik), serta parameter-model yang telah di-optimalkan setelah pelatihan. Dengan menyimpan model, kita dapat menggunakannya kembali tanpa harus melatih ulang dari awal.

Deployment



Pada tahap deployment, Streamlite adalah framework Python yang dirancang khusus untuk membuat aplikasi web interaktif yang cepat dan mudah. Untuk memprediksi kematangan pisang maka kita menggunakan streaamlite dengn judul Banana Detection App, dan memasukan model yang sudah disimpan, dengan label matang, mentah, dan terlalu matang. Lalu ada confidence/akurasi yang diambil dri model. Tampilan Hasil Deployment :

