Team ID	PNT2022TMID13366
Project Name	Fertilizer Recommendation System for Disease Prediction

Importing the libraries:

from keras.models import

Sequential from keras.layers

import Dense from

keras.layers import

Convolution2D from

keras.layers import

MaxPooling2D from

keras.layers import Flatten

Initializing the model:

Keras has 2 ways to define a neural network:

- Sequential
- Function API

The Sequential class is used to define linear initializations of network layers which then, collectively, constitute a model.

We will use the Sequential constructor to create a model, which will then have layers added to it using the add () method. Now, will initialize our model.
Initialize the neural network layer by creating a reference/object to the Sequential class.

model=Sequential()

ADD CNN Layers

We will be adding three layers for CNN

- Convolution layer
- Pooling layer
- Flattening layer

model.add(Conv2D(32, (3, 3), padding="same",input_shape=inputShape))

model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))
model.add(MaxPooling2D(pool_size=(3, 3)))
model.add(Dropout(0.25))

```python
model.add(Conv2D(64, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(BatchNormalization(axis=chanDim))
model.add(Conv2D(64, (3, 3), padding="same"))
model.add(Activation("relu"))
model.add(MaxPooling2D(pool_size  =  (2,2)))

model.add(Flatten())
```

Model.Summary()

Can be used to see all parameters and shapes in each layer in our models.

```python
 model.summary()
```

Fit and save the model

```python
history = model.fit_generator(
aug.flow(x_train, y_train, batch_size=BS),
validation_data=(x_test, y_test),
steps_per_epoch=len(x_train) // BS,
epochs=EPOCHS, verbose=1)

model.save("my_model.h5")
```