Google ML Kit

Digital ink recognition

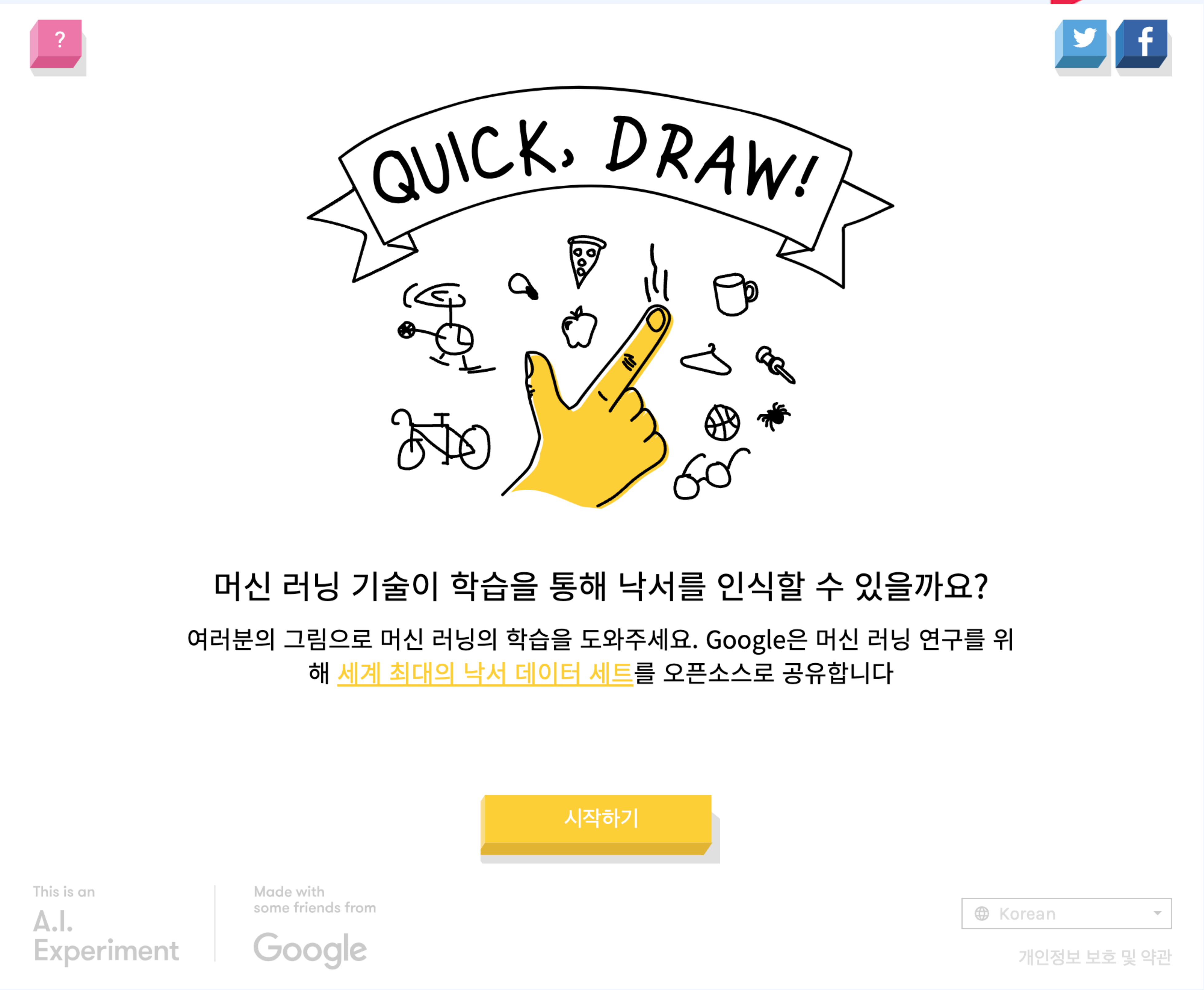
Google ML Kit

ML Kit - Digital ink recognition

ML Kit - Digital ink recognition 이론

With ML Kit's digital ink recognition API, you can recognize handwritten text and classify gestures on a digital surface in hundreds of languages, as well as classify sketches.

The digital ink recognition API uses the same technology that powers handwriting recognition in Gboard, Google Translate, and the Quick, Draw! game.



머신 러닝 기술이 학습을 통해 낙서를 인식할 수 있을까요?

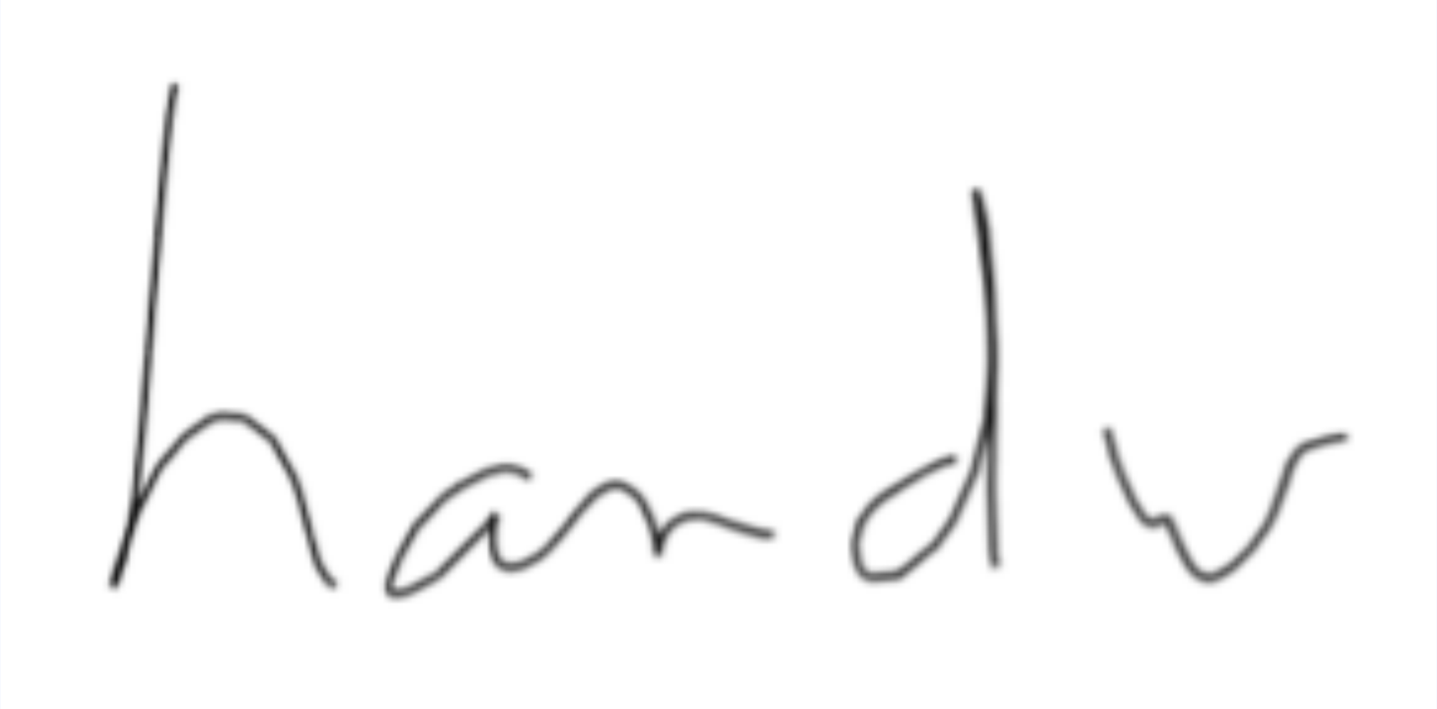
여러분의 그림으로 머신 러닝의 학습을 도와주세요. Google은 머신 러닝 연구를 위해 세계 최대의 낙서 데이터 세트를 오픈소스로 공유합니다

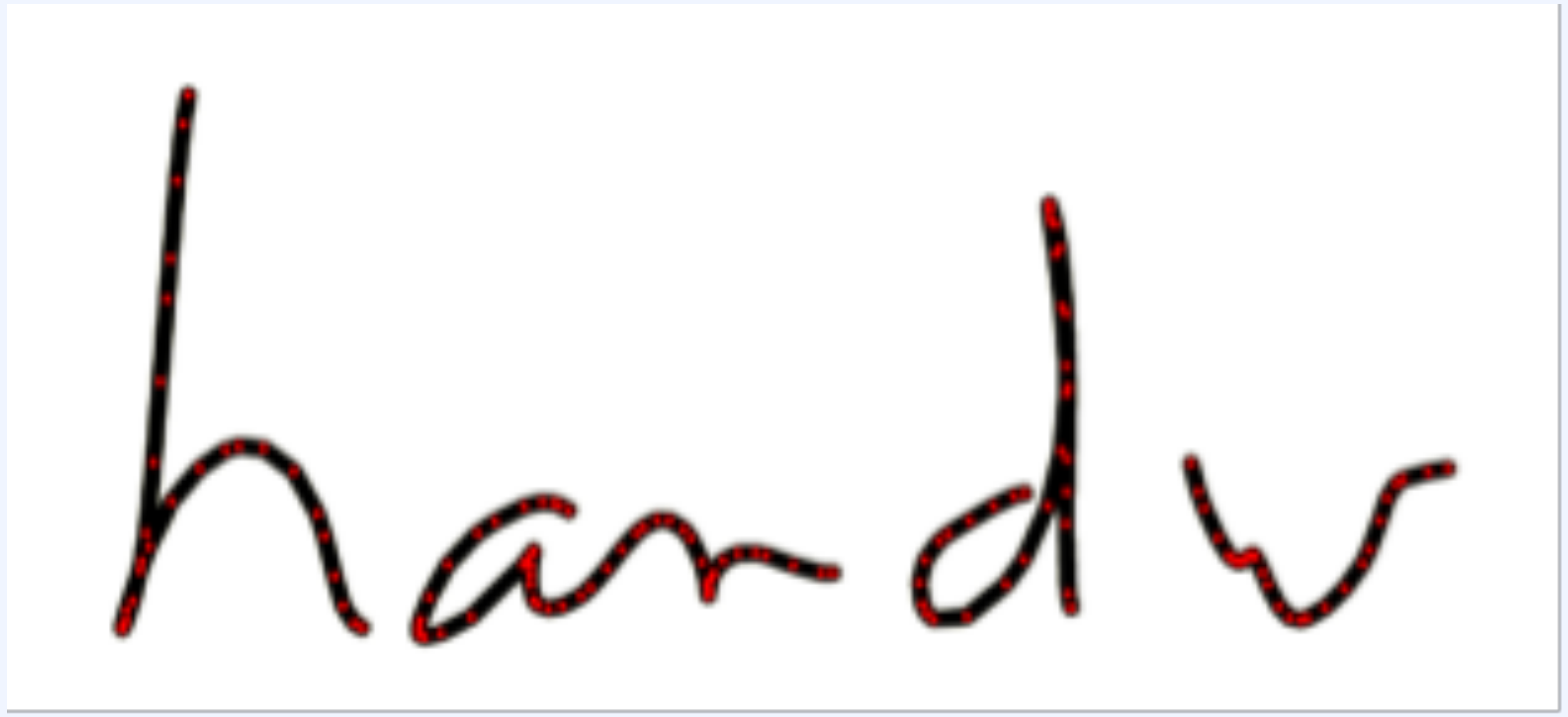
시작하기
This is an A.I. Experiment
Made with some friends from Google
Korean
개인정보 보호 및 약관

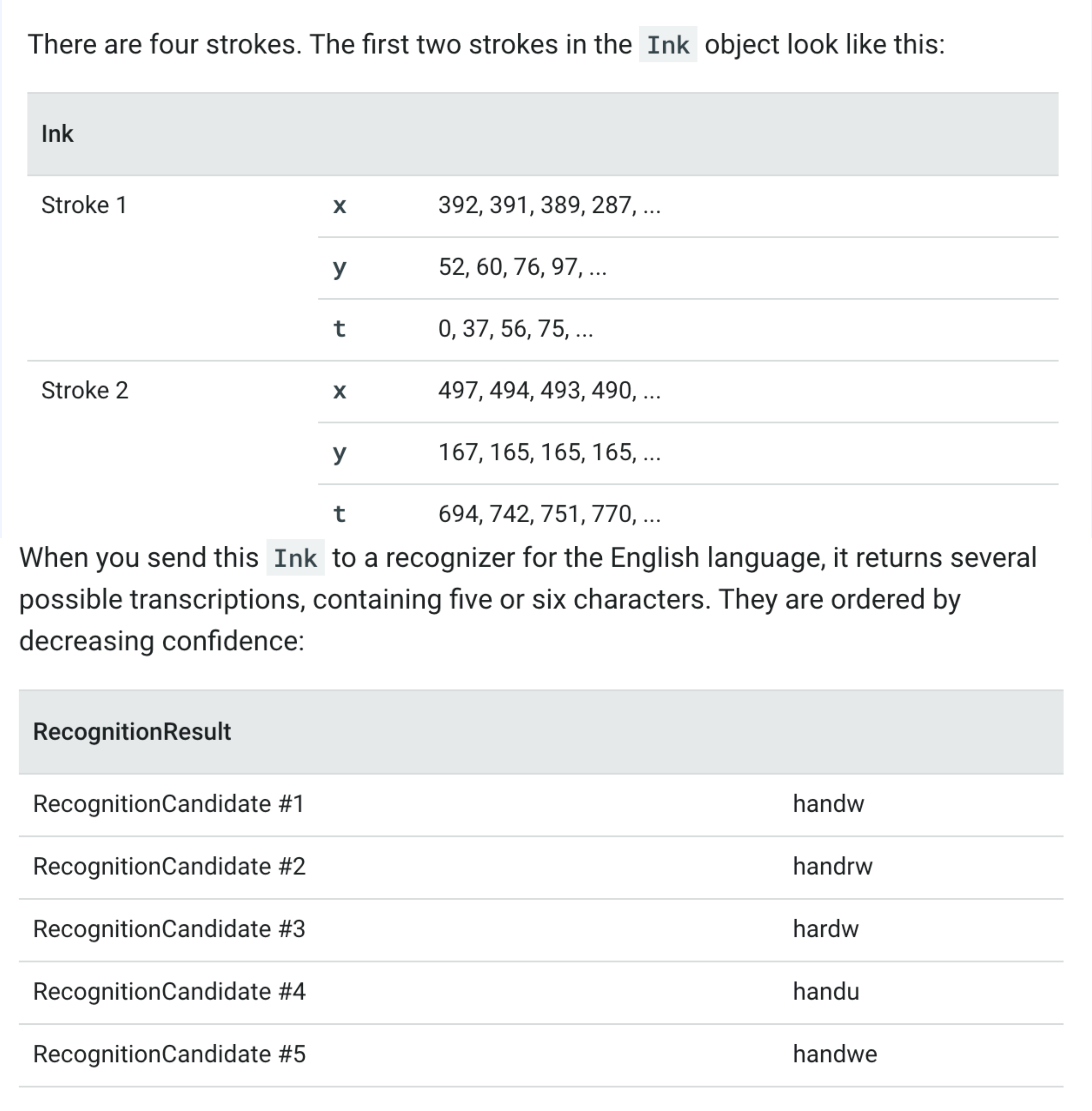
https://quickdraw.withgoogle.com/

https://developers.google.com/ml-kit/vision/digital-ink-recognition

ML Kit - Digital ink recognition 이론







There are four strokes. The first two strokes in the `Ink` object look like this:

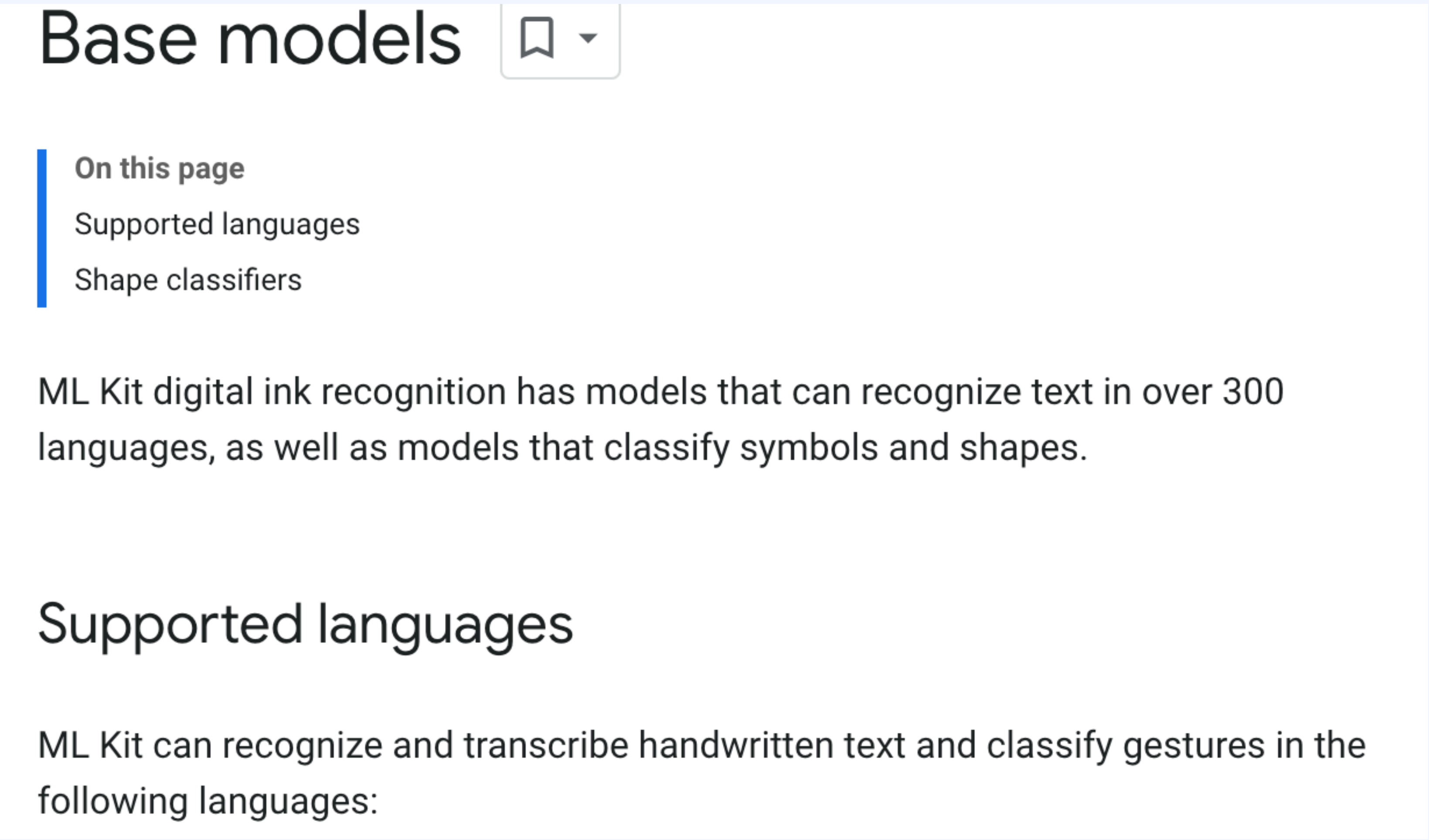
Ink		
Stroke 1	x	392, 391, 389, 287, ...
	y	52, 60, 76, 97, ...
	t	0, 37, 56, 75, ...
Stroke 2	x	497, 494, 493, 490, ...
	y	167, 165, 165, 165, ...
	t	694, 742, 751, 770, ...

When you send this `Ink` to a recognizer for the English language, it returns several possible transcriptions, containing five or six characters. They are ordered by decreasing confidence:

RecognitionResult	
RecognitionCandidate #1	handw
RecognitionCandidate #2	handrw
RecognitionCandidate #3	hardw
RecognitionCandidate #4	handu
RecognitionCandidate #5	handwe

https://developers.google.com/ml-kit/vision/pose-detection

ML Kit - Digital ink recognition 이론



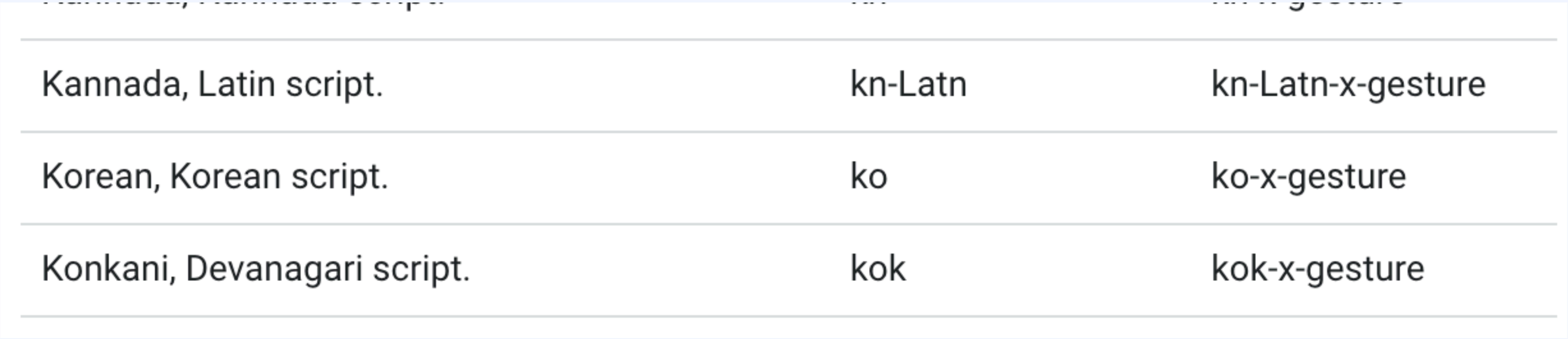
Base models

On this page
Supported languages
Shape classifiers

ML Kit digital ink recognition has models that can recognize text in over 300 languages, as well as models that classify symbols and shapes.

Supported languages

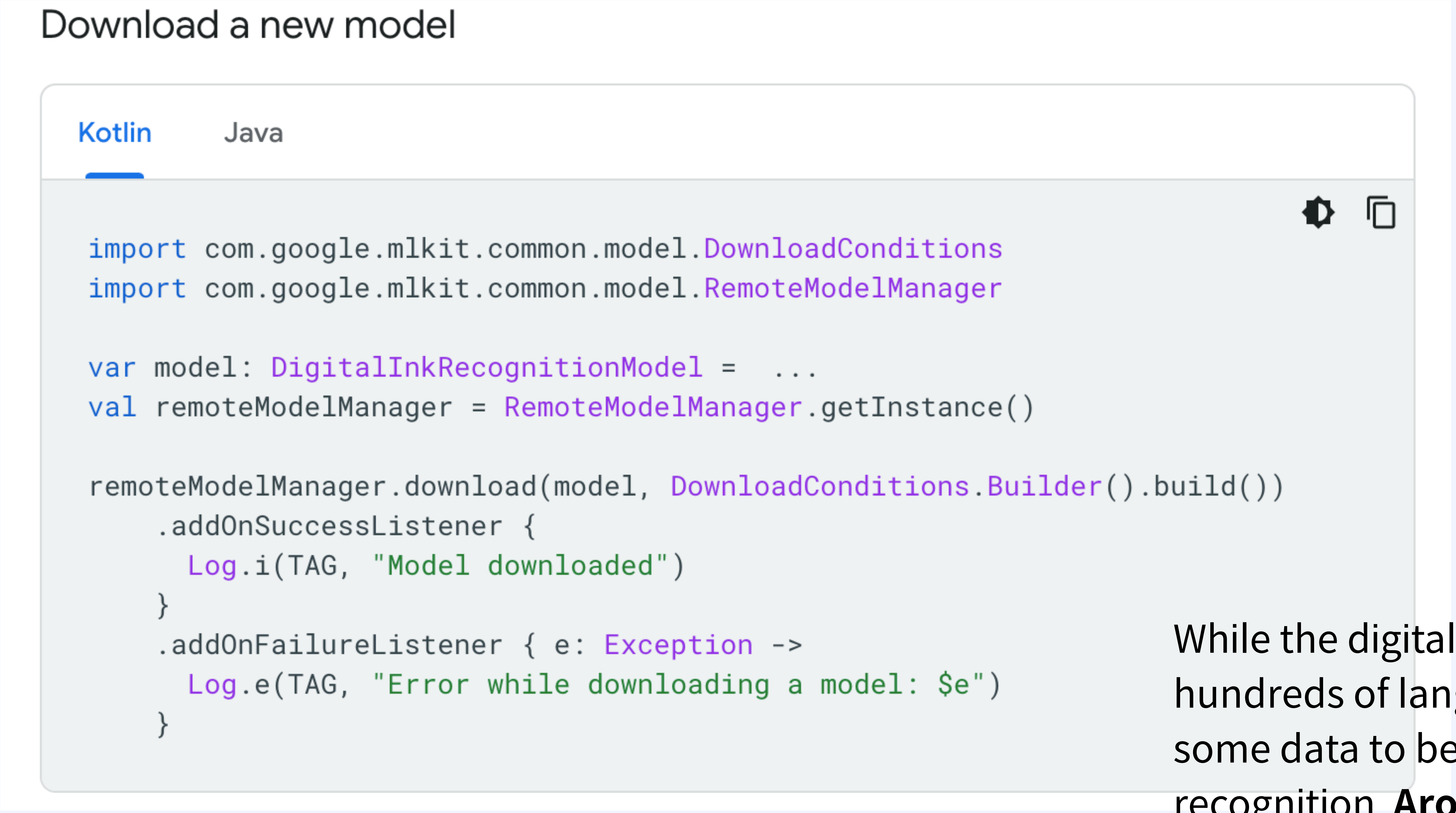
ML Kit can recognize and transcribe handwritten text and classify gestures in the following languages:



Kannada, Latin script.	kn-Latn	kn-Latn-x-gesture
Korean, Korean script.	ko	ko-x-gesture
Konkani, Devanagari script.	kok	kok-x-gesture

https://developers.google.com/ml-kit/vision/digital-ink-recognition/base-models

ML Kit - Digital ink recognition 이론



Download a new model

```kotlin
import com.google.mlkit.common.model.DownloadConditions
import com.google.mlkit.common.model.RemoteModelManager

var model: DigitalInkRecognitionModel =  ...
val remoteModelManager = RemoteModelManager.getInstance()

remoteModelManager.download(model, DownloadConditions.Builder().build())
    .addOnSuccessListener {
      Log.i(TAG, "Model downloaded")
    }
    .addOnFailureListener { e: Exception ->
      Log.e(TAG, "Error while downloading a model: $e")
    }
```

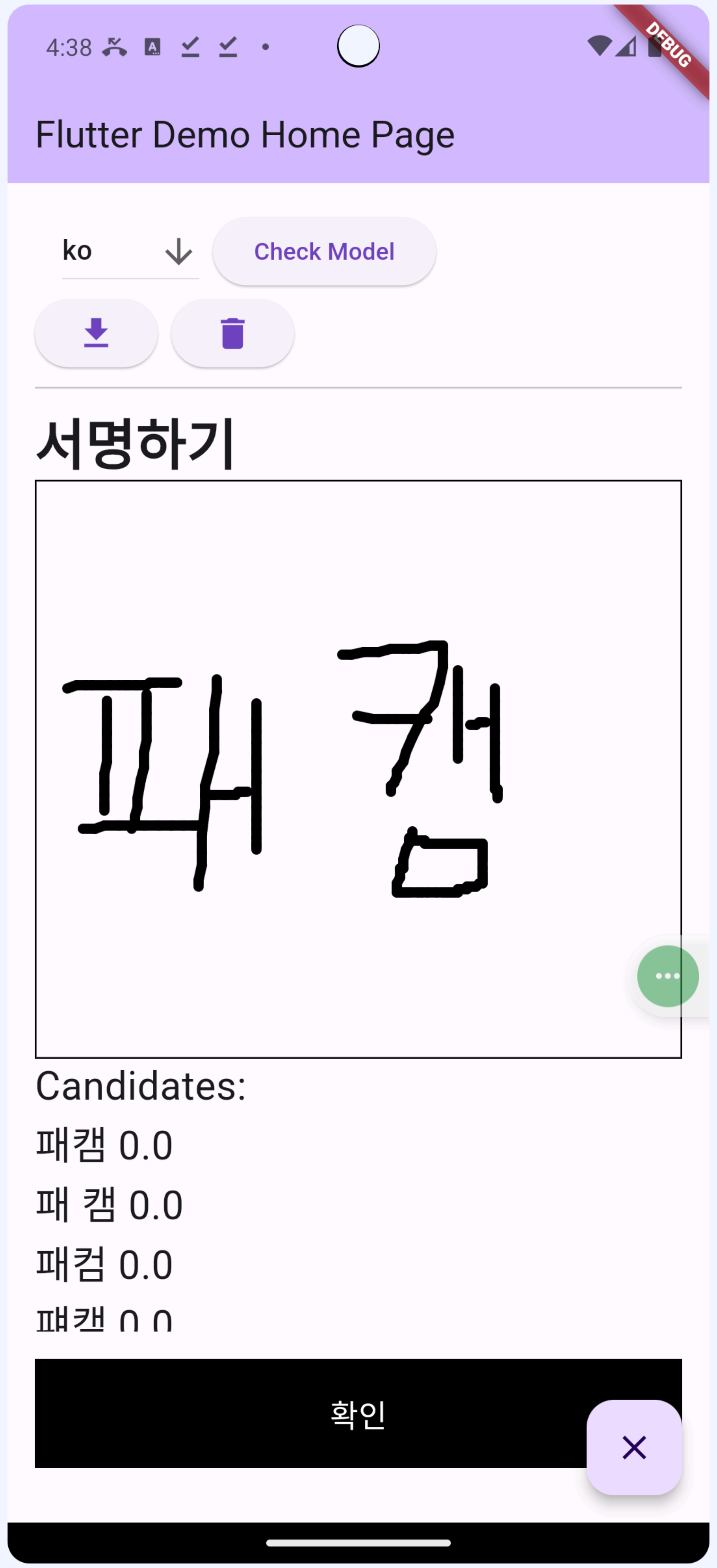
While the digital ink recognition API supports hundreds of languages, each language requires some data to be downloaded prior to any recognition. **Around 20MB of storage is required per language.**

https://developers.google.com/ml-kit/vision/digital-ink-recognition/android#kotlin

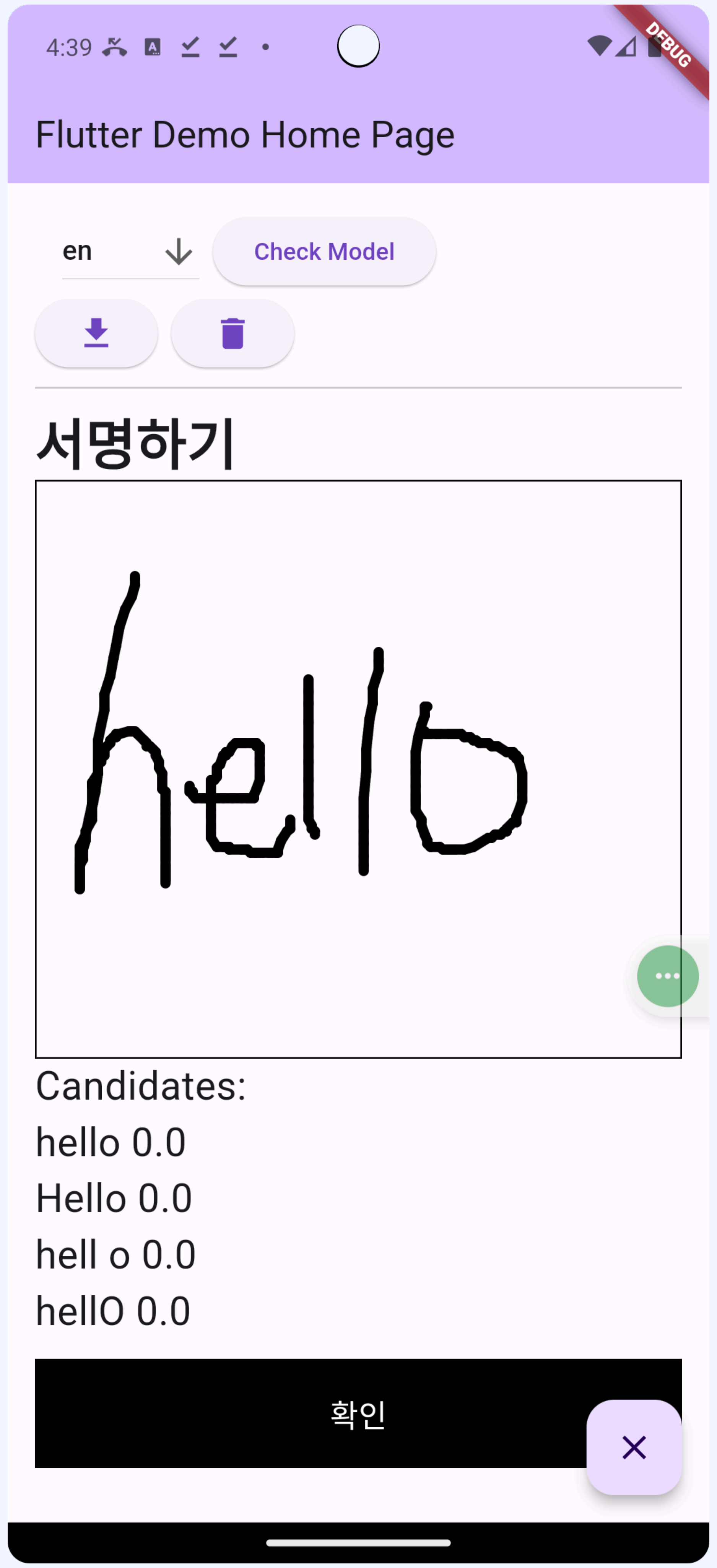
Google ML Kit

ML Kit - Digital ink recognition
화면디자인 리뷰

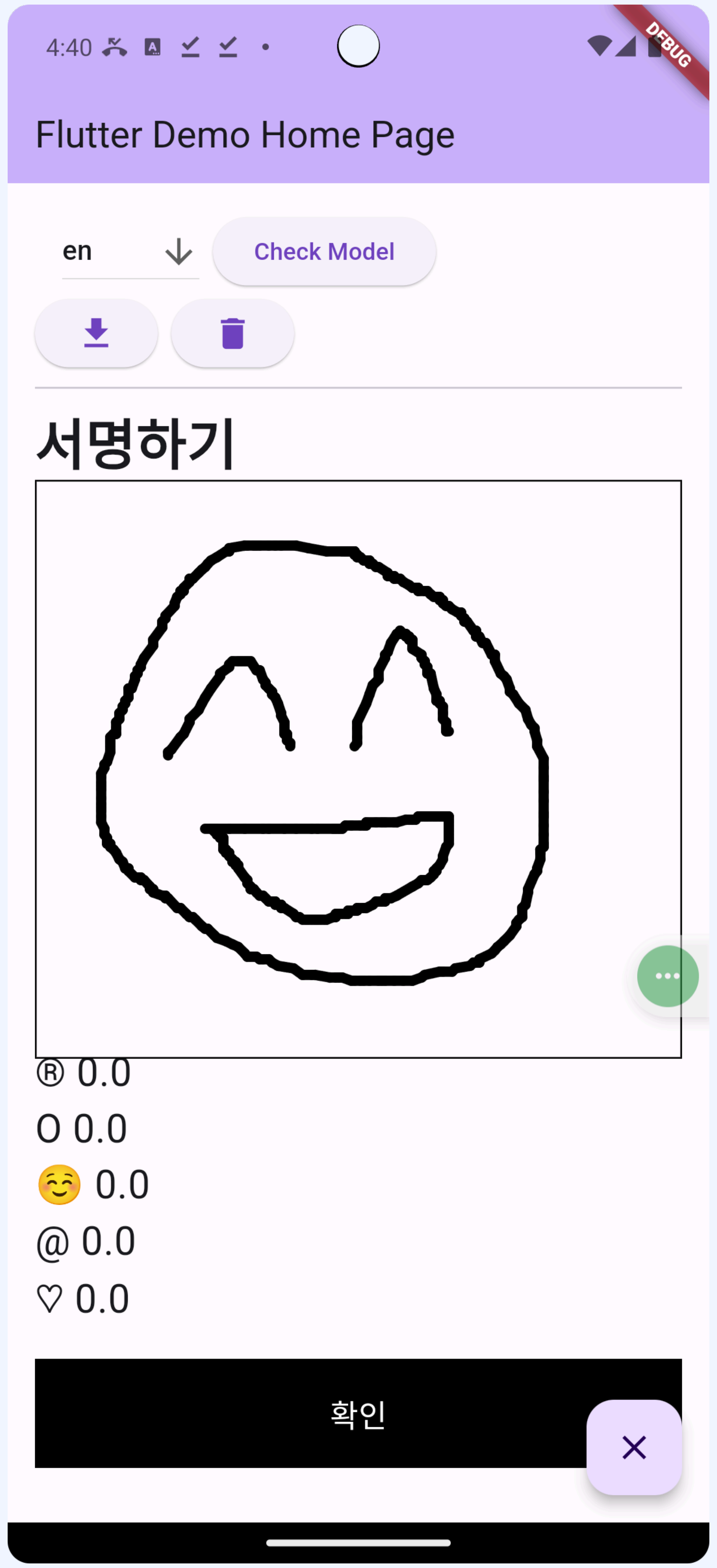
ML Kit - Digital ink recognition



Candidates:
패캠 0.0
패 캠 0.0
패컴 0.0
패캔 0.0

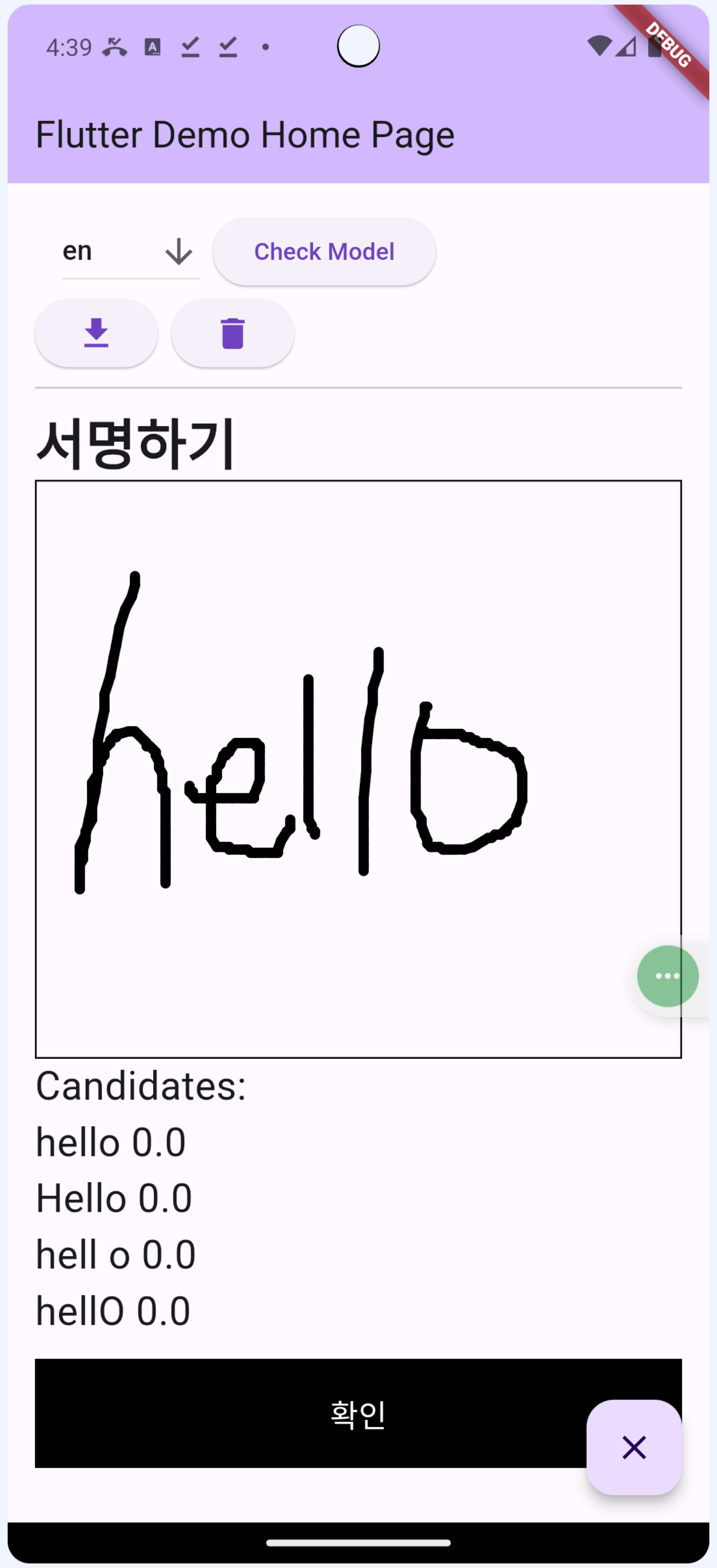


Candidates:
hello 0.0
Hello 0.0
hell o 0.0
hellO 0.0

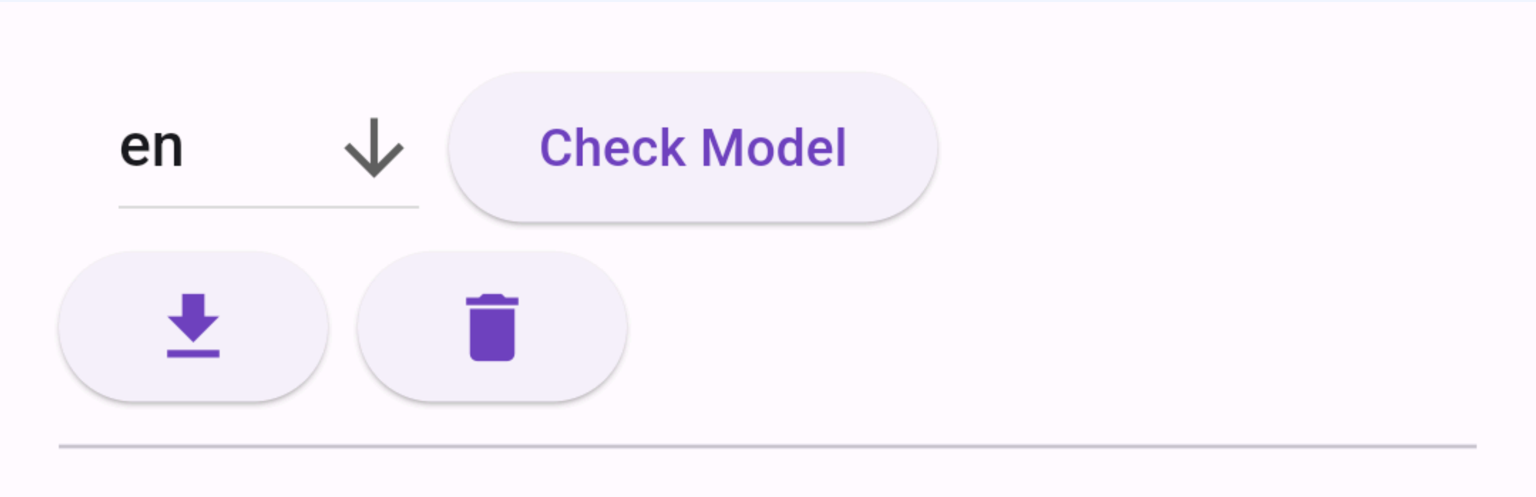


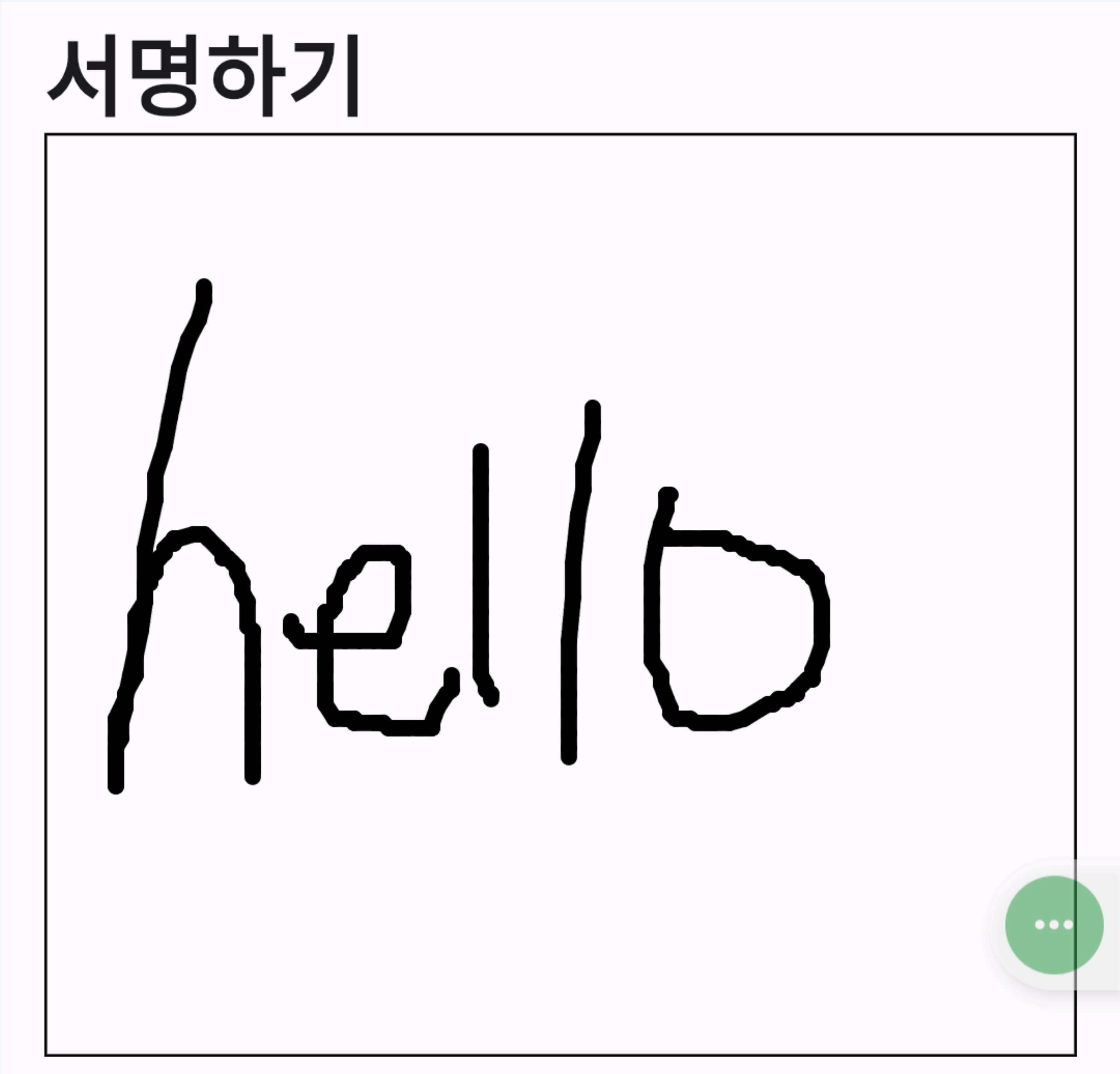
® 0.0
O 0.0
☺️ 0.0
@ 0.0
♡ 0.0

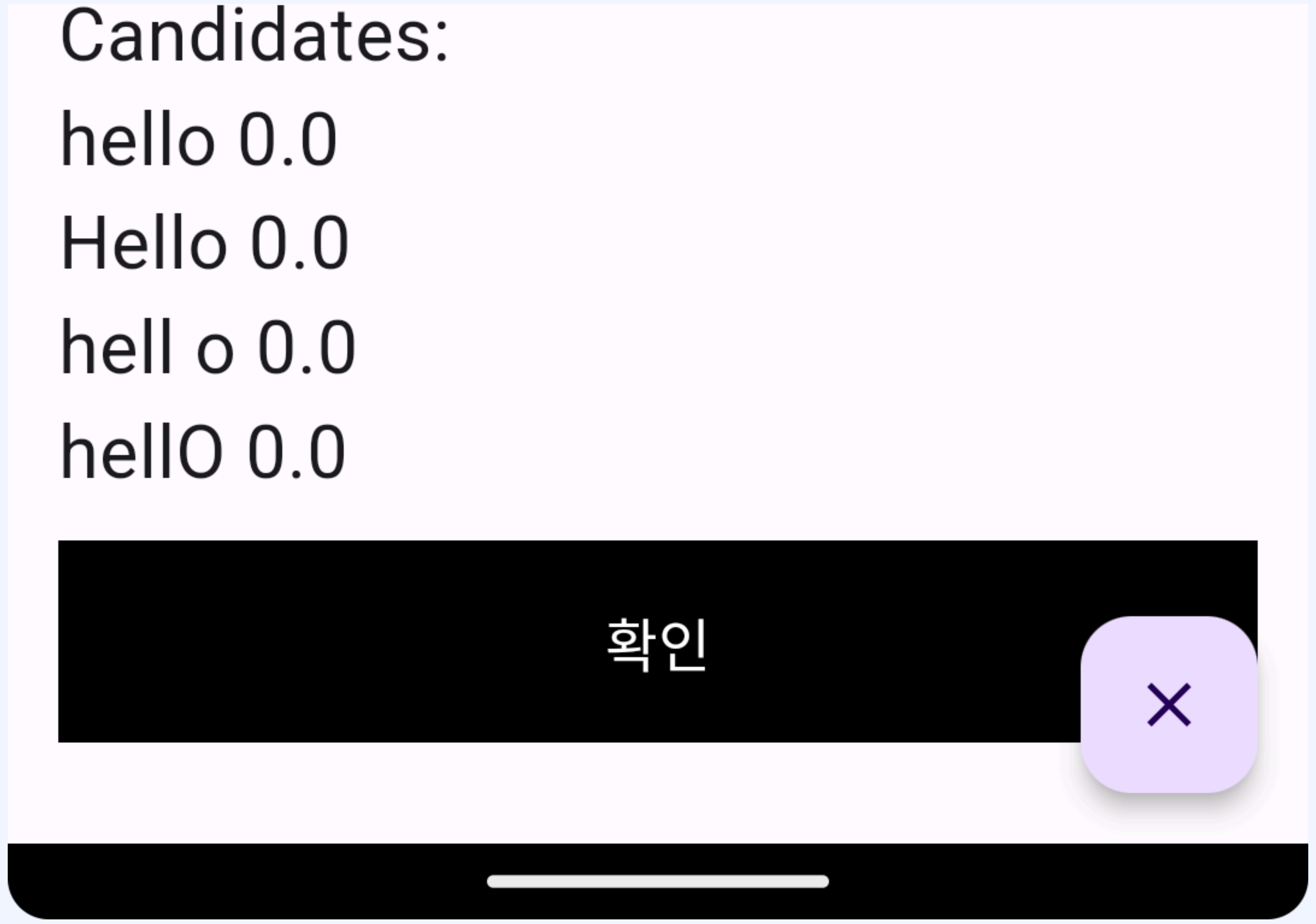
ML Kit - Digital ink recognition



```
Candidates:
hello 0.0
Hello 0.0
hell o 0.0
hellO 0.0
```





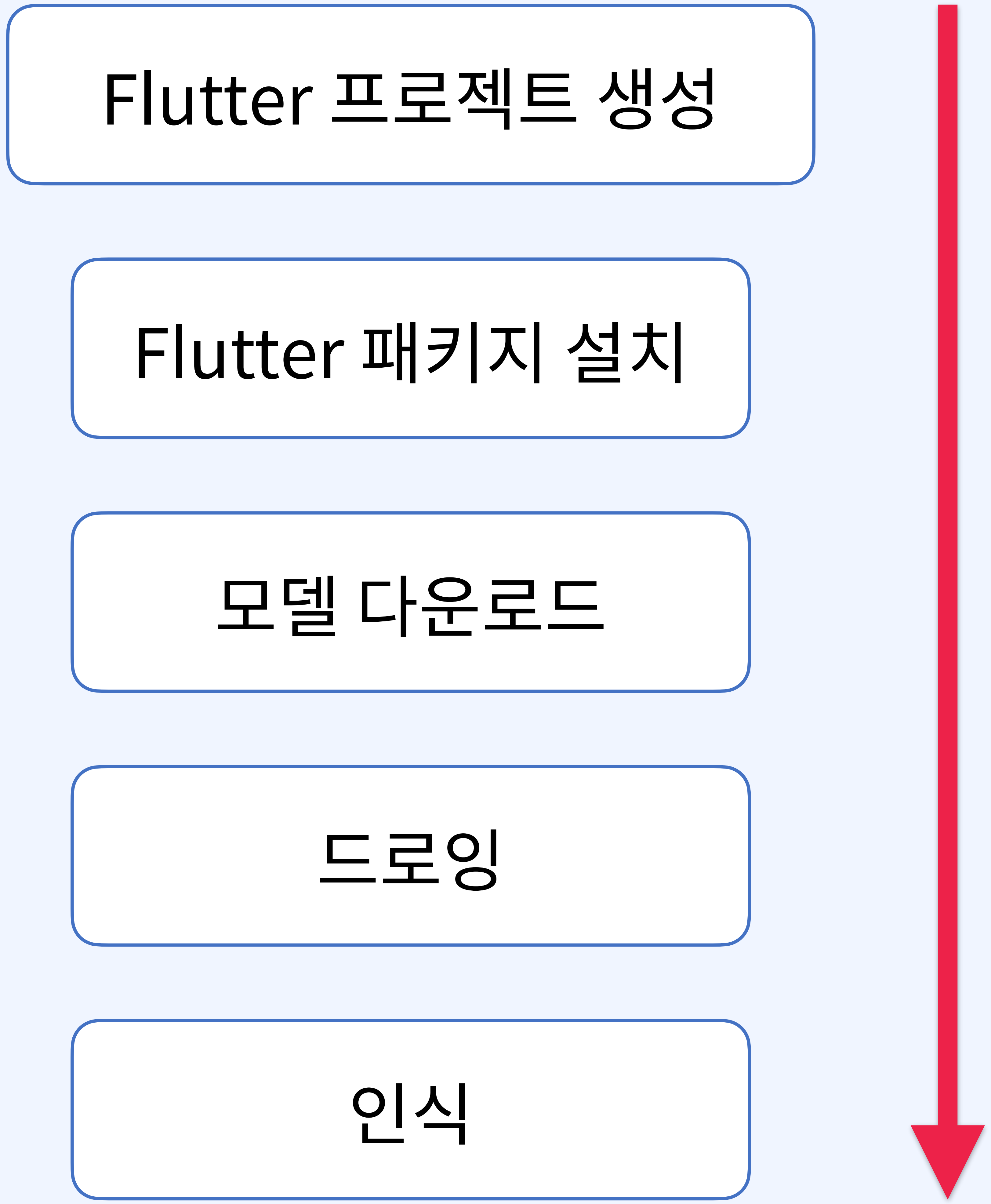


```
Candidates:
hello 0.0
Hello 0.0
hell o 0.0
hellO 0.0
```

Google ML Kit

ML Kit - Digital ink recognition
기능리뷰

ML Kit - Digital ink recognition



Flutter 프로젝트 생성

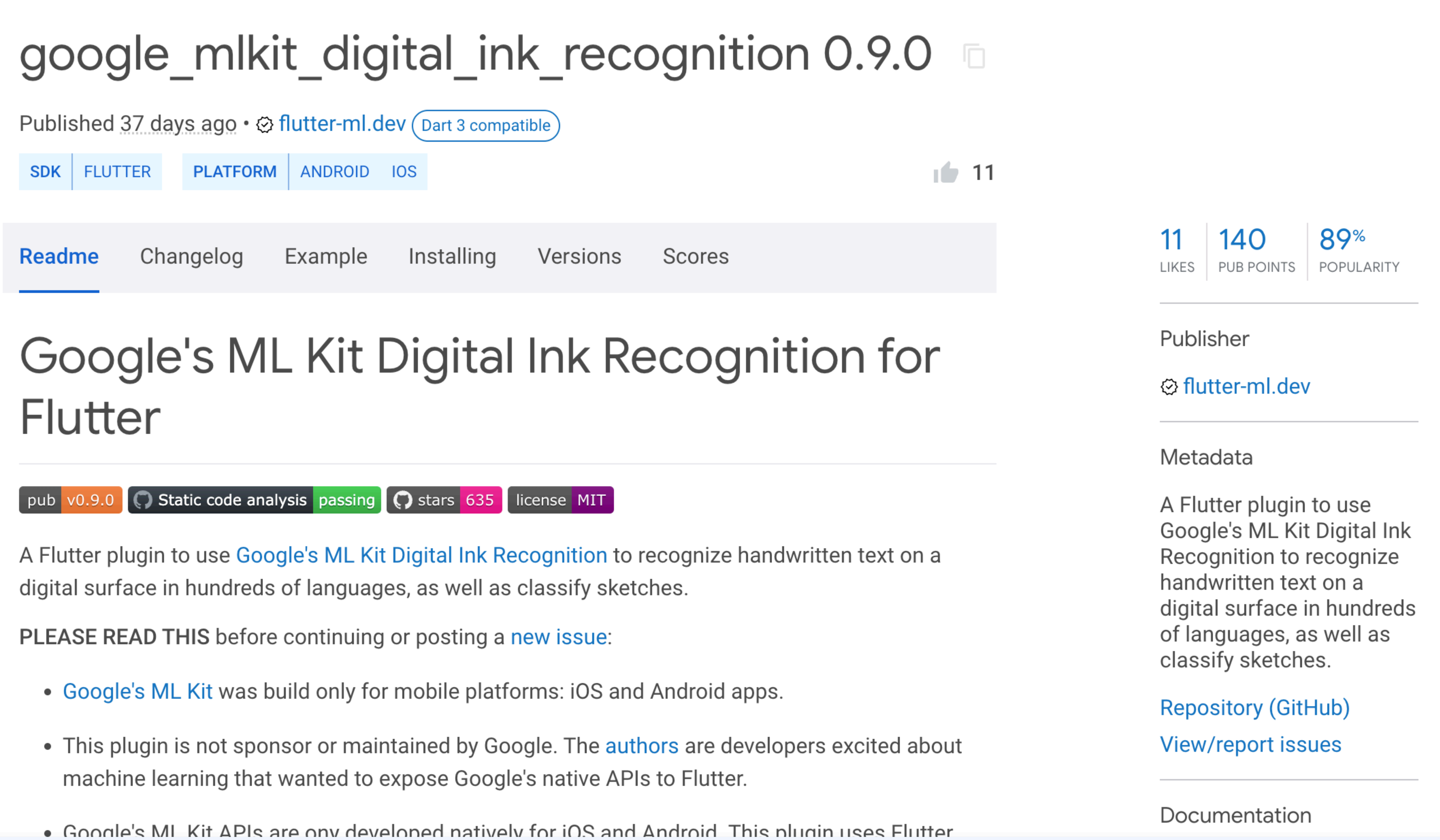
Flutter 패키지 설치

모델 다운로드

드로잉

인식

ML Kit - Digital ink recognition



google_mlkit_digital_ink_recognition 0.9.0

Published 37 days ago • flutter-ml.dev Dart 3 compatible

SDK FLUTTER PLATFORM ANDROID IOS 11

Readme Changelog Example Installing Versions Scores

Google's ML Kit Digital Ink Recognition for Flutter

pub v0.9.0 Static code analysis passing stars 635 license MIT

A Flutter plugin to use Google's ML Kit Digital Ink Recognition to recognize handwritten text on a digital surface in hundreds of languages, as well as classify sketches.

PLEASE READ THIS before continuing or posting a new issue:

- Google's ML Kit was build only for mobile platforms: iOS and Android apps.
- This plugin is not sponsor or maintained by Google. The authors are developers excited about machine learning that wanted to expose Google's native APIs to Flutter.
- Google's ML Kit APIs are ony developed natively for iOS and Android. This plugin uses Flutter

11 LIKES 140 PUB POINTS 89% POPULARITY

Publisher

flutter-ml.dev

Metadata

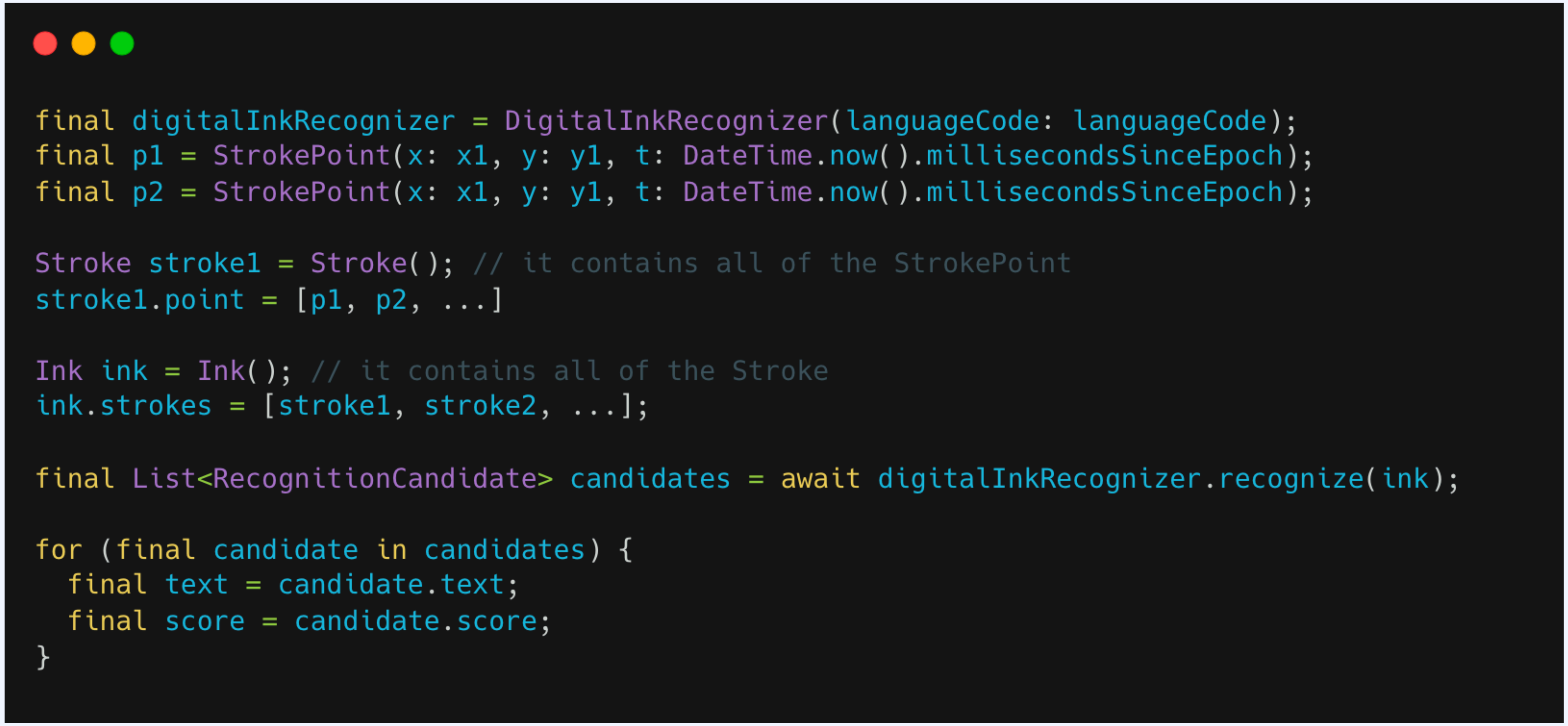
A Flutter plugin to use Google's ML Kit Digital Ink Recognition to recognize handwritten text on a digital surface in hundreds of languages, as well as classify sketches.

Repository (GitHub)
View/report issues

Documentation

https://pub.dev/packages/google_mlkit_digital_ink_recognition

ML Kit - Digital ink recognition



```
final digitalInkRecognizer = DigitalInkRecognizer(languageCode: languageCode);
final p1 = StrokePoint(x: x1, y: y1, t: DateTime.now().millisecondsSinceEpoch);
final p2 = StrokePoint(x: x1, y: y1, t: DateTime.now().millisecondsSinceEpoch);

Stroke stroke1 = Stroke(); // it contains all of the StrokePoint
stroke1.point = [p1, p2, ...]

Ink ink = Ink(); // it contains all of the Stroke
ink.strokes = [stroke1, stroke2, ...];

final List<RecognitionCandidate> candidates = await digitalInkRecognizer.recognize(ink);

for (final candidate in candidates) {
  final text = candidate.text;
  final score = candidate.score;
}
```

https://pub.dev/packages/google_mlkit_digital_ink_recognition

ML Kit - Digital ink recognition

모델 다운로드에 대한 처리
드로잉한 좌표에 대한 값 전달이 중요

이후 값 인식은 model 이 알아서 추론

Google ML Kit

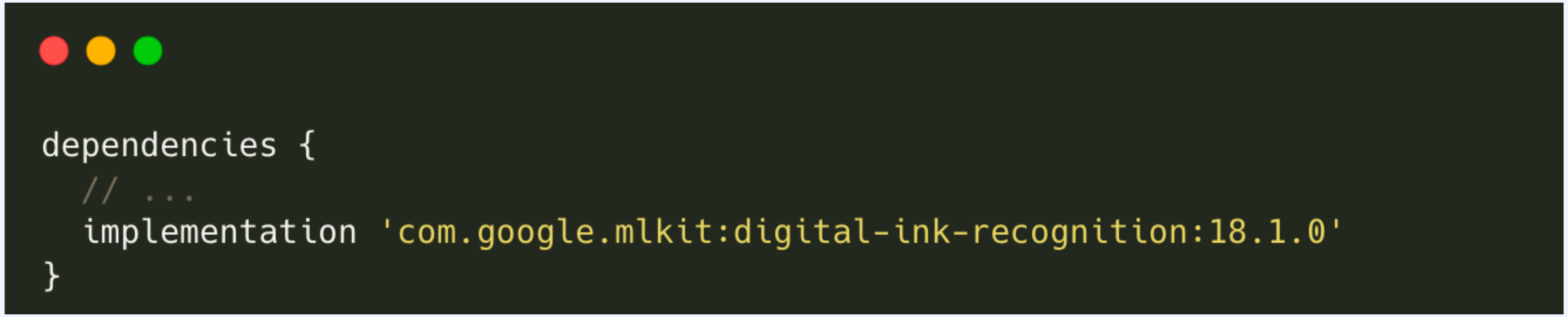
ML Kit - Digital ink recognition
구현 돌아보기

ML Kit - Digital ink recognition 구현 돌아보기

- 최소 요구사항
- Android
  - minSdkVersion: 21
  - targetSdkVersion: 33
  - compileSdkVersion: 33

ML Kit - Digital ink recognition 구현 돌아보기

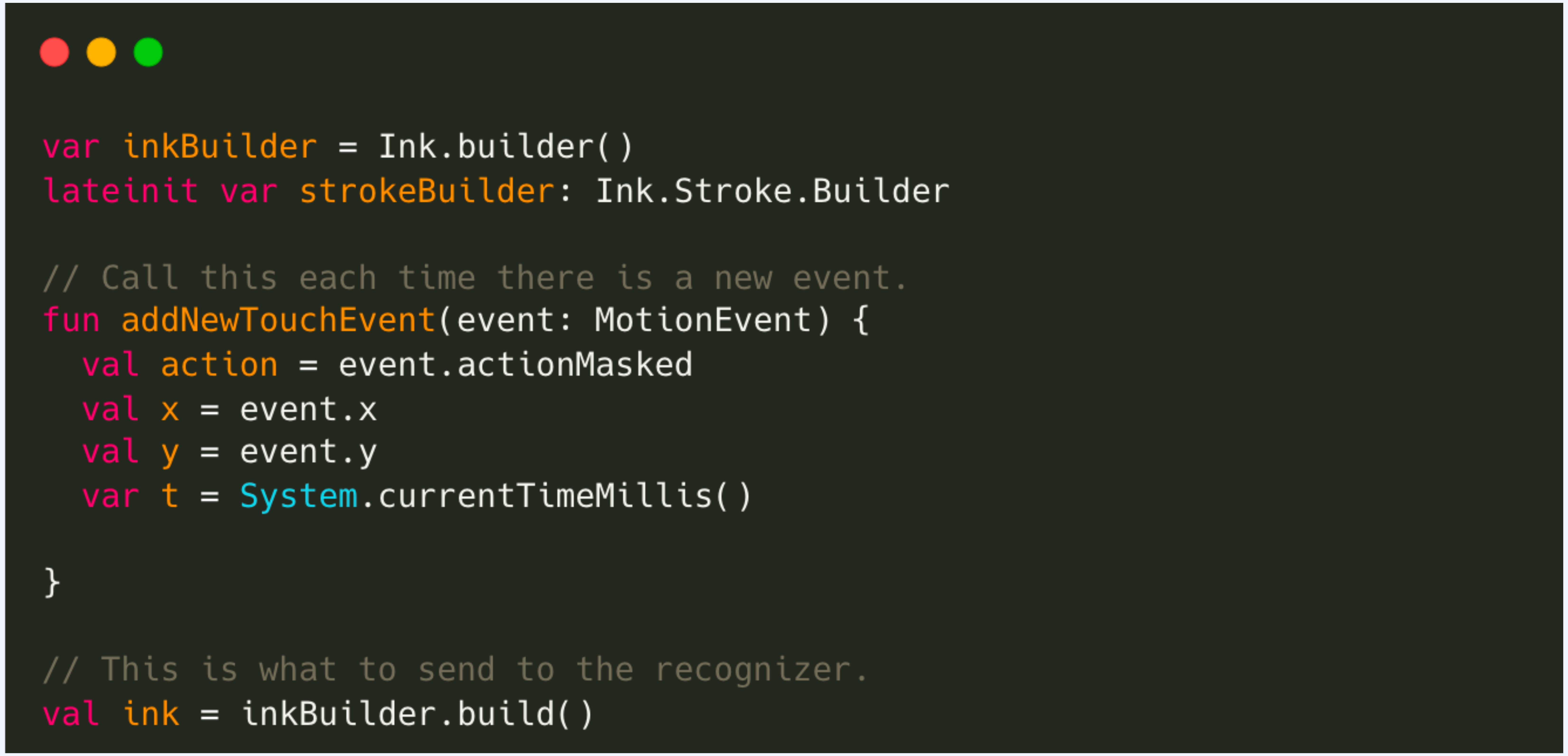
- Native - Android Side



```
dependencies {
  // ...
  implementation 'com.google.mlkit:digital-ink-recognition:18.1.0'
}
```

ML Kit - Digital ink recognition 구현 돌아보기

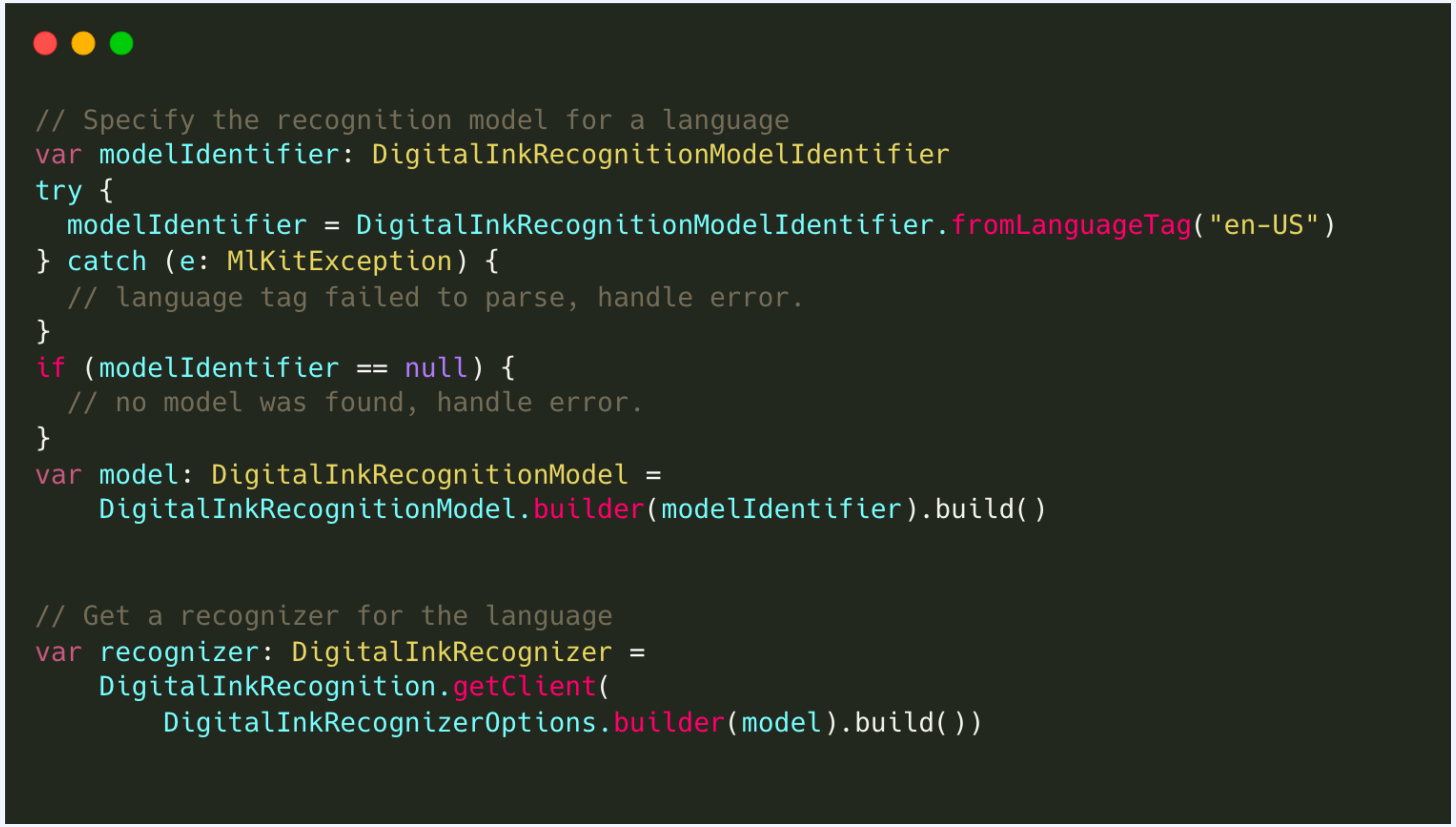
- Native - Android Side



```kotlin
var inkBuilder = Ink.builder()
lateinit var strokeBuilder: Ink.Stroke.Builder

// Call this each time there is a new event.
fun addNewTouchEvent(event: MotionEvent) {
  val action = event.actionMasked
  val x = event.x
  val y = event.y
  var t = System.currentTimeMillis()

}

// This is what to send to the recognizer.
val ink = inkBuilder.build()
```

ML Kit - Digital ink recognition 구현 돌아보기

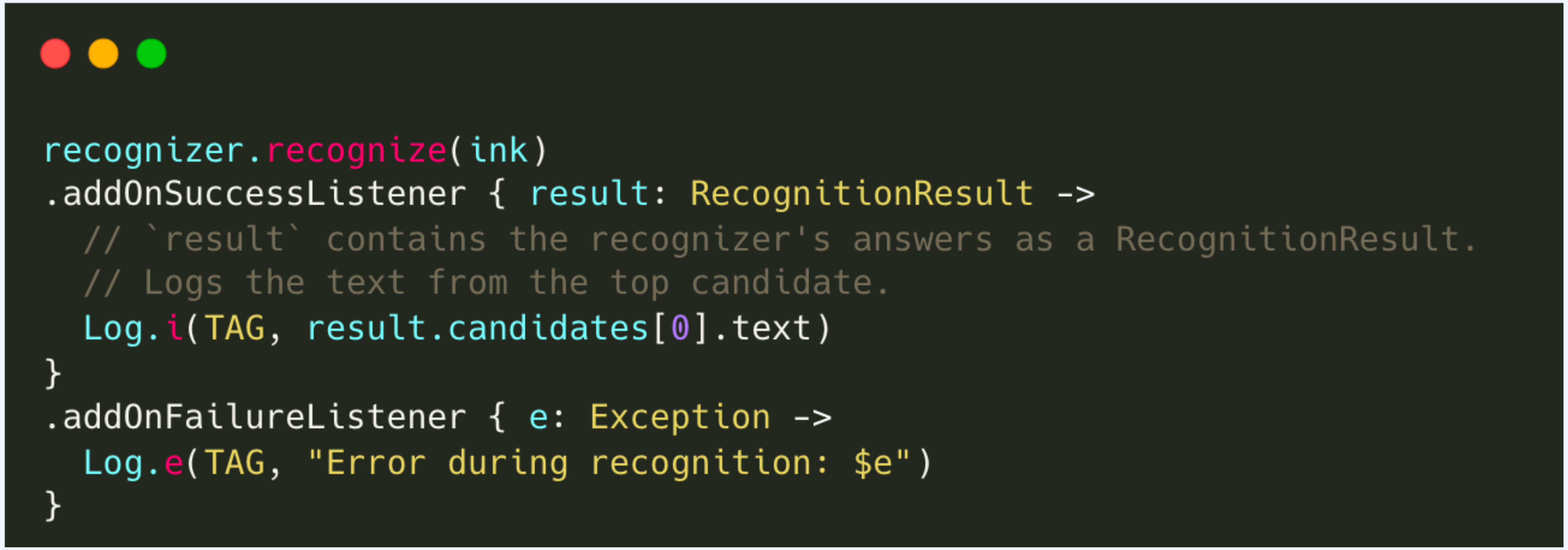
- Native - Android Side



```
// Specify the recognition model for a language
var modelIdentifier: DigitalInkRecognitionModelIdentifier
try {
  modelIdentifier = DigitalInkRecognitionModelIdentifier.fromLanguageTag("en-US")
} catch (e: MlKitException) {
  // language tag failed to parse, handle error.
}
if (modelIdentifier == null) {
  // no model was found, handle error.
}
var model: DigitalInkRecognitionModel =
    DigitalInkRecognitionModel.builder(modelIdentifier).build()


// Get a recognizer for the language
var recognizer: DigitalInkRecognizer =
    DigitalInkRecognition.getClient(
        DigitalInkRecognizerOptions.builder(model).build())
```

ML Kit - Face Detection 구현 돌아보기

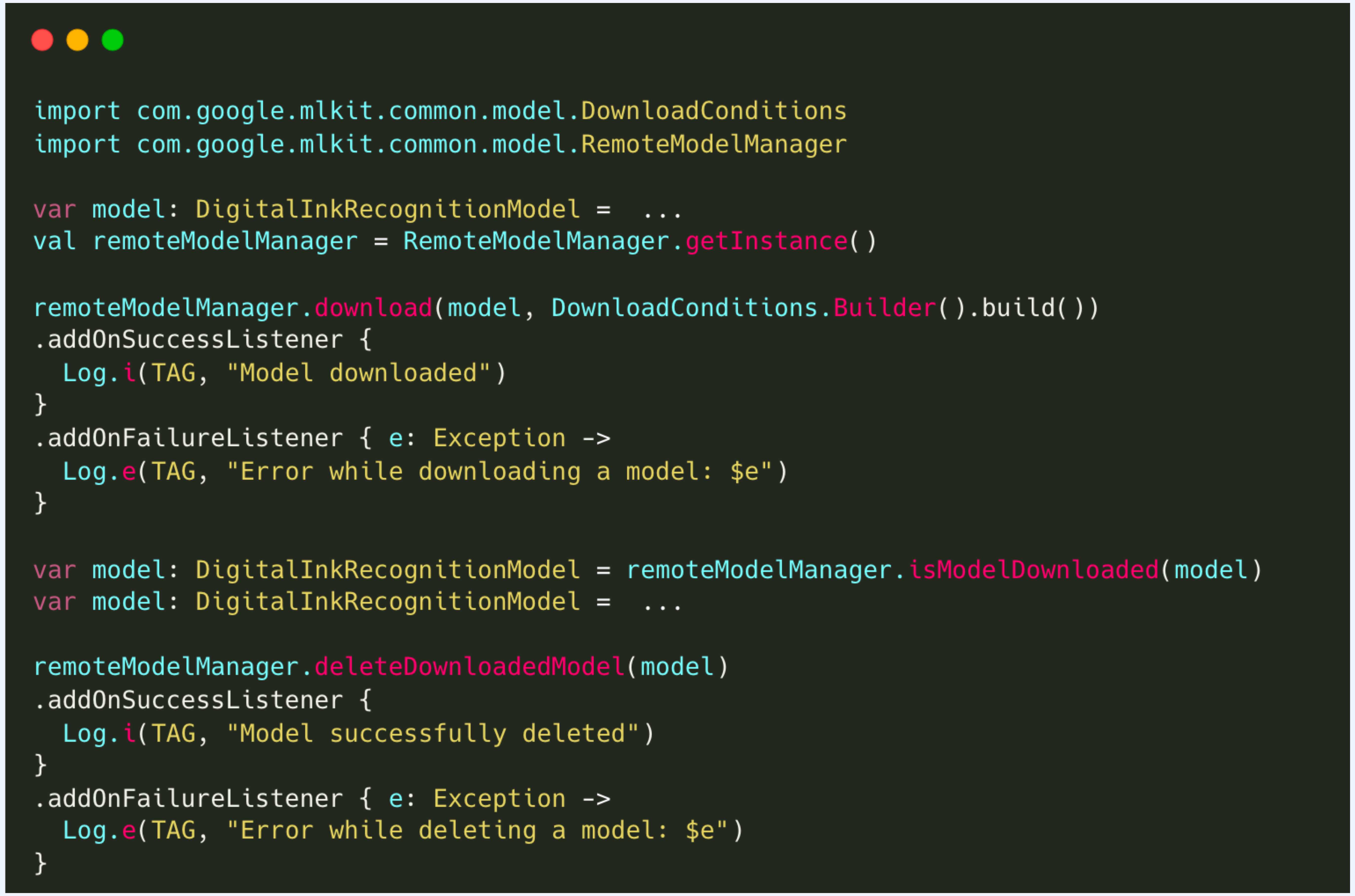
- Native - Android Side



```
recognizer.recognize(ink)
.addOnSuccessListener { result: RecognitionResult ->
  // `result` contains the recognizer's answers as a RecognitionResult.
  // Logs the text from the top candidate.
  Log.i(TAG, result.candidates[0].text)
}
.addOnFailureListener { e: Exception ->
  Log.e(TAG, "Error during recognition: $e")
}
```

ML Kit - Digital ink recognition 구현 돌아보기

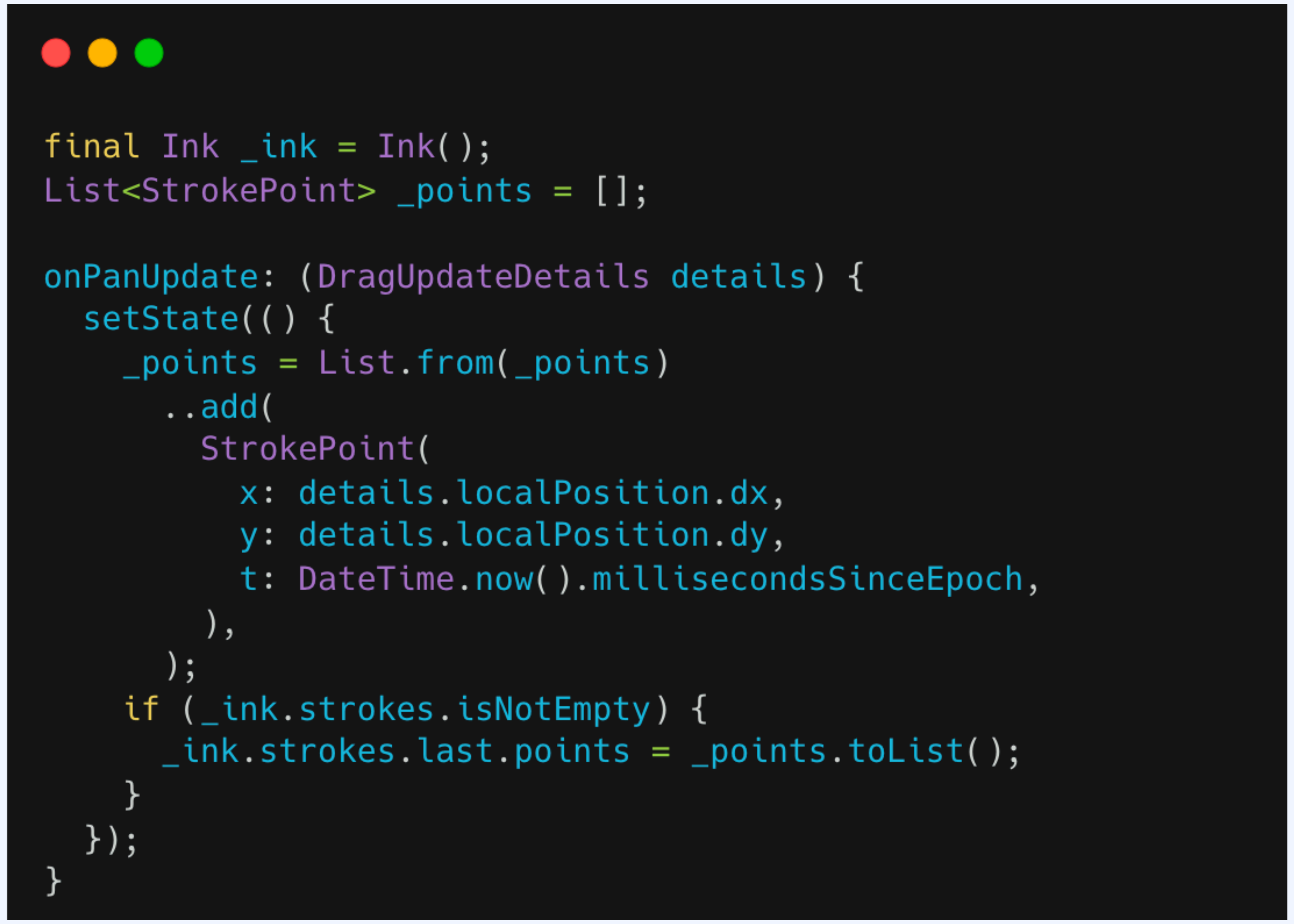
- Native - Android Side



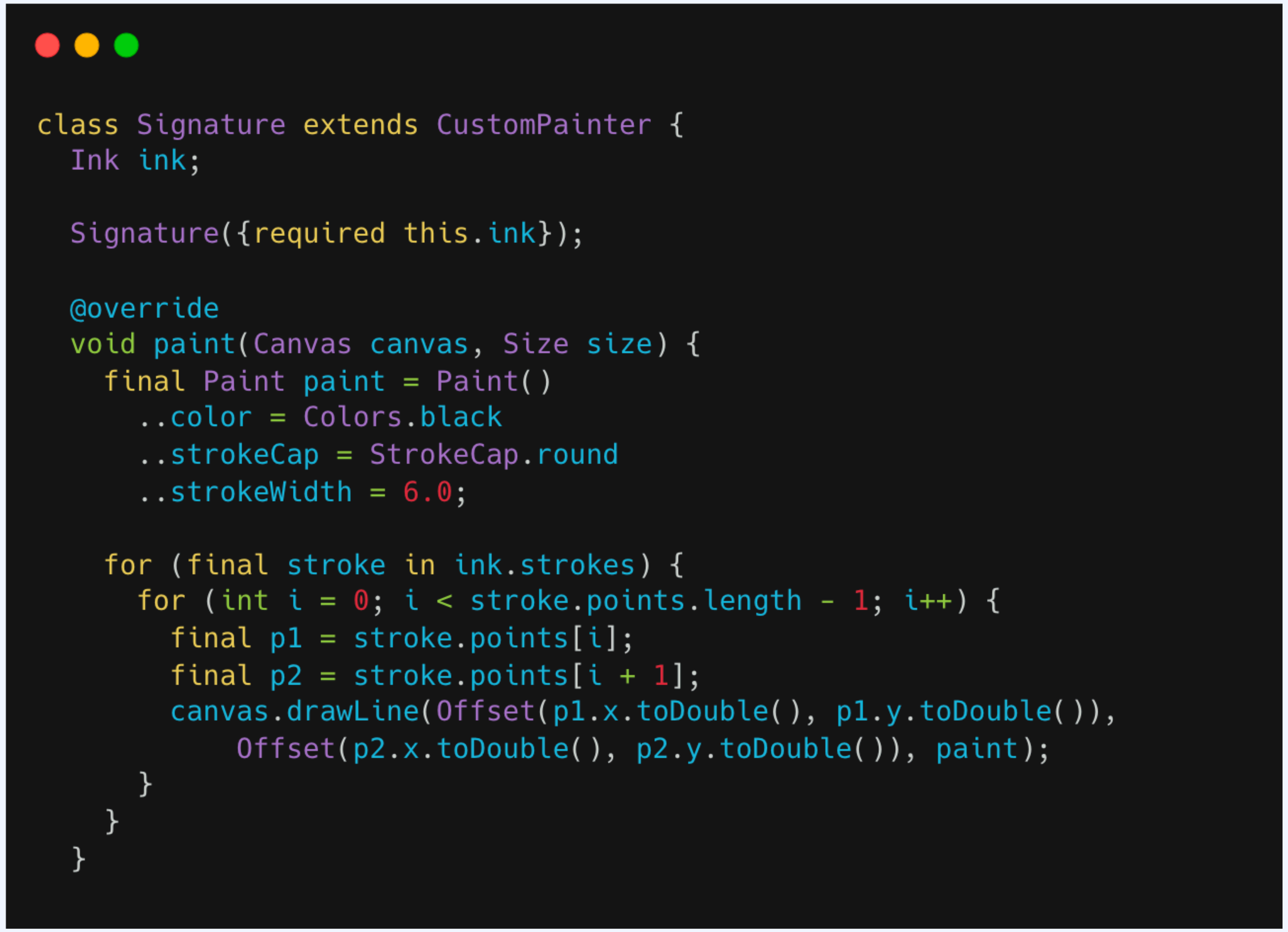
```
import com.google.mlkit.common.model.DownloadConditions
import com.google.mlkit.common.model.RemoteModelManager

var model: DigitalInkRecognitionModel =  ...
val remoteModelManager = RemoteModelManager.getInstance()

remoteModelManager.download(model, DownloadConditions.Builder().build())
.addOnSuccessListener {
  Log.i(TAG, "Model downloaded")
}
.addOnFailureListener { e: Exception ->
  Log.e(TAG, "Error while downloading a model: $e")
}

var model: DigitalInkRecognitionModel = remoteModelManager.isModelDownloaded(model)
var model: DigitalInkRecognitionModel =  ...

remoteModelManager.deleteDownloadedModel(model)
.addOnSuccessListener {
  Log.i(TAG, "Model successfully deleted")
}
.addOnFailureListener { e: Exception ->
  Log.e(TAG, "Error while deleting a model: $e")
}
```

ML Kit - Digital ink recognition 구현 돌아보기

- Flutter Side



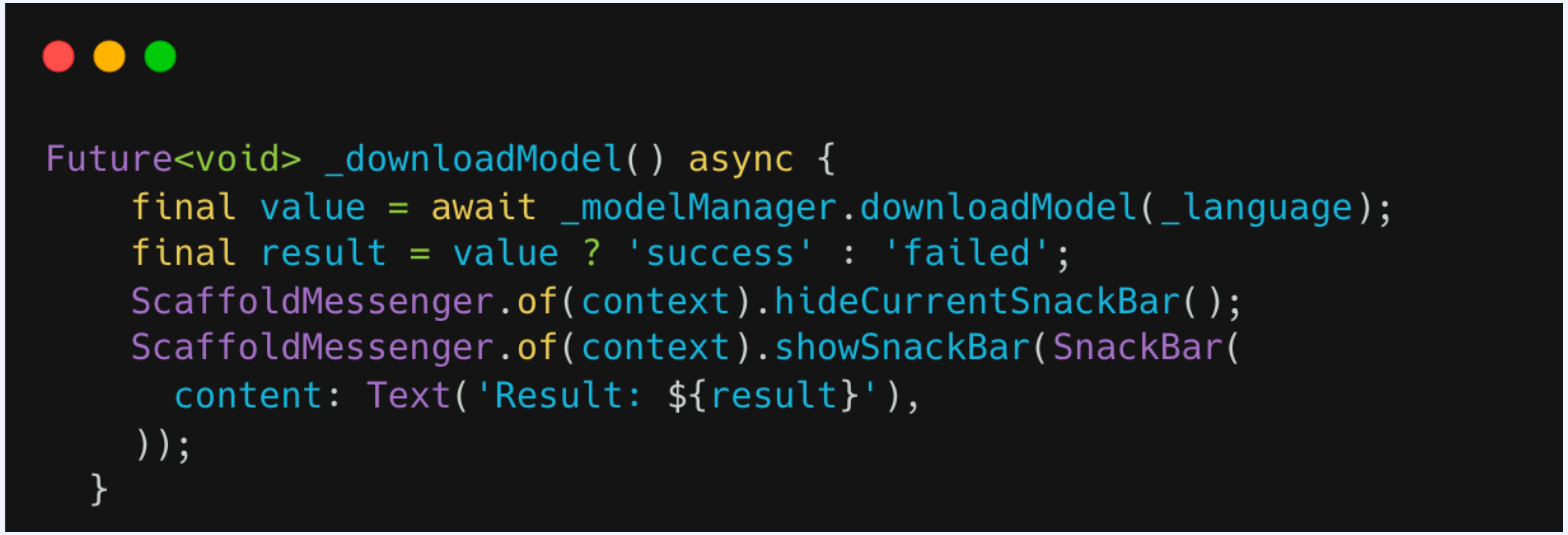
```dart
final Ink _ink = Ink();
List<StrokePoint> _points = [];

onPanUpdate: (DragUpdateDetails details) {
  setState(() {
    _points = List.from(_points)
      ..add(
        StrokePoint(
          x: details.localPosition.dx,
          y: details.localPosition.dy,
          t: DateTime.now().millisecondsSinceEpoch,
        ),
      );
    if (_ink.strokes.isNotEmpty) {
      _ink.strokes.last.points = _points.toList();
    }
  });
}
```



```dart
class Signature extends CustomPainter {
  Ink ink;

  Signature({required this.ink});

  @override
  void paint(Canvas canvas, Size size) {
    final Paint paint = Paint()
      ..color = Colors.black
      ..strokeCap = StrokeCap.round
      ..strokeWidth = 6.0;

    for (final stroke in ink.strokes) {
      for (int i = 0; i < stroke.points.length - 1; i++) {
        final p1 = stroke.points[i];
        final p2 = stroke.points[i + 1];
        canvas.drawLine(Offset(p1.x.toDouble(), p1.y.toDouble()),
            Offset(p2.x.toDouble(), p2.y.toDouble()), paint);
      }
    }
  }
```

ML Kit - Digital ink recognition 구현 돌아보기

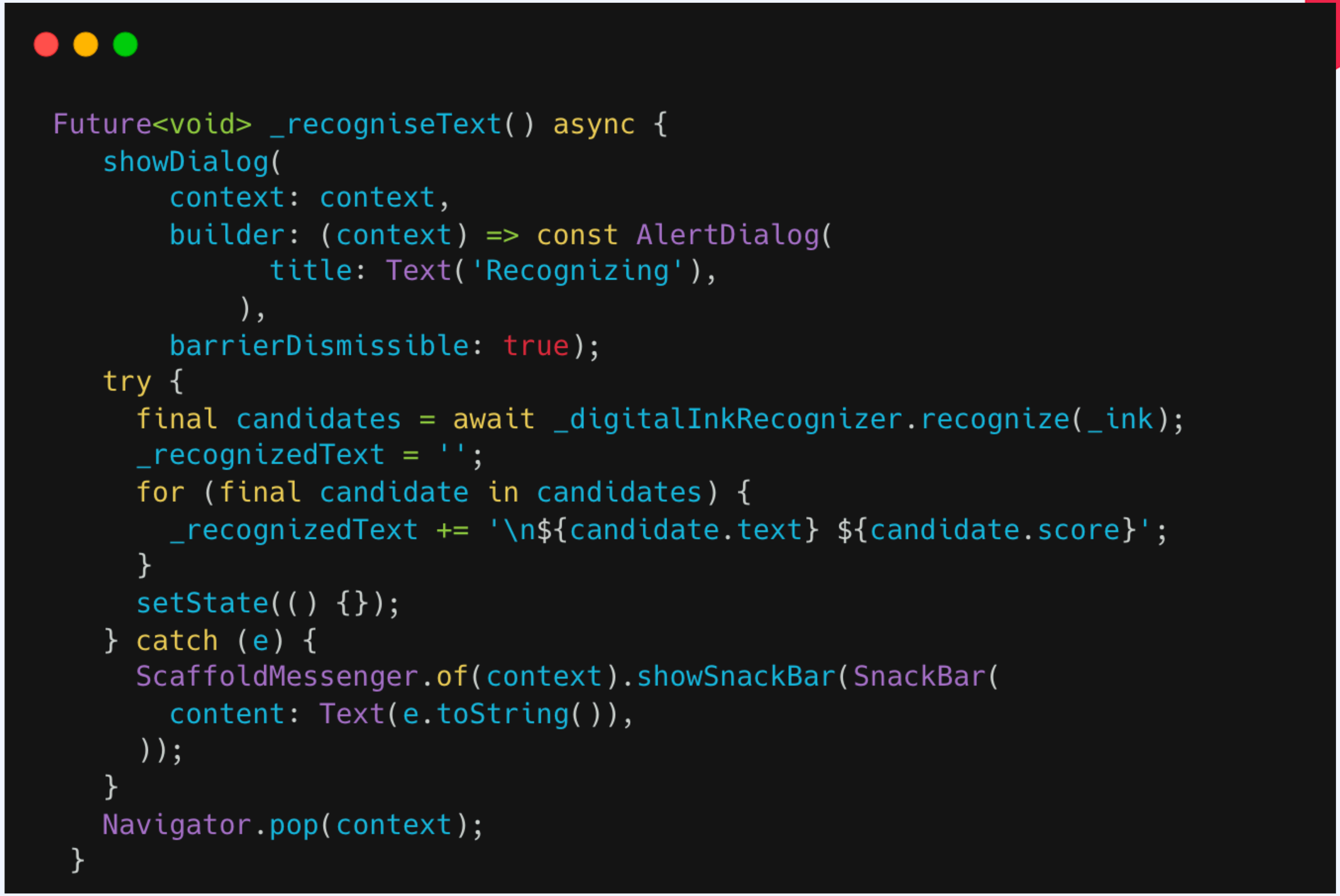
- Flutter Side



```
Future<void> _downloadModel() async {
    final value = await _modelManager.downloadModel(_language);
    final result = value ? 'success' : 'failed';
    ScaffoldMessenger.of(context).hideCurrentSnackBar();
    ScaffoldMessenger.of(context).showSnackBar(SnackBar(
      content: Text('Result: ${result}'),
    ));
  }
```

ML Kit - Digital ink recognition 구현 돌아보기

- Flutter Side



```
Future<void> _recogniseText() async {
   showDialog(
       context: context,
       builder: (context) => const AlertDialog(
             title: Text('Recognizing'),
           ),
       barrierDismissible: true);
   try {
     final candidates = await _digitalInkRecognizer.recognize(_ink);
     _recognizedText = '';
     for (final candidate in candidates) {
       _recognizedText += '\n${candidate.text} ${candidate.score}';
     }
     setState(() {});
   } catch (e) {
     ScaffoldMessenger.of(context).showSnackBar(SnackBar(
       content: Text(e.toString()),
     ));
   }
   Navigator.pop(context);
 }
```