

# スクリーンショットを自動で取得する Android デバイスのスワイププログラム

## 1. 概要

この Python プログラムは、Android デバイス上で自動的にスクリーンショットを取得し、ユーザーが指定した方向にスワイプする機能を提供しています。プログラムは Python 言語を使用し、Android Debug Bridge (ADB) を介してデバイスとの通信を行います。具体的には、`take_screenshot` 関数が ADB を使用してスクリーンショットを取得し、`swipe` 関数が指定された方向にスワイプを模倣します。また、`capture_swipe_and_save` 関数は指定されたページ数分のスクリーンショットを取得し、指定された方向にスワイプしながら、整理された形で保存します。これにより、プログラムは柔軟性と効率性を備え、特定の使用ケースに適した Android デバイス上のユーザーアクションを自動的に模倣します。

Android デバイス上のユーザーアクションを自動的に模倣します。Kindle アプリにて、保存したい書籍をスクリーンショットにて保存するために作成しました。

## 2. 方法

プログラムの実行には、まず Python がインストールされており、Android Debug Bridge (ADB) が使えることが前提となります。ADB の設定は、<https://sp7pc.com/google/android/34263> の Web ページを参考にして設定しました。その後、Android デバイスを PC に接続し、デバッグモードを有効にしました。プログラムをテキストエディタで開き、`capture_swipe_and_save` 関数内のパラメータを設定し、コマンドラインまたはターミナルでプログラムの保存ディレクトリに移動します。最後に、コマンド `python` プログラムのファイル名.py (または `python3` を使用している場合) を実行することで、Android デバイス上で指定したページ数分のスクリーンショットを取得し、指定した方向にスワイプして保存します。

### 3. 結果

このプログラムを実行すると、指定されたページ数分のスクリーンショットが Android デバイス上で取得され、さらに指定された方向にスワイプされ、それぞれのスクリーンショットが指定されたディレクトリに保存されます。各スクリーンショットは、指定されたファイルパスとプレフィックスに基づいて連番が付与され、整理された形で格納されます。

この結果を利用することで、特定のアプリケーションやウェブページにおける異なる状態やコンテンツの変化を観察することが可能です。また、取得されたスクリーンショットは、デバッグ、テスト、もしくはドキュメンテーションの目的に利用することができます。プログラムの柔軟性により、異なる方向やページ数、待機時間の設定が可能であり、ユーザーが特定のテストケースやユースケースに合わせて利用できます。