

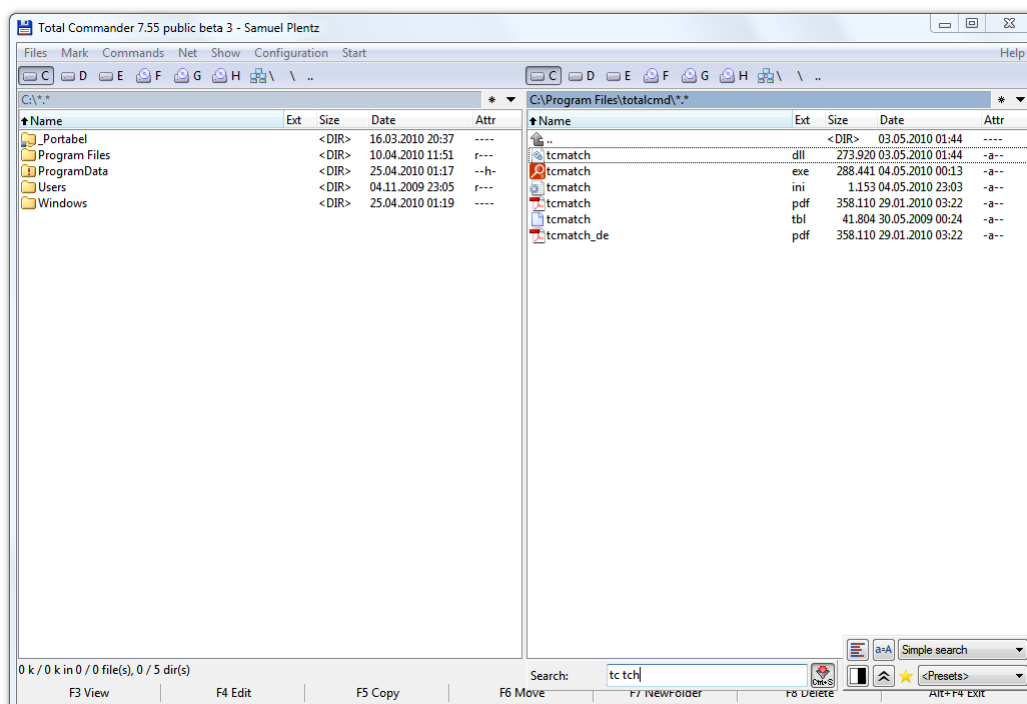
# QuickSearch eXtended Total Commander 7.5+ számára

Ingyenes (freeware) szoftver, készítette Samuel Plentz - [www.familie-plentz.de](http://www.familie-plentz.de)

1. Telepítés
2. Képességek
3. Testreszabás
4. Korlátozások
5. Verziótörténet

## 1. Telepítés

- Töltsd le a következő fájlt: [“QuickSearch eXtended.zip”](#)
- A plugin automatikusan telepítésre kerül, amint megnyitod a zip fájlt Total Commanderen belül, majd az “Igen”-re kattintasz a felugró ablakban. A manuális telepítéshez mindössze annyit kell tenned, hogy kicsomagolod az archívumot a Total Commander mappájába.
- A “QuickSearch eXtended” aktiválásához **újra kell indítanod** a Total Commandert.
- A “QuickSearch eXtended” használatához nyomd meg a **Ctrl+S** billentyűkombinációt, majd gépeld be a keresendő szöveget.
- Ha uninstallálni szeretnéd a “QuickSearch eXtended”-et, mindössze annyi a teendő, hogy eltávolítod a “tcmatch\*.\*” fájlokat a Total Commander mappájából.



Így néz ki a “QuickSearch eXtended” telepítés után.  
(a képen alul, jobb oldalt látható)

## 2. Képességek

Az eszköz legfőbb jellemzője, hogy a keresőszót a szóközöknél szövegrészletekre bontja. Egy fájl akkor egyezik a keresőszóval, ha az összes szövegrészlet megtalálható benne. A “|” elválasztó karakter használatával az elválasztótól balra és jobbra található szavak egyaránt keresésre kerülnek. Ha szeretnéd, elhelyezheted a “!” karaktert a keresőszó elején a találatok listájának invertálásához.

### Egyszerű keresés:

Keresés:	“macska kutya”
Találatok:	Egy <b>kutya</b> és egy <b>macska</b> elköltözött Kínába.txt <b>MaCSkA</b> félék és egy <b>kiskUtYa</b> .avi <b>macskajancsi</b> <b>kutyas</b> zorítóban.doc
Keresés:	“macska zene kutya vicc”
Találatok:	<b>Zene</b> bonáló <b>macskaféle</b> .pdf <b>vicces</b> <b>Kutya</b> .avi
Keresés:	“!.pdf”
Találat:	macska.bmp
Nem találat:	kutya.pdf

### RegEx keresés:

Keresés:	“?exe _\d”
Találatok:	tcmd_75b4.exe AutoHotkey_104800_Install.exe IZArc_4b1.exe

### Hasonlóság alapú keresés: ([Levenshtein-távolság](#) használatával)

Keresés:	“<szín saját”
Találatok:	<b>szín</b> kavalkád.bmp <b>saját</b> kedvenc <b>Szín</b> .jpg
Keresés:	“<2lefenstein”
Találat:	<b>Levenshtein</b> .cpp

### Krs keresés: (a Quick Silver használja Mac alatt)

Keresés:	“*ftl”
Find:	<b>Foto</b> album.zip <b>fatál</b> .png <b>Ferit</b> elhívták kaszálni.docx

Alapértelmezésként az “Egyszerű keresés”-t nem szükséges aktiválni. Csak gépeld be a keresőszavad, és máris működésbe lép. Ha a “RegEx keresés”-t szeretnéd alkalmazni, írd “?” karaktert egy reguláris kifejezés elé. A “Hasonlóság alapú keresés” használatához gépelj egy “<” karaktert a keresendő szöveg elé. Meghatározhatod a szavanként engedélyezett különbözőségek számát is, ha hozzáfűzöl egy számot. Például: “<2synchronise”. A “Krs keresés” használatához a keresőszó elé helyezz egy “\*” karaktert.

## PinYin keresés:

Keresés:	"ext sz zgrand"
Találat:	中国randomtext 神舟.bmp

Az egyszerű keresés, a hasonlóság alapú keresés és a Krs keresés használatakor teljes értékű PinYin támogatás áll a felhasználó rendelkezésére. Ez annyit jelent, hogy egy Kínai karakter egyenértékű az annak megfelelő PinYin karakterrel. "RegEx keresés" használatakor minden egyes szövegrészlet PinYin stringként kerül keresésre, PinYin karakterek használata azonban nem támogatott reguláris kifejezéseken belül. Amennyiben a rendszer nyelve Kínai, a PinYin a szoftver első használatakor aktiválásra kerül. Egyébként külön, manuálisan kell aktiválni, elsősorban teljesítménybeli okok miatt.

## Koreai vezető szótag keresés:

Keresés:	"ㅏ ㅑ 야퍼 ㅓ rand"
Találat:	정 법randomtext 평양양평.bmp


Teljes értékű vezető szótag keresés-támogatás áll a felhasználó rendelkezésére egyszerű kereséskor, hasonlóság alapú kereséskor és Krs kereséskor. Ez azt jelenti, hogy egy Koreai karakter egyenértékű az annak megfelelő vezető szótaggal. "RegEx keresés" használatakor minden egyes szövegrészlet keresésre kerül az annak megfelelő vezető szótagként is, vezető szótagok használata viszont nem támogatott reguláris kifejezéseken belül. Ha a rendszer nyelve Koreai, a vezető szótag keresés automatikusan aktiválásra kerül a szoftver első használatakor. Egyébként manuálisan kell aktiválni, elsősorban teljesítménybeli okok miatt.

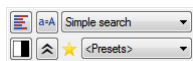
## Sablonok:

Elmentve:	".pdf .doc .ppt .xls .txt>1"
Előhívás:	">1"

Hozzáadhatsz egy keresőszót a sablonhoz, ha a ">[char]"-t hozzáfűzöd (ahol "[char]" egy tetszőlegesen megadott karakter). A későbbiekben mindössze a ">[char]"-ra keresésre lesz szükséged a sablon betöltéséhez.

## Kiegészítő felhasználói kezelőfelület

A kiegészítő felhasználói felülettel fontos beállításokat módosíthatsz, kiválaszthatod az aktív keresést, és sablonokat használhatsz. Ha további, részletesebb módosításokat szeretnél eszközölni a szoftveren belül, kattints a következő gombra: 



Amennyiben korábban kikapcsoltad a kiegészítő felhasználói felületet, a "tcmatch.exe" (vagy a "tcmatch64.exe") elindításával szintén elérheted a részletes testreszabásához szükséges opciókat.

## Tartalom beépülők: (\*.wdx)

A fájlnevek mellett a Total Commander által használt tartalom beépülők adataiban is kereshetsz. Mindössze annyit kell tenned, hogy meghatározol egy-egy wdx csoportot minden egyes, általad használni kívánt tartalom beépülő számára. Az első wdx csoportban lévő összes mező tartalma magával a fájlnévvel kerül szűrésre. Amennyiben hozzá szeretnél fűzni további wdx csoportok mezőéhez is, használhatod a wdx elválasztó karaktert: “/”.

A következő példa az első wdx csoportban egy tartalom beépülőt, és annak “Szerző” mezőjét használja, míg a második wdx csoportban egy másik tartalom beépülőt, és annak “Leírás” mezőjét:

Keresés:	“doc Samuel/Total”		
Találatok:	tcmatch.doc	Szerző: Samuel	Leírás: ...total...
	Samuel.docx	Szerző: Peter	Leírás: ...subtotal...
	Samuel.xls	Szerző: Doctor K.	Leírás: ...totals...

32 bites Total Commander használatakor kizárólag “\*.wdx” beépülőket vehetsz igénybe, míg a Total Commander 64 bites verziója alatt egyedül a “\*.wdx64” kiterjesztésű tartalom beépülők engedélyezettek. Ezek a beállítások a megfelelő kiegészítő felhasználói felületben szabhatók testre. (“tcmatch.exe”, vagy “tcmatch64.exe” – rendszertől függően)

## Helyettesítési szabályok

A felhasználói felületen belül elérhető karakter helyettesítési szabályok mellett lehetőség van szavak cseréjére is, méghozzá szótár (dictionary) használatával. Ha létezik “tcmatch.dic” elnevezésű fájl (Formátum: UTF16-LE) akkor feldolgozásra kerül, érvényes bejegyzések reményében. Egy fájlban belüli érvényes bejegyzés egy sor, mely egy “TAB” (tabulátor / “tab stop”) karakterrel elválasztott kulcsot és értéket tartalmaz. A fájl nem tartalmazhatja ugyanazt a kulcsot kétszer.

Amikor a keresés folyamatban van, a fájlneveben / szűrő szóban található bármely kulcs összes előfordulása helyettesítésre kerül az annak megfelelő értékkel. Amikor több helyettesítés is elérhető, a hosszabb kulcsok helyettesítése a magasabb prioritású.

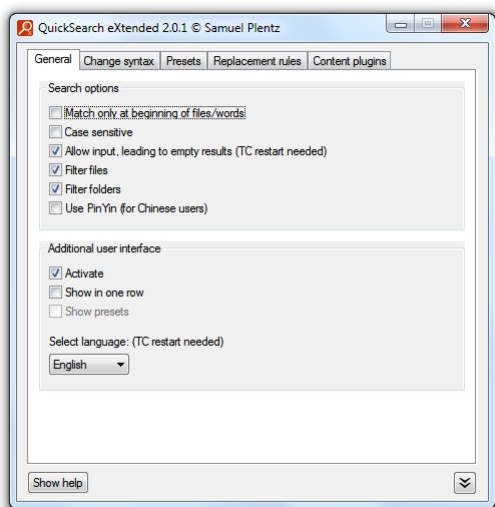
### Hátrányok:

- A “tcmatch.dic” fájl módosításai a Total Commander újraindítását követően lépnek életbe.
- A szöveg már helyettesített részei nem kerülnek további helyettesítések alá, mivel ez végtelen ciklusokhoz vezethetne.
- Bármely kulcs, mely több mint egy karakterből áll nem lesz a találatok között, ha a kulcs első részére (“leading part”) keresünk. Példának okáért vegyük a “kya” kulcsot, és a “きゃ” értéket. Amennyiben a fájl neve “kya.txt”, a keresőszó pedig mindössze “ky”, a “kya.txt” nem lesz a találatok között. Az új fájlnev “きゃ.txt” lenne, de az új keresőszó még nem “きゃ”, hanem még mindig “ky”.

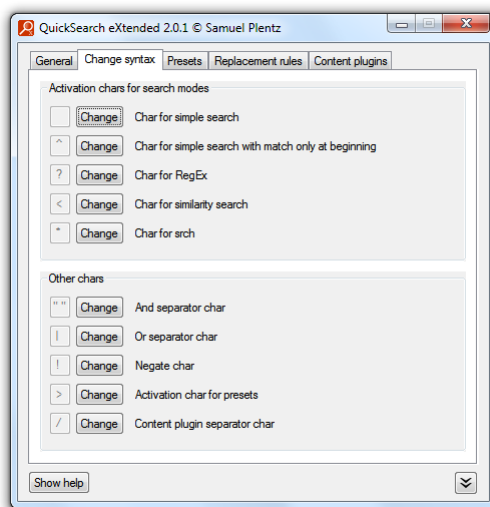
Bármilyen javaslatot szívesen fogadok.

### 3. Testreszabás

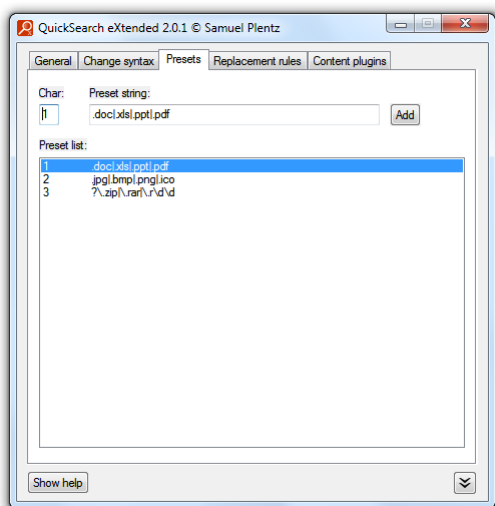
A "QuickSearch eXtended" az első indításkor létrehozza a "tcmatch.ini" fájlt ugyanabban a mappában, ahol a "tcmatch.dll" is megtalálható. Ha ez a hely írásvédett, a fájl a "wincmd.ini"-t tartalmazó mappában lesz létrehozva. Könnyedén módosíthatod a beállításokat a felhasználói kezelőfelület segítségével:



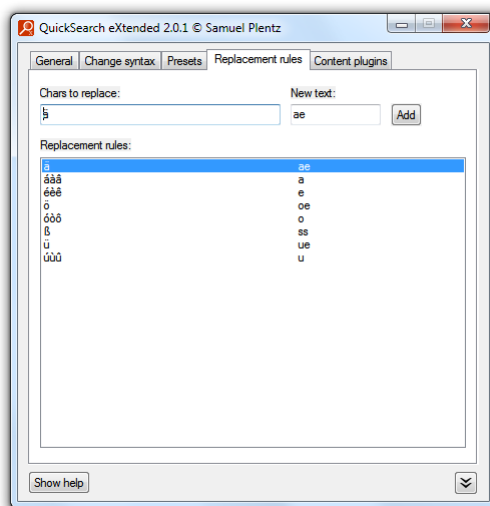
Az alapvető opciókon túl, aktiválhatod a kiegészítő felhasználói felületet, és testreszabhatod a kinézetét.



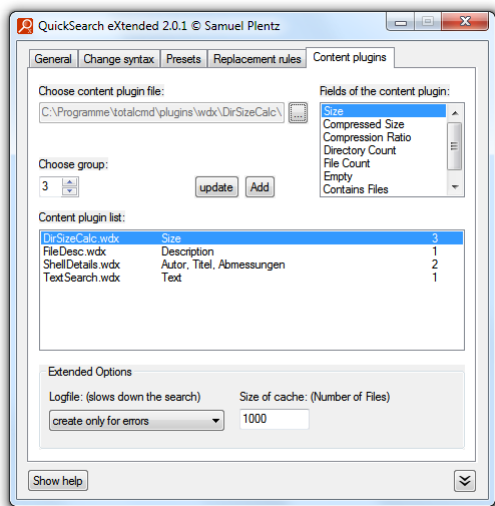
Eldöntheted, mely karakterek legyenek a különböző keresési típusok aktiválói. Az lesz az alapértelmezett keresési típus, amelynél nem adtál meg aktiváló karaktert. Testreszabhatod a kereső által használt további karaktereket is.



Létrehozatsz sablonokat, melyeket később elérsz a kiegészítő felhasználói felületből, vagy akár ">[char]" begépelésével is.



Készíthetsz különböző helyettesítési szabályokat. Amikor a keresés folyamatban van, egy karakter szövegbeli összes előfordulása helyettesítésre kerül az új szöveggel.



Itt szabhatod testre a tartalom beépülőket. A gyorsítótár értékének módosításával megadhatod, hogy a tartalom beépülő értékei hány fájlra legyenek gyorsítótárazva (cache-elve). Használd a naplófájlt, ha úgy véled, nem működik a tartalom beépülő adatai alapján történő keresés. A "tcmatch.log" fájl ugyanabban a mappában lesz létrehozva, ahol a "tcmatch.ini" is.

## 4. Korlátozások

- A “Total Commander” > “Általános beállítások” > “Gyorskeresés” beállításai nem kerülnek figyelembevételre.  
(Pontos kezdet és vég)
- RegEx: a “kis-/nagybetű érzéketlenség” nem megfelelően működik bizonyos nyelvek egyes karaktereinek használatakor.
- Tartalom beépülők: A “ContentGetDetectString” és a “Units” nem kerülnek figyelembevételre.
- Tartalom beépülők: Lefordított Mezőnevek nem használhatóak.
- Tartalom beépülők & fájlok/mappák szűrése: Hálózati meghajtók, FTP és archívumok nem támogatottak.

## 5. Verziótörténet

### Version 2.2.1

- Added Spanish translation for the user interface - thank you Victor Emmanuel

### Version 2.2.0

- Added support for Korean lead syllable search
- Added support for dictionaries (tcmatch.dic) - see help file for details
- Added example dictionary for Japanese
- Added Korean readme - thank you sheppaul

### Version 2.1.2

- Fixed: Pinyin search was broken in x64 version

### Version 2.1.1

- Fixed: additionally user interface is always shown at the right position now

### Version 2.1.0

- Added support for the new 64-bit Total Commander. (Total Commander (x64) 8.0ß2 and later)
- Also added support for the new 64-bit content plugins: "\*.wdx64"

### Version 2.0.1

- Reworked the readme (formulations (especially for content plugins), translated forgotten text to German, created German images)
- Added customization of content plugins to the user interface
- Added language dropdown to user interface
- Added: the additional user interface may now hide the preset dropdown
- Added tooltips to the additional user interface
- Fixed: It's now possible to store "tcmatch.ini" in "UTF16 LE" instead of ANSI (thanks to AHK\_L)
- Fixed: the additional user interface closes when TC is minimized to tray
- Fixed: the user interface is closed if TC is not running

### Version 2.0.0

- Added an additional user interface, to choose the search mode, change most important search options and to use presets
- Added an user interface to customize all options beside the content plugin options. (they still need to be changed in the ini)
- Improved the speed, when content plugins are activated but not used. (thanks Hacker)
- Added ini option "wdx\_cache" to allow caching of content plugin searches.
- Reworked the readme.
- Added German readme and language for user interface.

### Version 1.6.0

- Added a new search-mode: "srch" (default activation char: "\*") With this search you may skip letters of the filename. So the filter "dwn" will find the files "Download.zip" and "Sandra went to town.docx".

### Version 1.5.3

- Added a new [replace] section to the ini where it's possible to replace chars by a string. Could be used to replace diacritics by any other char or string. See "readme > 3. Customize > Replace section" for details and examples.
- Fixed a bug when converting ANSI wdx output to Unicode. Diacritics were not translated properly.



#### Version 1.5.2

- Greatly improved the performance by around 80%-90%! It's just like a lightning now in a folder with 10.000 files.
- Added uninstall information's in the readme.

#### Version 1.5.1

- Important change: The name "Google like Quicksearch" changed to "QuickSearch eXtended".
- Clean up the first post, so that the user will more likely read the readme. ;-) (Full history is now only available in readme.)
- Change: Default value of "wdx\_separator\_char" is now "/" instead of "\". The backslash prevented to use the RegEx search.
- Fixed: Filtering files in FTP mode is possible again. (broken in Version 1.5)
- Disabled wdx and "filter\_files\_and\_folders" for network drives, FTP and inside of zip files.
- Some readme cleanup and redesign.

#### Version 1.5+ (silent update)

- Fixed: Filtering files in FTP mode is possible again.

#### Version 1.5

- Added: New activate char "simple\_search\_match\_beginning\_activate\_char=^" available in ini.
- Added: Changes in the ini have immediately effect. Restart TC only necessary for: "allow\_empty\_result"
- Fixed: The option "case\_sensitive" was ignored for Russian language. (and probably other languages too) Didn't fix it for the RegEx library.
- Improved wdx support:
  - Added: Groups for wdx plugins. Create them by adding a "wdx\_group=..." entry to the ini. (see readme for syntax) Use groups with the new "wdx\_separator\_char=\".
  - Dirsizecalc seems to work.
  - Unicode filenames are converted to 8.3 when calling plugins without Unicode support.
  - "ft\_fulltext" should work now without size limit. (not tested)
  - Changed: Ini option "debug\_output" has now 4 states: 0=no output, 1=output wdx errors (default), 2=additional information about wdx values, 3=output all wdx function calls
  - For performance reasons wdx content is only taken from the file if the file is not already included because of the filename.

#### Version 1.4.2

- Change for ini option "match\_beginning": If there is a space before the first written substring the substring can match anywhere in the filename.

#### Version 1.4.1

- Added: New ini parameter "negate\_char=!" to invert the search. The char must be written in front of the whole search string to invert the search.
- Changed: The ini is now searched at path of "tcmatch.dll" first. If not found it is searched in path of "wincmd.ini". If both no ini does exist at all it is tried to create the ini in the same order.
- Changed: The default value of "allow\_empty\_result" is now 1. (Not changed for existing inis)
- Change for ini option "match\_beginning": Just the first written substring must match the beginning of the filename.
- Improved readme file

#### Version 1.4

- Added: Limited wdx plugin support. See "3. Customizing" of readme for usage.
  - does not work with real unicode files and nonunicode plugins (The conversion from unicode filename to nonunicode filename is currently done char by char.)

- “ContentGetDetectString” and “Units” are ignored
- only untranslated Fieldnames can be used
- only 8192 chars are extracted for long strings
- DirSizeCalc doesn’t work for an unknown reason
- Added: “match\_beginning” to match only when the substrings match either the beginning of the filename or a part of it which follows a space immediately (just try it)
- Changed: A new entry with standard value is created in ini file if it doesn’t exist or has an invalid value. (through ini files from previous installations)
- Rewrote and cleaned up big parts of code. Look out for inconsistencies and bugs.
- Fixed: Small mismatches (case sensitive and pinyin).
- Due to a small bugfix TC 7.5 public beta 6 should be used.

#### Version 1.3

- Changed: Renamed “separator\_char” to “and\_separator\_char” in ini.
- Added: “or\_separator\_char” to search either one substring or the other one - use “allow\_empty\_result=1” in ini to make them work better
- Added: ability to create and use presets
- Added: ini option “filter\_files\_and\_folders” to only filter files or only filter folders (doesn’t work in zip files)
- Added: Auto detect system language on first run to set PinYin ini option to “use\_pinyin=1” for Chinese
- Did not fix PinYin support again :-D

#### Version 1.2.2

- Fixed PinYin support again ;-)

#### Version 1.2.1

- Fixed PinYin support

#### Version 1.2

- Added (untested) support for PinYin - thanks to Christian Ghisler and Shanny (Because of slightly performance reasons it has to be activated in the ini: “use\_pinyin=1”)
- Added new ini parameter to be able to change the Separator from space to any other Separator. Example “separator\_char=/"
- Added some more comments to source code.

#### Version 1.1

- Added the introduced function “MatchGetSetOptions” from Total Commander 7.5 public beta 5
- Drop support for Total Commander 7.5 public beta 4
- Added: always override internal search
- Added: always remove leading/trailing asterisk
- Removed workarounds needed for Total Commander 7.5 public beta 4
- Added history to the readme
- Added new option to ini: “allow\_empty\_result=0/1”

#### Version 1.0.2

- Added version informations to dll
- Added readme to the package
- Added readme and history to the post in the Total Commander forum

#### Version 1.0.1

- Changed: The “tcmatch.ini” isn’t bundled anymore. It’s created on first run in same folder like “wincmd.ini”.
- Changed: Renamed “google\_search\_activate\_char” to “simple\_search\_activate\_char” in ini.

#### Version 1.0

- Added a similarity search. (Levenshtein)
- Added Google like search for RegEx.
- Added a workaround for open, but not closed parenthesis. (“{}”)
- Added support for an ini file. (activation chars, case sensitivity)
- Fixed absolute paths in source.

#### Version 1.0 pre3

- Fixed RegEx - thx to “j” it works

#### Version 1.0 pre2

- Added buggy RegEx :-D

#### Version 1.0 pre1

- Support for simple search