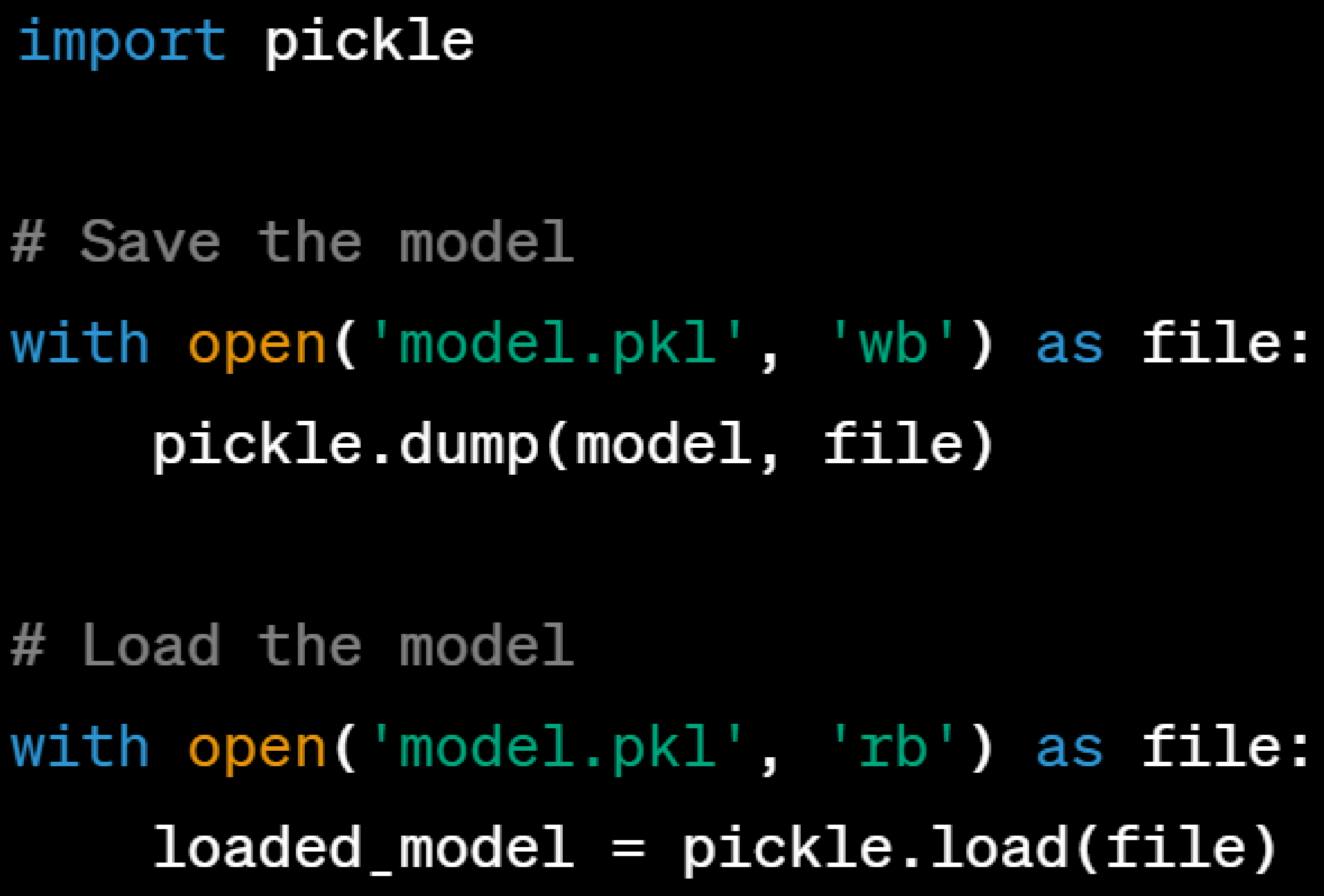
1. Pickle

The easiest way to save and load Python objects, including machine learning models



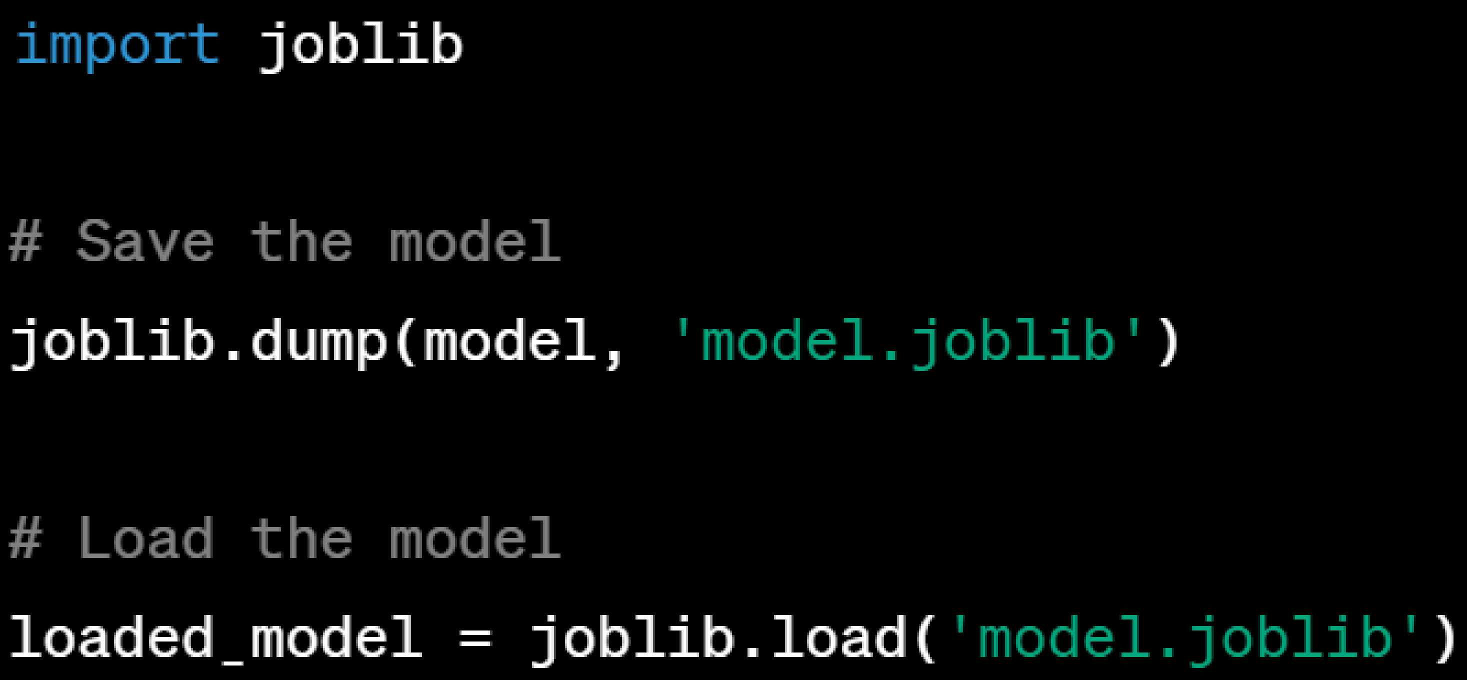
```python
import pickle

# Save the model
with open('model.pkl', 'wb') as file:
    pickle.dump(model, file)

# Load the model
with open('model.pkl', 'rb') as file:
    loaded_model = pickle.load(file)
```

Follow: in/Raghav Vashisht

2. Joblib

Joblib is a more efficient alternative to pickle for saving and loading large NumPy arrays and objects.

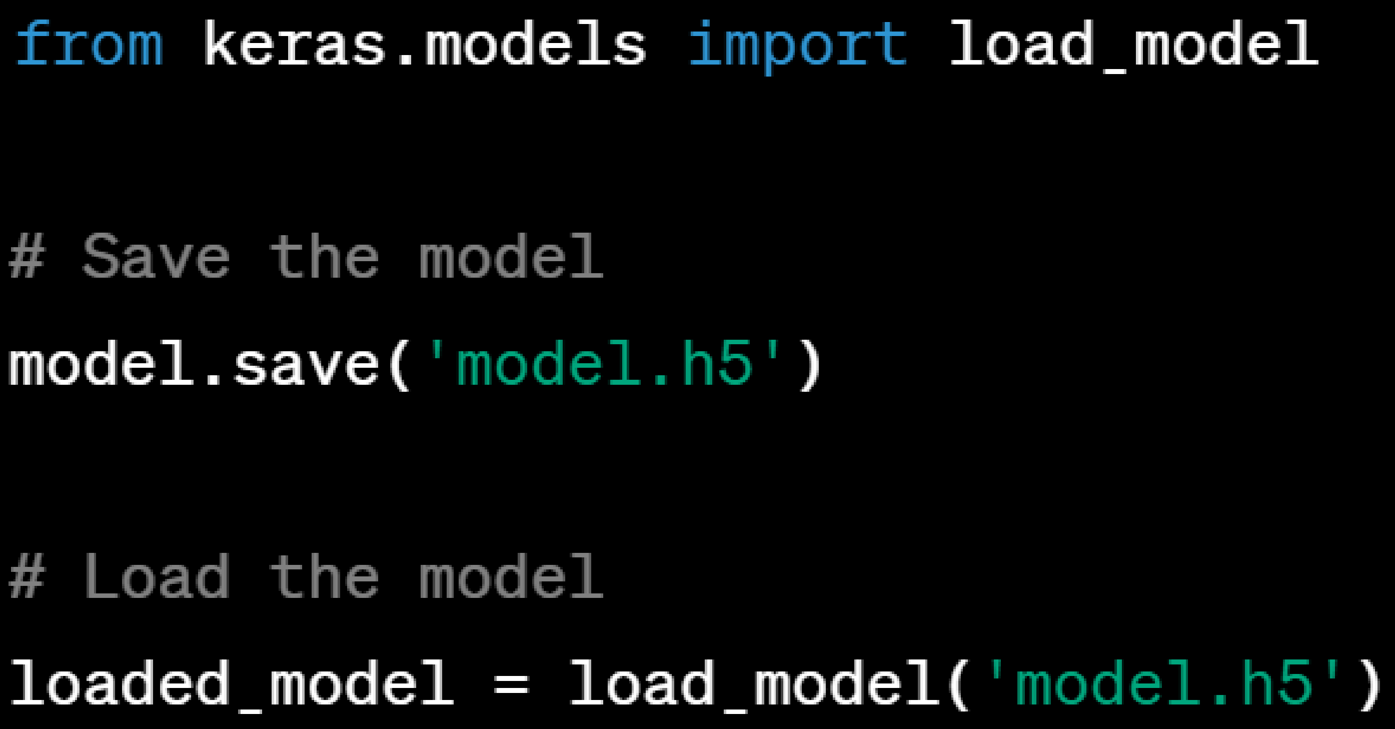
It's often preferred for scikit-learn models.



```
import joblib

# Save the model
joblib.dump(model, 'model.joblib')

# Load the model
loaded_model = joblib.load('model.joblib')
```

3. HDF5

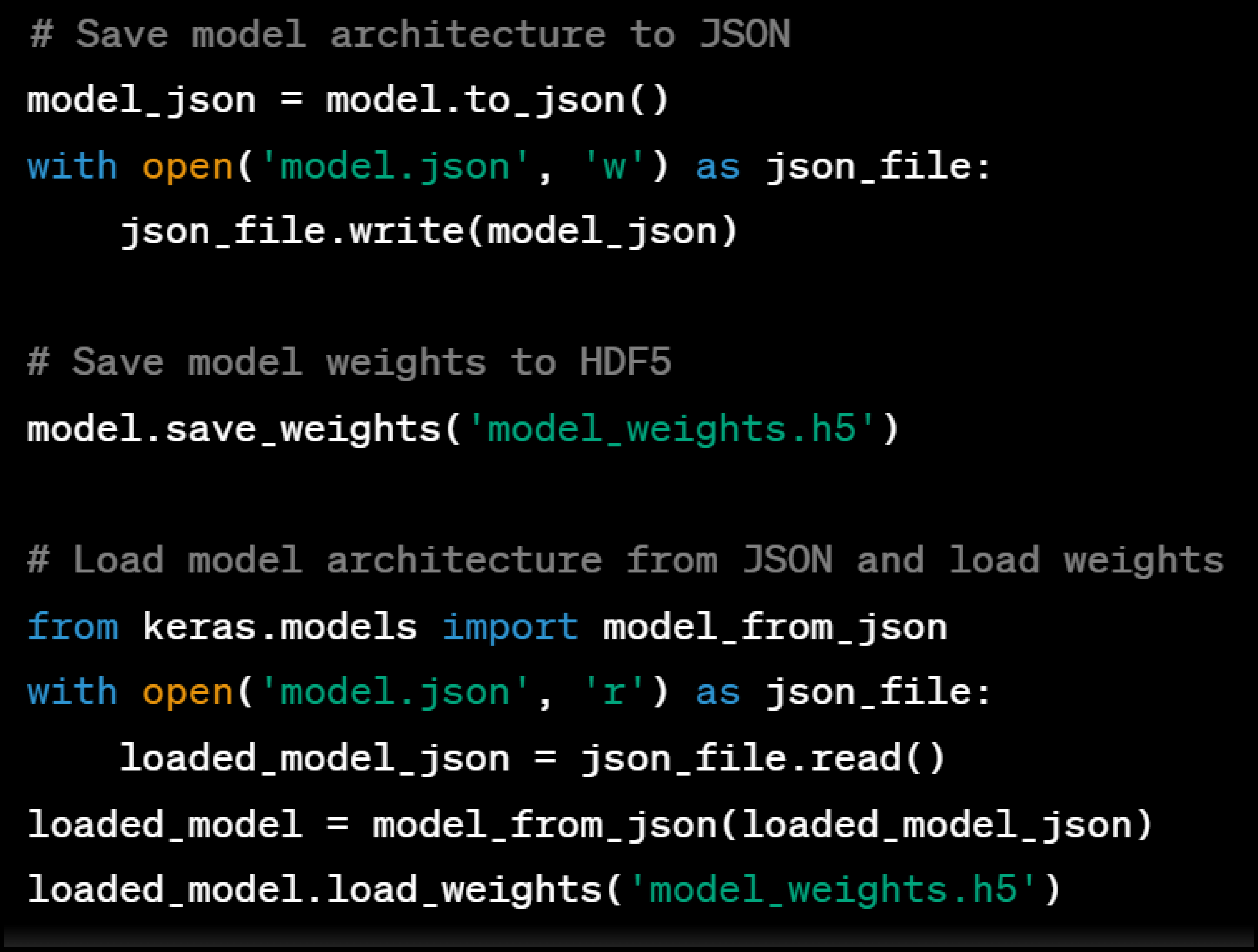
If you're working with deep learning models created with libraries like TensorFlow or Keras, it's common to save them in the HDF5 format.



```python
from keras.models import load_model

# Save the model
model.save('model.h5')

# Load the model
loaded_model = load_model('model.h5')
```

4. JSON or YAML

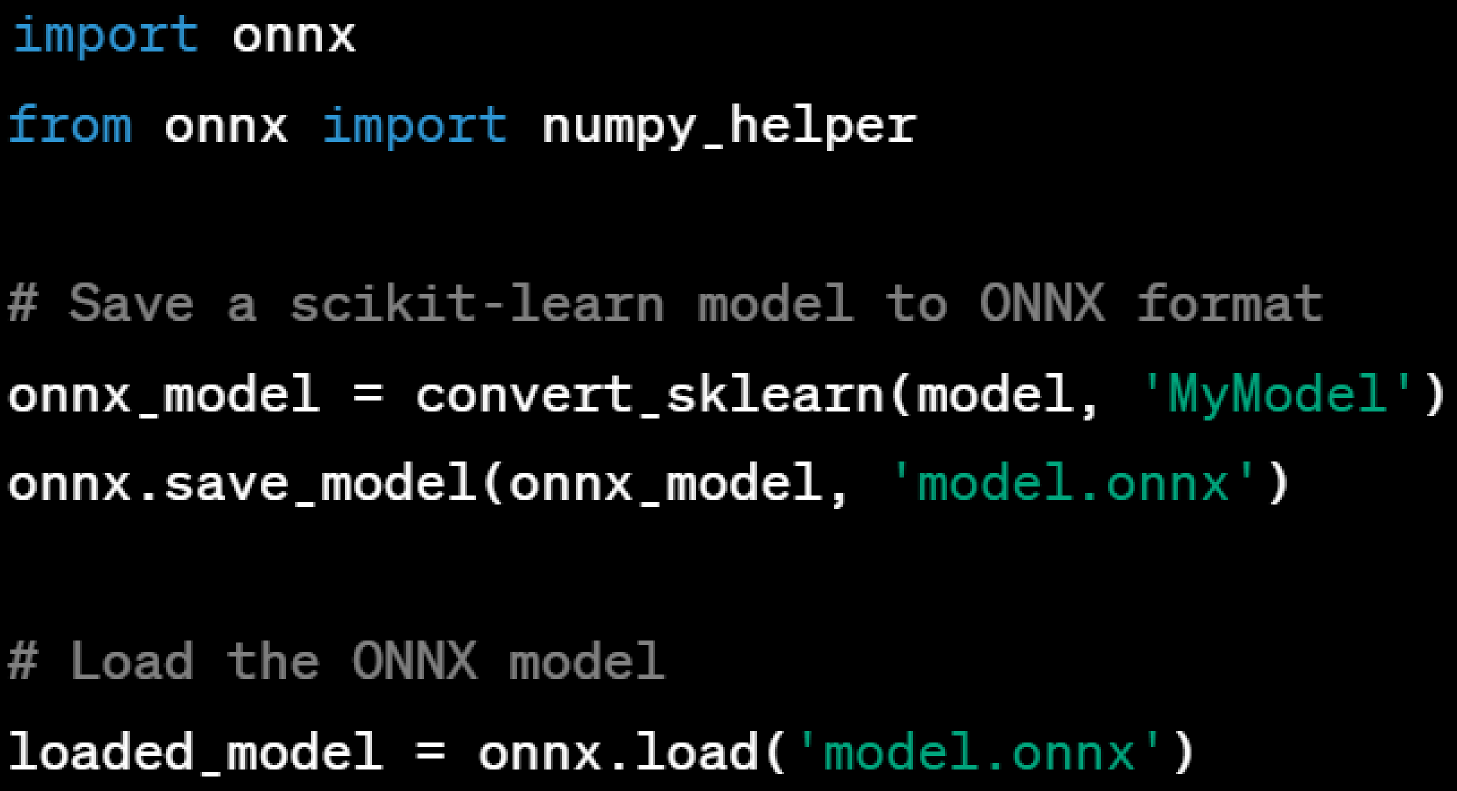
You can save model architecture and weights separately in JSON or YAML formats for better readability and sharing model details.



```
# Save model architecture to JSON
model_json = model.to_json()
with open('model.json', 'w') as json_file:
    json_file.write(model_json)

# Save model weights to HDF5
model.save_weights('model_weights.h5')

# Load model architecture from JSON and load weights
from keras.models import model_from_json
with open('model.json', 'r') as json_file:
    loaded_model_json = json_file.read()
loaded_model = model_from_json(loaded_model_json)
loaded_model.load_weights('model_weights.h5')
```

5. ONNX (Open Neural Network Exchange):

ONNX is an open format for representing machine learning models. You can convert models from various frameworks to ONNX and save them for interoperability.



```
import onnx
from onnx import numpy_helper

# Save a scikit-learn model to ONNX format
onnx_model = convert_sklearn(model, 'MyModel')
onnx.save_model(onnx_model, 'model.onnx')

# Load the ONNX model
loaded_model = onnx.load('model.onnx')
```

Follow: in/Raghav Vashisht

Remember Version Control:

Always consider using version control systems (e.g., Git) to track changes to your code and models. You can store model files in a repository for collaboration and tracking model versions.