Windows設定

# Ctrl⇔CapsLock

| Windows Registry Editor Version 5.00  ;英数(CapsLock)→左Ctrl 3A→1D  ;左Ctrl→英数(CapsLock) 1D→3A  ;無変換→全角/半角(漢字) 7B→29  [HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Control\Keyboard Layout]  "Scancode Map"=hex:00,00,00,00,00,00,00,00,\  04,00,00,00,\  1d,00,3a,00,\  3a,00,1d,00,\  29,00,7b,00,\  00,00,00,00 |
| --- |

# \_vimrc

| syntax enable  colorscheme darkblue  set showmode  set showmatch  set shiftwidth=4  set tabstop=4  set expandtab  " set encoding=cp932  set directory=c:\temp  set undodir=c:\temp  set nobackup  set fileformats=unix,dos  set fileformat=dos  set ignorecase  set belloff=all  " CTRL-X are Cut  vnoremap <C-X> "+x  " CTRL-C are Copy  vnoremap <C-C> "+y  " CTRL-V are Paste  map <C-V> "+gP  set guifont=MS\_Gothic:h12 |
| --- |

# context menu

| [["-1","YouTube","http://www.youtube.com/results?search\_query=TESTSEARCH",true],["-1","DMM.R18","http://www.dmm.co.jp/search/=/searchstr=TESTSEARCH",true],["-1","MGS","https://www.mgstage.com/search/search.php?search\_word=TESTSEARCH",true],["-1","FC2","https://adult.contents.fc2.com/search/?q=TESTSEARCH",true],["-1","翻訳","https://translate.google.co.jp/?hl=ja&authuser=0#en/ja/TESTSEARCH",true],["-1","Amazon","https://www.amazon.co.jp/s?k=%s",true],["-1","Sukebei","https://sukebei.nyaa.si/?q=TESTSEARCH",true],["-1","Wolframlpha","https://www.wolframalpha.com/input/?i=TESTSEARCH",true],["-1","IMDB","http://www.imdb.com/find?s=all&q=TESTSEARCH",true],["-1","Bing","http://www.bing.com/search?q=TESTSEARCH",true],["-1","Bing Images","http://www.bing.com/images/search?q=TESTSEARCH",true],["-1","Wikipedia","http://en.wikipedia.org/wiki/Special:Search?search=TESTSEARCH&go=Go",true],["-1","","",true]] |
| --- |

# Gitのbareレポジトリ作成方法

EclipseでWIndow→Show ViewでGit repositoriesを開く。

ディスクに+の付いたアイコンをクリックしてリポジトリを作成する。

このとき「create as bare repository」をチェックする。

# Homeエディションにgpeditをインストール

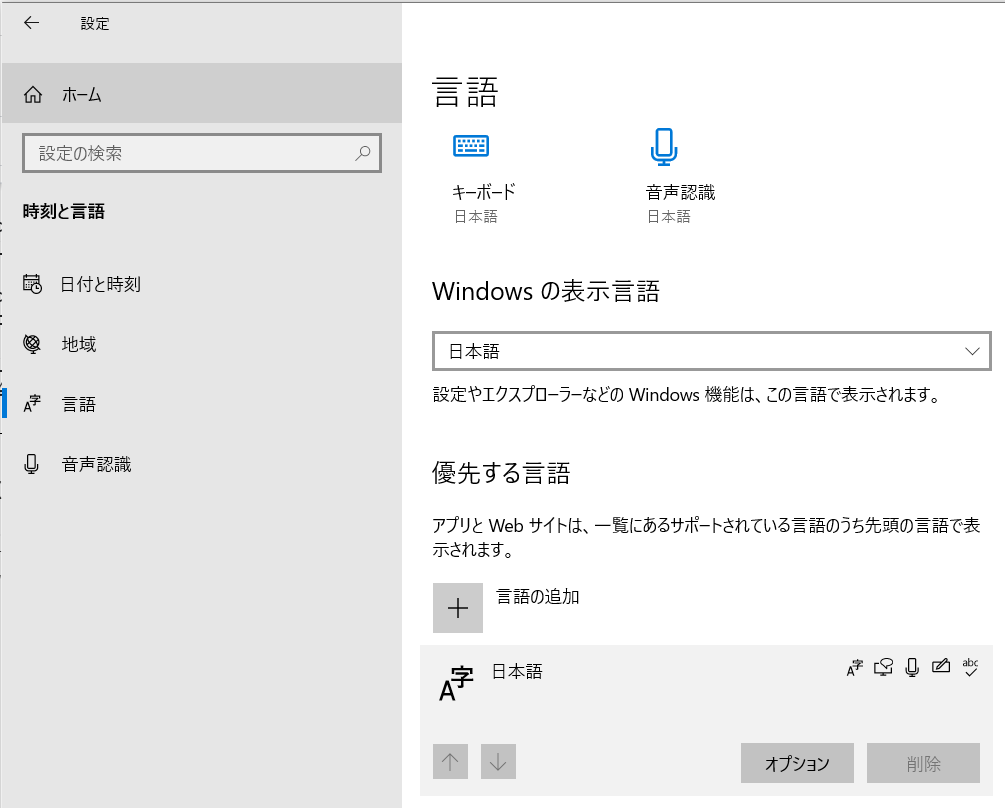
以下のバッチファイルを作成して実行する。

| pushd "%~dp0"  dir /b %SystemRoot%\servicing\Packages\Microsoft-Windows-GroupPolicy-ClientExtensions-Package~3\*.mum >List.txt  dir /b %SystemRoot%\servicing\Packages\Microsoft-Windows-GroupPolicy-ClientTools-Package~3\*.mum >>List.txt  for /f %%i in ('findstr /i . List.txt 2^>nul') do dism /online /norestart /add-package:"%SystemRoot%\servicing\Packages\%%i"  pause |
| --- |

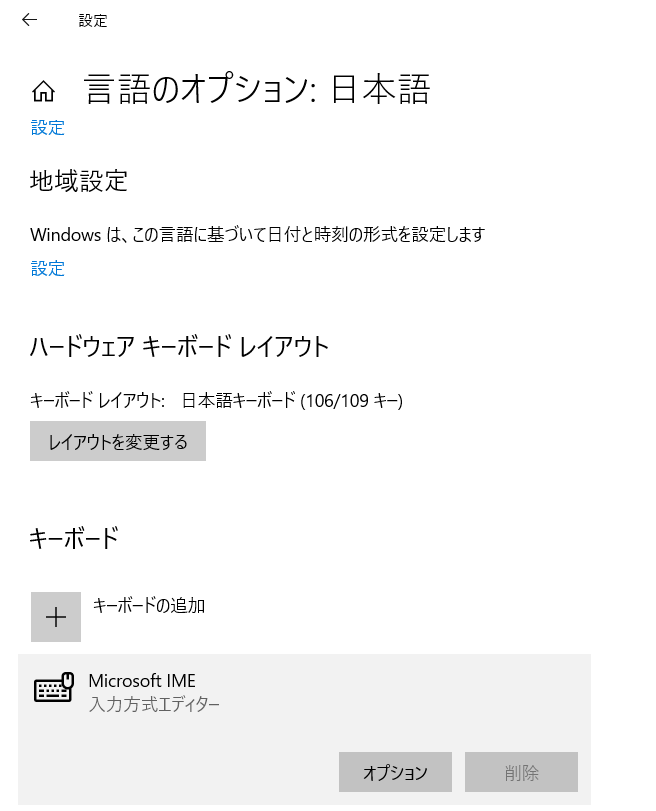
# IMEの切り替えをCtrl+Spaceで行う

設定から「時刻と言語」を選択する。

左のメニューから「言語」を選択する。

「優先する言語」で「日本語」をクリックする。  
出てきた「オプション」をクリックする。

「キーボード」の「Microsoft IME」をクリックする。



出てきた「オプション」をクリックする。



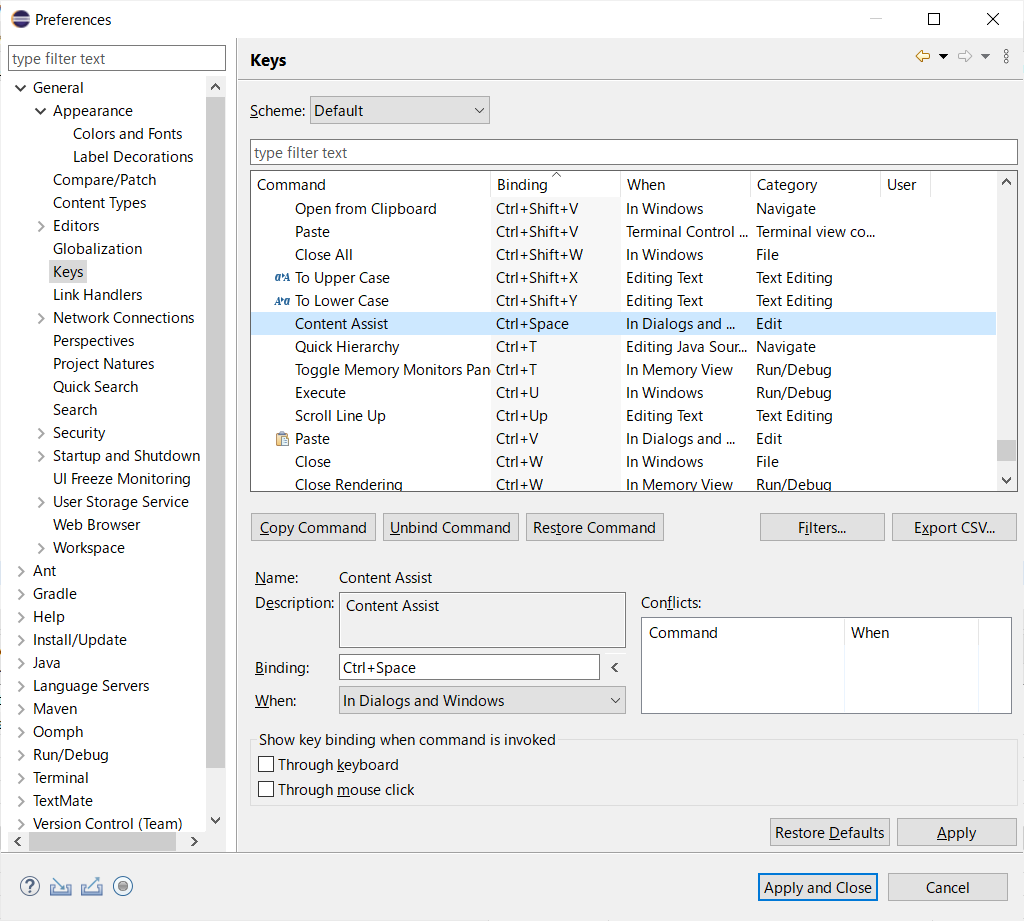
「キーとタッチのカスタマイズ」をクリックする。



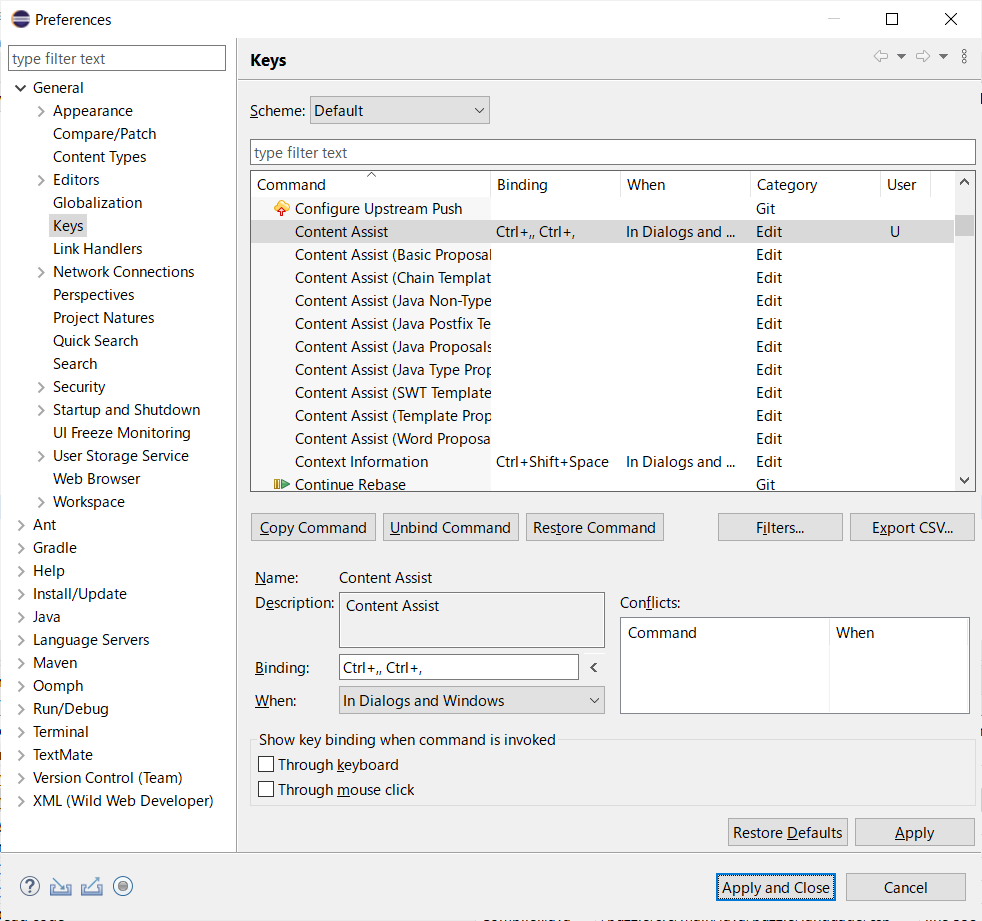
「キーの割り当て」で「Ctrl + Space」に「IME-オン/オフ」を割り当てる。

# eclipseでContent AssistにCtrl+,(カンマ）二度打ちを割当てる

IMEの切り替えにCtrl+Spaceを割り当てるとeclipseでContent Assistのキーバインド(Ctrl+Space)が使えなくなる。これはPreferenceで「General → Keys」から変更できる。



「Content Assist」を選択して、「Binding」にカーソルを移動しCtrl+,を二度打ちする。



# eclipseのvrapperでEscape時にIMEを自動でオフにする

ホームディレクトリ(C:\Users\(ユーザ名))に.vrapperrcを作成する。

「set imd」の行を追加する。

| **set imd**  set expandtab  set shiftwidth=4  set ignorecase |
| --- |

# gvimでファイルの暗号化

編集中のファイルを暗号化する場合は「:X」（コロン・大文字X）を入力してパスワードを設定する。

普通に保存するとファイル上で暗号化される。

暗号化されたファイルをgvimで開くとパスワードを要求されるので、パスワードを入力すると複合化される。編集後通常に保存すると暗号化された状態でファイルが更新される。

編集中に:Xを入力するとパスワードを変更できる。

:Xを入力して新しいパスワードを空文字列にすると暗号化が解除される。

# vimの正規表現

「+」(一回以上の繰り返し)はエスケープが必要。グルーピングの括弧もそれぞれエスケープが必要。

[vim正規表現リファレンス - Qiita](https://qiita.com/kawaz/items/d0708a4ab08e572f38f3)

* a\* は0回以上の a の繰り返しにマッチする。（最長一致）
* a+ は単に a+ という文字列にマッチする。
* a\+ は1回以上の a の繰り返しにマッチする。（最長一致）
* a? は単に a? という文字列にマッチする。
* a\? は0または1個の a にマッチする。
* a\= は0または1個の a にマッチする。（\? と基本同じだが \? は後方検索で使えない）
* a\{n,m} 系（『pregのa{n,m}系』と基本同じ、どれも最長一致）
  + a\{n} は丁度 n 回の a の繰り返しにマッチする。
  + a\{n,} は n回以上の a の繰り返しにマッチする。（最長マッチ）
  + a\{1,} は 1 回以上 a の繰り返しにマッチする。（最長マッチ、a\+ と同じ）
  + a\{n,m} は n回以上 m 回以下の a の繰り返しにマッチする。（最長マッチ）
  + a\{,m} は 0 回以上 m 回以下の a の繰り返しにマッチする。（最長マッチ）
  + a\{,} は 0 回以上 a の繰り返しにマッチする。（最長マッチ、a\* と同じ）
  + a\{} は 0 回以上 a の繰り返しにマッチする。（最長マッチ、a\* と同じ）
* a\{-n,m} 系（a\{n,m} と基本同じだがブレス内の頭に - をつけることで最短一致になる）
  + a\{-n} は丁度 n 回の a の繰り返しにマッチする。（-のアリナシで意味に違いはない）
  + a\{-n,} は n 回以上の a の繰り返しにマッチする。（最短マッチ、『pregのa{n,}?と同じ』））
  + a\{-1,} は 1 回以上 a の繰り返しにマッチする。（最短マッチ、『pregのa+?』と同じ）
  + a\{-n,m} は n 回以上 m 回以下の a の繰り返しにマッチする。（最短マッチ、『pregのa{n,m}?と同じ』）
  + a\{-,m} は 0 回以上 m 回以下の a の繰り返しにマッチする。（最短マッチ、『pregのa{,m}?と同じ』）
  + a\{-,1} は0か1個の a にマッチする。（最短マッチ、『pregのa??』と同じ）
  + a\{-,} は 0 回以上 a の繰り返しにマッチする。（最短マッチ、『pregのa\*?』と同じ）
  + a\{-} は 0 回以上 a の繰り返しにマッチする。（最短マッチ、『pregのa\*?』と同じ）
* \@> …強欲な量指定子。直前のパターンが強欲にマッチするようになる。（『(Perlの(?>pattern)と似ている）
  + \(a\*\)\@>b は aaab にマッチするが、\(a\*\)\@>ab は aaab に決してマッチしない。なぜなら \(a\*)\@> が先に aaa まで取ってしまいバックトラックもしないので次の文字は b となり、後が ab になることはないため。