msys2インストール

msys2をインストールする。

確認動作環境

- Windows11（64bit）

インストール

下記URLのインストールファイルを実行しインストール（64bit版、32bit版のインストーラがあるので自分のWindowsOSに合う方を使う。

<http://www.msys2.org/>

- 64bit版「`msys2_x86_64_*.exe`」のファイル。
- インストール先は、デフォルトの「`C:¥msys64`」で 進める。（必要あれば変更してもよい）

msys2 ucrt64 とは

「MSYS2 UCRT64[1)]」とは、Windows環境でC言語やC++などのネイティブアプリを開発するための代表的な64bit開発環境・ツール群の一つで、MSYS2に追加されていた。
これまで、MINGW64環境を利用していたが、最近ではこの、UCRT64環境の利用が推奨されている模様。

今後は、この「msys2 ucrt64」を使って行こうと思うので、ここの記事もその環境に合わせた記述に見直していく。(2026/6/1追記）

初回アップデート

インストールが終わると、ターミナルが開くので、下記コマンドを入力して、アップデートする。ターミナルが開かない場合は、インストール先の`ucrt64.txt`を実行し更新する。

```
$ pacman -Syu↓
```

- 画面サイズや文字サイズは、左上アイコンの右クリックメニューの Options... から変更できる。

開発ツールのインストール

パッケージのインストールは、`pacman -S <パッケージ名>`でインストールする。

```
$ pacman -S base-devel
$ pacman -S msys2-devel
```

```
$ pacman -S mingw-w64-x86_64-toolchain
$ pacman -S mingw-w64-x86_64-gnutls
$ pacman -S mingw-w64-x86_64-ruby
```

その他ツールをインストール

ucrt64.exe を実行（以降、この環境で作業していく）

```
$ pacman -S nano        #（簡易エディター）
$ pacman -S make        #（make をインストール ※必須※）
$ pacman -S openssh     #（openssh をインストール）
$ pacman -S git         #（Git をインストール）
$ pacman -S ruby        #（Ruby をインストール）
$ pacman -S ruby-docs   #（Ruby documentation をインストール）
$ pacman -S p7zip       #（7z をインストール）
```

旧MINGW64

```
$ pacman -S mingw-w64-x86_64-ag  #(ag 高速検索コマンドをインストール)
```

Git 初期設定

Gitを最初に使い始める前には、最低限の情報登録（ユーザ名、メールアドレス）が必要。

「Gitの初期設定」を参照

実行用のショートカットを作る

エクスプローラからc:¥msys64¥mingw64.exeを右クリックして、送る→デスクトップ　でデスクトップに実行用のショートカットアイコンを作っておくと便利。

日常のアップデート

```
$ pacman -Syu
```

- シェルの更新があった場合は、ウィンドウを強制終了を要求される場合があるが、終了させて再度コマンドを実行する。

msys2のディレクトリ表現

msys2では、/はWindows上ではmsys2がインストールされたフォルダになる。すなわち64bit版の場合はC:¥msys64、32bit版の場合はC:¥msys32を示す。

又Windows上のC:¥のトップは、msys2では/C/又はC:/になる。すなわちWindowsのログイン・ユーザuserのデフォルト・ダウンロードフォルダは、C:¥User¥user¥Downloadなのでmsys2上では/C/User/user/Downloadとなる。

ホームディレクトリ

msys2のホームディレクトリは、/home/<ユーザ名>になるWindows上では、C:¥msys64¥home¥<ユーザ名>となる。

環境を整える

msys2のターミナルが起動する時に、ホームディレクトリの.bashrcシェルスクリプトが実行される。このファイルは、デフォルトで作られているが、下記のエイリアス定義をコメントアウトして有効にしておくと便利。

.bashrc

```
alias ll='ls -l'                # long list
alias la='ls -A'                # all but . and ..
alias l='ls -CF'                #
```

pacmanの基本的な使い方

msys2では、pacmanというパッケージ管理コマンドが利用できる。基本的なコマンドを下記に示す。

1. pacman -h でヘルプ。 2. pacman -S -h で-Sオプションのヘルプ。 3. pacman -Su でパッケージのアップデート(事前にupdate-coreが必要)。 4. pacman -Ss [pattern] でパッケージを検索。 5. pacman -Sl | grep [pattern] でもだいたい分かる。 6. pacman -S [package] でインストール。 7. pacman -Rs [package] でアンインストール。 8. pacman -Sg でパッケージグループの一覧を表示。 9. pacman -Qqe でインストール済みのパッケージ一覧を表示。 10. pacman -Si [package] で詳しい情報を表示。 11. pacman -Sc で使っていないキャッシュを削除。

補足

- 従来バージョンでは、初回アップデートの最初にupdate-coreコマンド実行が必要だったが、最近のは不要。

32bit環境(mingw32.exe) で使う場合

下記の開発パッケージをインストールする。（基本的には mingw64.exe を推奨）

```
$ pacman -S mingw-w64-i686-toolchain
$ pacman -S mingw-w64-i686-ruby
```

環境変数のHOMEが設定してある

OrCADなどをインストールしている場合、環境変数のHOMEが設定してある。msys2はデフォルトではHOMEを参照して利用しており衝突するので一時的に変更する必要がある。変更方法として下記の方法がある。

- インストール先の「 `mingw64.ini` 」又は「 `mingw32.ini`」に「 `HOME=/home/<ユーザ名>`」を追記する。

関連記事

- msys2でWindows上でのビルド環境をつくる（msys2のインストール）
- （Windows/msys2）Emacs 26.2の64bit版をインストール（pacmanパッケージから）
- （msys2）（lha）コマンドを使う（.lzh書庫の解凍・圧縮）

参考

1. MSYS2 installer 2. Pacman | Wikipedia 3. pacman | archlinux 4. Arch Linux | wikipedia

[1)]
UCRTとは: Microsoft Visual C++でも使われる。Windows 10以降に標準搭載された新しいCランタイム

From:
https://yokoweb.net/dokuwiki/ - **新石器Wiki**

Permanent link:
https://yokoweb.net/dokuwiki/software/msys2/msys2-install?rev=1780273388

Last update: **2026/06/01 09:23**