

## bunny 0.9.0 導入ガイド（ドラフト）

### 動作環境

Bunny は GHC (Haskell Platform) が動作する Windows (MINGW) または Linux 上で動作可能です。ビルドするためには Haskell Platform に含まれるもの以外に、以下のツール・ライブラリをあらかじめインストールしておく必要があります。

ツール：

- hlint
- Alex
- Happy

ライブラリ：

- ansi-wl-pprint
- optparse-applicative

Bunny を利用して Android アプリを作成するには、Android Studio<sup>\*1</sup> / Android SDK Platform Tools<sup>\*2</sup> が必要です。Linux の場合は、Android Studio に platform tools も同梱されているようです。Windows の場合には、Android Studio とは別に Platform Tools をインストールする必要があるようでした。

Platform Tools をインストールしたら、パスを通しておいてください。javac/java にもパスが通っている必要があります。これは、Android Studio に同梱されているものを用いるようにした方が確実です<sup>\*3</sup>。Linux の場合には、`$INSTALL_DIR/android-studio/jre/bin` に、Windows の場合には `/c/Program Files/Android/Android Studio/jre/bin` にパスを通しておくことになります。

パスが通っていることの確認例を以下に示します。具体的なパスは環境毎に異なります。

確認例 (Linux の場合)：

```
$ which javac
/home/<user name>/android-studio/jre/bin/javac

$ which java
/home/<user name>/android-studio/jre/bin/java

$ which adb
/home/<user name>/Android/Sdk/platform-tools/adb
```

---

\*1 <https://developer.android.com/studio?hl=ja>

\*2 <https://developer.android.com/studio/releases/platform-tools?hl=ja>

\*3 要修正：testrun で作った class ファイルが Android Project に混入しないように、bunny コマンド側で対処すべき。

確認例 (Windows (MINGW) の場合) :

```
$ which javac
/c/Program Files/Android/Android Studio/jre/bin/javac

$ which java
/c/Program Files/Android/Android Studio/jre/bin/java

$ which adb
/c/Users/<user name>/android/platform-tools/adb
```

作者の手元では、Linux, Windows それぞれ以下の環境で動作を確認済です。

- Windows 10
  - MINGW64\_NT-10.0
  - Haskell Platform 8.2.2
  - Gnu make 3.81
  - Android Studio 4.4.1 / Android SDK Platform tools r30.0.5
- Linux
  - Ubuntu 20.04.1 LTS
  - Haskell Platform 2014.2.0.0.debian8 (apt install haskell-platform したもの)
  - Gnu make 4.2.1
  - Android Studio 4.0.2

## コンパイル

ソースファイルを展開したら、compiler ディレクトリに移動し make してください。

```
$ cd compile
$ make
```

コンパイルが成功したら、簡単な Haskell プログラムを Bunny でコンパイルして動作確認します。

```
$ bin/bunny testrun sample/Hello.hs
```

実行過程を示す文字列や、コンパイルにおける警告などが表示されるかもしれませんが、その後に以下のように表示されたら動作確認は成功です。

```
Hello, World!  
こんにちは世界!
```

動作確認ができたなら、`install.sh` をつかってインストールします。そのまえに、環境に応じた `local.properties` ファイルを用意する必要があります。`~/AndroidStudioProjects/` など、既存の Android Studio プロジェクトが格納されたディレクトリから、適当なプロジェクトの `local.properties` ファイルをコピーして用います\*4。(以下は、`MyFirstApp` プロジェクトからコピーしていますがどこからもらっても同じです)

```
cp ~/AndroidStudioProjects/MyFirstApp/local.properties AndroidProjectPrototype/
```

その後、`install.sh` スクリプトを用いてインストールします。

```
$ ./install.sh
```

デフォルトでは、`$HOME/bunny/0.9.0` の下にインストールされます。`$HOME` 以外の場所にインストールしたい場合には、`install.sh` の引数としてパスを与えます。

インストールできたら、`$HOME/bunny/0.9.0/bin` (または、インストール場所を指定した場合にはそれに応じたパス) を `PATH` に加えてください。

正常にパスを通せたら、`bunny --version` と打ってみてください。以下のように表示されるはずです。

```
$ bunny --version
```

```
Bunny - A Haskell compiler for android, version 0.9.0 (Basecamp)
```

---

\*4 もし Android Studio をインストールした直後でプロジェクトがひとつも存在しない場合は、Android Studio を起動して、なんでもいいのでひとつ新しいプロジェクトを作っておいてください。