```
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

```
(x_train, y_train), (x_test, y_test) = tf.keras.datasets.cifar10.load_data()
```

```
x_train, x_test = x_train / 255.0, x_test / 255.0
```

```
val_datagen = ImageDataGenerator(rescale=1./255)
```

```
y_train = tf.keras.utils.to_categorical(y_train, 10)
y_test = tf.keras.utils.to_categorical(y_test, 10)
```

```
model = Sequential([
    Conv2D(32, (3, 3), activation='relu', input_shape=(32, 32, 3)),
    MaxPooling2D((2, 2)),
    Conv2D(64, (3, 3), activation='relu'),
    MaxPooling2D((2, 2)),
    Conv2D(128, (3, 3), activation='relu'),
    Flatten(),
    Dense(512, activation='relu'),
    Dense(10, activation='softmax')  # 10 classes for CIFAR-10
])
```

```
C:\Users\Parth\AppData\Local\Programs\Python\Python312\Lib\site-packages\keras\src\layers\convolutional\base_conv.py:107: UserW
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

```
model.compile(
    loss='categorical_crossentropy',
    optimizer=Adam(),
    metrics=['accuracy']
)

history = model.fit(
    x_train, y_train,
    epochs=10,
    validation_data=(x_test, y_test)
)
```

```
Epoch 1/10
1563/1563 ━━━━━━━━━━━━━━━━━━━━ 83s 34ms/step - accuracy: 0.3818 - loss: 1.6728 - val_accuracy: 0.5946 - val_loss: 1.1401
Epoch 2/10
1563/1563 ━━━━━━━━━━━━━━━━━━━━ 58s 37ms/step - accuracy: 0.6147 - loss: 1.0849 - val_accuracy: 0.6488 - val_loss: 0.9999
Epoch 3/10
1563/1563 ━━━━━━━━━━━━━━━━━━━━ 63s 40ms/step - accuracy: 0.6955 - loss: 0.8668 - val_accuracy: 0.6784 - val_loss: 0.9332
Epoch 4/10
1563/1563 ━━━━━━━━━━━━━━━━━━━━ 55s 35ms/step - accuracy: 0.7468 - loss: 0.7218 - val_accuracy: 0.7109 - val_loss: 0.8495
Epoch 5/10
1563/1563 ━━━━━━━━━━━━━━━━━━━━ 65s 41ms/step - accuracy: 0.7922 - loss: 0.5893 - val_accuracy: 0.7076 - val_loss: 0.8558
Epoch 6/10
1563/1563 ━━━━━━━━━━━━━━━━━━━━ 56s 36ms/step - accuracy: 0.8353 - loss: 0.4700 - val_accuracy: 0.7099 - val_loss: 0.9106
Epoch 7/10
  81/1563 ━━━━━━━━━━━━━━━━━━━━ 50s 34ms/step - accuracy: 0.8659 - loss: 0.3815
```

```
loss, accuracy = model.evaluate(x_test, y_test)
print(f"Test loss: {loss:.4f}")
print(f"Test accuracy: {accuracy:.4f}")
```

```
Start coding or generate with AI.
```