

第1回BlancoFW勉強会 作業手順 (Win版)

文書の目的：作業を確実に再現させる

※別紙「第1回BlancoFW勉強会 作業手順 (Win版) 作業・Log 画像.pdf」ファイルも併せて参照ください

1. ツールなどのインストールと設定

1. 「IntelliJ IDEA CE」をインストール (→[操作画像1](#))

インストーラでインストール

<https://www.jetbrains.com/idea/>

a. 折り返し (soft wrap) の設定 (→[操作画像2](#))

Preferences > Editor > General パネルで Soft Wraps の「Use soft wraps in editor」にチェックを入れる

2. JDK (Java Development Kit) のインストール (→[操作画像3](#))

すでにインストールされているか、されているならバージョンを確認。

コマンドプロンプトで

```
>java -version
```

を実行

- a. インストールされていない場合 → インストールが必要です
- b. 以前のバージョンが表示される場合
(ver 1.6以上なら動作しますが、最新バージョンをインストールします)
 - 最新バージョンのインストールが必要です。
 - 環境変数の設定によるバージョンの切り替えが必要です。
- c. 最新バージョン (10.0.2) が表示される場合 → 環境変数の設定を確認してください。

参考

インストールされていない場合

```
1 >java -version
2 'java' は、内部コマンドまたは外部コマンド、
3 操作可能なプログラムまたはバッチ ファイルとして認識されていません。
```

以前のバージョンが表示される場合 (例はバージョン1.8)

```
1 >java -version
```

```
2 java version "1.8.0_181"
3 Java(TM) SE Runtime Environment (build 1.8.0_181-b13)
4 Java HotSpot(TM) 64-Bit Server VM (build 25.181-b13, mixed mode)
```

最新バージョン (10.0.2) が表示される場合

```
1 >java -version
2 java version "10.0.2" 2018-07-17
3 Java(TM) SE Runtime Environment 18.3 (build 10.0.2+13)
4 Java HotSpot(TM) 64-Bit Server VM 18.3 (build 10.0.2+13, mixed mode)
```

インストール

(参考 : <https://eng-entrance.com/java-install-jdk-windows>)

- a. インストーラをダウンロードしてインストール

[公式ページ](#)で、「Accept License Agreement」にチェックを入れ、インストーラーをダウンロード、ダブルクリックしてインストール

- b. インストールの確認

コマンドプロンプトを一度閉じ、開いてください

例)

```
1 >java -version
2 java version "10.0.2" 2018-07-17
3 Java(TM) SE Runtime Environment 18.3 (build 10.0.2+13)
4 Java HotSpot(TM) 64-Bit Server VM 18.3 (build 10.0.2+13, mixed mode)
```

不要なファイルの削除と環境変数の設定

1. C:\Windows\System32\のjava*.exeが (あれば) 削除
2. 環境変数JAVA_HOMEにC:\Program Files\Java\jdk-10.0.2を設定。
3. 環境変数PATHに%JAVA_HOME%\binを追加。
4. 環境変数PATHからC:\Program Files (x86)\Common Files\Oracle\Java\javapathを (あれば) 削除。
5. 環境変数PATHからC:\ProgramData\Oracle\Java\javapathを (あれば) 削除。

バージョンの確認

```
1 >java -version
2 java version "10.0.2" 2018-07-17
3 Java(TM) SE Runtime Environment 18.3 (build 10.0.2+13)
4 Java HotSpot(TM) 64-Bit Server VM 18.3 (build 10.0.2+13, mixed mode)
```

※意図していないバージョンが表示される場合

レジストリに古いバージョンのデータが残っている可能性があるので、マイクロソフトから提供されている[自動修正プログラム](#) `MicrosoftProgram_Install_and_Uninstall.meta.diagcab` で「アンインストール」を選び、必要なバージョン以外のJavaを削除する (参考)

3.Antのインストール

- すでにインストールされているか確認

```
> ant -version
```

- インストール

圧縮ファイルをダウンロードしてインストール

(参考: <http://note.onichannn.net/archives/3243>)

- 公式ページで最新バージョンを確認 (ver 1.10.5)
- 任意のディレクトリにファイルをダウンロードして解凍
- 解凍したファイルをc:\ に移動

- 環境変数の設定

- ANT_HOMEの設定

「システム環境変数」の「ANT_HOME」の値にC:\apache-ant-1.10.5をセット

- PATHの設定

「システム環境設定」で「PATH」を選択し、%ANT_HOME%\binを追加

- インストールの確認

- > ant -version

```
Apache Ant(TM) version 1.10.5 compiled on July 10 2018
```

- > set ANT_HOME

```
ANT_HOME=C:\apache-ant-1.10.5
```

- > set PATH

```
C:\apache-ant-1.10.5\bin; が含まれていることを確認
```

※ java のver 1.8未満を使用する場合、Ant のver 1.9.13をインストール、環境変数を設定してバージョンを切り替える (今回は使用しない)

[AntのJava対応バージョン](#)

Ant Version	Minimum Java Version
Any 1.9.x release and the git branch 1.9.x	1.5
Any 1.10.x release and the current git master branch	1.8

4.SQLite3のインストール

- すでにインストールされているか確認

```
> sqlite3 -version
```

- されていなければインストール

- 公式ページから圧縮ファイルsqlite-tools-win32-x86-3240000.zipをダウンロードして解凍 (フォルダ名sqlite-tools-win32-x86-3240000)

- 解凍したファイルをc:\ に移動

- 環境変数の設定

- SQLITE_HOMEの設定

「システム環境変数」の「SQLITE_HOME」の値にC:\sqlite-tools-win32-x86-3240000をセット

- PATHの設定

「システム環境設定」で「PATH」を選択し、%SQLITE_HOME%を追加

- インストールの確認

- > sqlite3 -version

```
3.24.0 2018-06-04 19:24:41
```

```
c7ee0833225bfd8c5ec2f9bf62b97c4e04d03bd9566366d5221ac8fb199a87ca
```

- > set SQLITE_HOME

```
SQLITE_HOME=C:\sqlite-tools-win32-x86-32400005
```

- `> set PATH`
`C:\sqlite-tools-win32-x86-3240000;`が含まれていることを確認

5. (gitのインストール.....は、本来は必要ですが割愛)

2. antでBlancoDbEEをビルドし、JDBCコードを生成する

1.以下のgit branchを同じディレクトリに配置 (git clone または zipファイルをダウンロードして解凍)

- <https://github.com/igapyon/blancoDbEE>
- <https://github.com/igapyon/blanco2gDistribution>

配置のイメージ

```
1 .
2 |— blanco2gDistribution
3 |— blancoDbEE
```

→ `./blancoDbEE` フォルダに移動

2.IntelliJ IDEA CE で `./blancoDbEE` ディレクトリを読み込み

3./blancoDbEE/build.xmlを変更 (1箇所)

```
1 <171行目>
2 -         <blancodb basepackage="my.db" jdbcdriver="org.sqlite.JDBC"
          jdbcurl="jdbc:sqlite:./test/data/sqlite/sqlite.db" jdbcuser="sqlite" jdbcpass
          word="password" metadir="test/data/sqlite/" targetdir="test/result/blanco"
          sql="true" table="true" encoding="Shift-JIS" />
3 +         <blancodb basepackage="my.db" jdbcdriver="org.sqlite.JDBC"
          jdbcurl="jdbc:sqlite:./test/data/sqlite/sqlite.db" jdbcuser="sqlite" jdbcpass
          word="password" metadir="test/data/sqlite/" targetdir="test/result/blanco"
          sql="true" table="true" encoding="UTF-8" />
```

4. `./blancoDbEE`ディレクトリで `$ ant meta` 実行 (→参考log.1)

このとき、

- `./blancoDbEE/tmp`

ディレクトリが作成されています。

再実行する場合にはこのディレクトリを削除してから行ってください。

5.成果物の確認

- `./blancoDbEE/test/result/blanco/main/my/db/`以下のファイル

3. 生成したJDBCをテストする

1. IntelliJ IDEA CE で `./blancoDbEE` ディレクトリを読み込む

2. (IntelliJ IDEA CEの設定1) 使用するJDKを設定する (→[操作画像4](#))

3. (IntelliJ IDEA CEの設定2) `./blancoDbEE/test/src/sqlite`ディレクトリを「open module setting」から**Sources**にマークする (→[操作画像5](#))

4. (IntelliJ IDEA CEの設定3) 以下のディレクトリをライブラリに登録する (→[操作画像6](#))

- `./blancoDbEE/lib`
- `./blancoDbEE/lib.ant`
- `./blancoDbEE/lib.test`

5. IntelliJ IDEA CE で

`./blancoDbEE/test/src/sqlite/SimpleSqliteTester.java` を実行する (→[操作画像7](#))

ツールウィンドウで「`./blancoDbEE/test/src/sqlite/SimpleSqliteTester.java`」を右クリック、Run

エラーが表示されるが、実行ファイルがビルドされていないため。ビルドは自動で行われ、`./blancoDbEE/bin`フォルダが作成され、再実行の際にはエラーは出ない。

実行結果

```
1 "C:\Program Files\Java\jdk-10.0.2\bin\java.exe" ...
2 Search result: [Field 1 value]
3
4 Process finished with exit code 0
```

以上