Introduction

Table des matières

1. Présentation

1.1. Définition

Une expression régulière (ang.*regular expression*, jargon : *regex*, *regexp*) est une formule ou un motif (ang.*pattern*) qui permet de décrire les caractéristiques d'une chaîne de caractères afin de pouvoir la rechercher dans un texte pour la valider, la modifier, l'effacer ou la déplacer. L'adjectif « régulier » est utilisé ici dans le sens de « qui obéit à des règles ».

1.2. Utilisations

1.2.1 En édition

La plupart des éditeurs de textes orientés programmation comme Sublime Text, JetBrains, Dreamweaver ou Notepad++ permettent l'utilisation d'expressions régulières si on prend soin de cocher l'option adéquate.

Figure 1. Dreamweaver

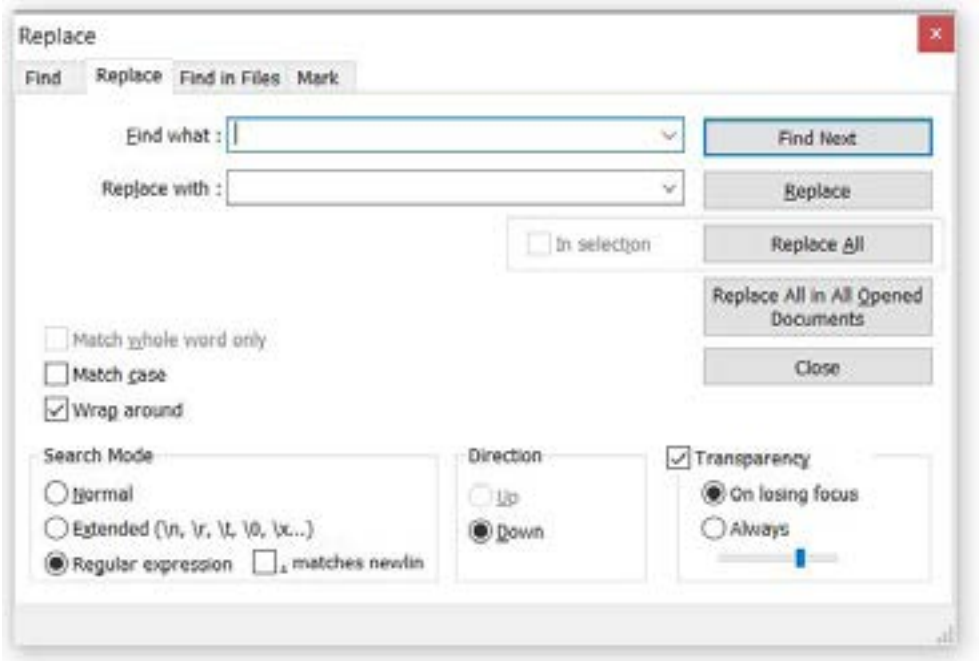
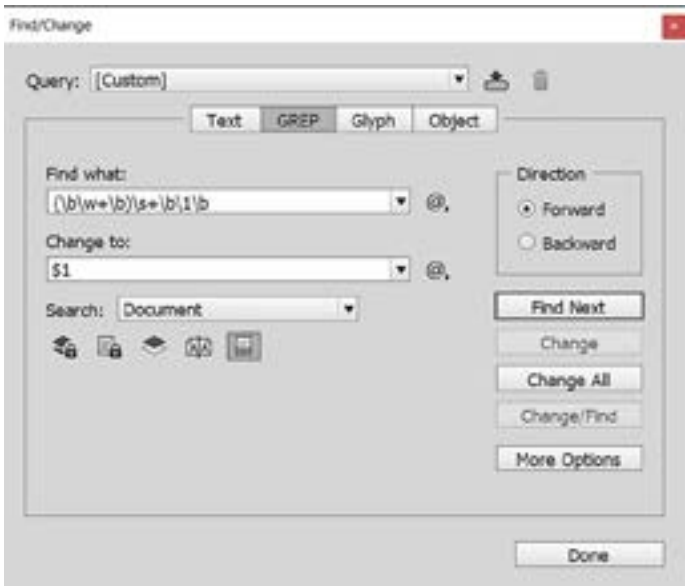


Figure 2. Notepad++



1.2.2 En programmation

2. Les métacaractères

Une expression régulière est formée de caractères littéraux, qui signifient littéralement les caractères qu'ils représentent et de métacaractères, qui prennent une signification spéciale.Commençons par décrire ces métacaractères

2.1. Le point

Le métacaractère point permet de substituer n'importe quel autre caractère.

Ainsi, la chaîne `e.` représente la lettre `e` suivie de `.` , mais l'expression régulière `e.` signifie la lettre `e` suivie de n'importe quel autre caractère.

Exemples `m.n` retrouve `man` et `men` et aussi `mon`, `min`, `mun`, même dans mo`men`t, otto`man`, `mi-`nimum, `mun`ition.

L'expression `m..n` retrouve `main`tenant, `moin`dre, `meun`ier.

2.2. Les quantificateurs

Ils sont placés après un caractère ou un métacaractère ou encore d'autres expressions pour indiquer le nombre d'occurrences nécessaires.

Quantificateur		Signification
`?`	Point d'interrogation	0 ou 1 fois -
`*`	Astérisque	0 ou plusieurs fois
`+`	Signe plus	1 ou plusieurs fois
`{n}`	Un nombre entre accolades	n indique le nombre d'occurrences voulues.
`{m,n}`	Deux nombres séparés par une virgule	entre m et n occurrences

Exemples

L'expression `r.?r` retrouve gue`rre`, te`rre` et o`rdre`.

L'expression `r.*r` retrouve f`ranco-br`itannique mais aussi

mo`rt de froid à la guerre en Mer Noir`e

Pour * et +, l'interpréteur recherche le maximum d'occurrences. L'expression ci-dessus ne retrouve donc pas mo`rt de fr`oid ou gue`rr`e ou Me`r Noir`e

Pour retrouver le moins d'occurrences possible, on ajoute encore un ? à l'astérisque.

Exp. : `r.*?r` retrouve mo`rt de fr`oid ou gue`rr`e ou Me`r Noir`e

Exp. : r`.+?r` ne retrouve pas guerre et terre mais bien ordre ou Mer Noire r.{3}r retrouve exactement 3 occurrences entre les 2 r : ter`ritor`ial, `rappr`ochement

Exp. : `r.{0,3}r` retrouve entre 0 et 3 occurrences entre les 2 r

2.3. Ancres de positionnement

L'accent circonflexe `^` et le dollar `$` peuvent servir à indiquer la position de la chaîne recherchée.

L'accent circonflexe placé au début d'une expression régulière signifie que le texte recherché doit se situer en début de ligne ou de chaîne de caractères (lorsqu'on utilise les expressions régulières en programmation en dehors d'un éditeur)

Exp. : `^Madame` ne retrouve pas `Chère Madame`

Exp. : `^http` vérifie qu'une chaîne ressemble (un peu, au début) à une adresse web

Le dollar placé en fin d'expression régulière indique que le texte recherché doit se situer en fin de ligne ou de chaîne

Exp. : `-$` retrouve les lignes qui se terminent par un tiret

Exp. : `com$` vérifie qu'une chaîne se termine par com

Exp. : `^$` retrouve une ligne vide

Exp. : `^.*$` capture tout le contenu d'une ligne = tous les caractères entre le début et la fin.

2.4. Le backslash

La contre barre oblique (ang. *backslash*) sert de caractère d'échappement : elle donne au caractère qu'elle précède une signification alternative. Ainsi, placée devant un métacaractère, elle signifie que ce caractère reprend sa signification littérale, tandis que, placée devant un caractère normal, elle lui donne une valeur de métacaractère.

2.4.1 Échappement de métacaractères

Par exemple, pour retrouver un point au moyen d'une expression régulière, il faut échapper le point au moyen d'un *backslash*.

`\.`	`.`
`\?`	`?`
`\+`	`+`
`\\`	`\`
`\{`	`{`
`\(`	`(`

2.4.2 Caractères de contrôle

Le *backslash* peut être placé devant certains caractères pour créer des séquences qui représentent des caractères de contrôle (donc des caractères sans glyphe).

`\t`	tabulation
`\r`	retour chariot (CR = Carriage Return
`\n`	saut de ligne (Newline) (LF = Line Feed)
`\f`	saut de page (Form Feed)

2.4.3 Familles de caractères

Avec d'autres caractères littéraux, la séquence permet de représenter des classes de caractères

`\d`	les chiffres (pensez à *digit*)
`\s`	les espaces (y compris les tabulations et les sauts de ligne)
`\w`	les caractères alphanumériques y compris l'underscore
`\D`	Complément de \d tout sauf les chiffres
`\S`	Complément de `\s`
`\W`	Complément de `\w`

Exp. : `\d+\s*\$` indique une suite de 1 ou plusieurs chiffres suivis de 0 ou plusieurs espaces suivis d'un $: `100$` ou `50 $`

2.4.4 Ancres de positionnement

La séquence `\b` indique une limite de mot, au début ou à la fin.

L'expression `ent\b` retrouve tous les mots qui se terminent par ent : présid`ent`, Ori`ent`, mett`ent`

L'expression `\bent` retrouve tous les mots qui commencent par ent : `ent`repreneur, `ent`rez, `ent`relarder

L'expression `\bentre\b` retrouve le mot `entre` mais pas rentre, rentrer ou entrer

La séquence `\B` est son complément, donc à l'intérieur d'un mot

`\Bent\B` retrouve t`ent`ative, sept`ent`rionale

2.5. Les crochets

Ils permettent de créer des ensembles de caractères qui servent de critères de sélection. L'ensemble peut être décrit par une énumération, par un intervalle ou une exclusion de cet intervalle ou une combinaison de toutes ces possibilités et permet de retrouver un seul caractère de cet ensemble.

Exp. : `[aeiou]` énumération de l'ensemble des voyelles.

Exp. : `[àâäéèêëîïôöùüû]` énumération de l'ensemble des caractères accentués.

On peut spécifier un ensemble par un intervalle.

Exp. : `[a-zA-Z]` spécifie l'ensemble des caractères alphabétiques non accentués minuscules et majuscules

Exp. : `[0-9]` spécifie l'ensemble des chiffres.Identique à `\d`

Exp. : `F[123]` retrouve `F1`, `F2`, `F3`

Exp. : `7[34678]7` retrouve tous les modèles de Boeing actuels `737`, `747`, `767`, etc

Exp. : `[123456]0[1234]` retrouve quelques modèles de la gamme Peugeot (même ceux qui n'existent pas encore) : `101`, `102`, `402`, `403`, `604`

Exp. : `Dupon[dt]` retrouve `Dupond` et `Dupont`

Exp. `[A-Z]\.[A-Z]\.[A-Z]\.` retrouve tout sigle de 3 lettres capitales séparées par un point : `P.I.B.`, `O.G.M.`

Pour rechercher un tiret en plus de l'intervalle, il faut le placer en dehors de l'intervalle

Exp. : `[-a-z]`.

Pour rechercher un crochet, il faut l'échapper `\[`, `\]`

Exemples Toute chaîne formée de deux groupes de lettres séparés par un tiret :

Exp. : `[a-zA-Z]+-[  a-zA-Z]+` : `ceux-ci`, `austro-hongrois`, moldo-valaques, Jean-Pierre

Exp. : `[A-Z]{3}[0-9]{3}` Ancienne plaque d'immatriculation belge : `APP740`, `CBB765`

Exp. : `1-[A-Z]{3}-[0-9]{3}` Nouvelle plaque d'immatriculation belge : `1-BDF-456`

Numéro de téléphone belge 0[0-9]{1,2}\s[0-9]{2,3}\s[0-9]{2}\s[0-9]{2}

retrouve 02 123 45 67 ou 081 45 89 96

On peut rendre les espaces facultatifs 0[0-9]{1,2}\s?[0-9]{2,3}\s?[0-9]{2}\s?[0-9]{2}

Retrouve 04 123 4567 et 041234567

2.6. L'accent circonflexe

Au début d'une chaîne dans une paire de crochets, il signifie l'exclusion

Exp. : `[^4-7]` signifie n'importe quel caractère sauf un 4, 5, 6 ou 7

2.7. La barre verticale

Appelée tube ou *pipe* en anglais, elle exprime une alternative. On reconnaît l'opérateur binaire | de JavaScript ou PHP.

Exp. : `[a-z0-9]+\.(gif|png|jpg|jpeg)` retrouve `image05.gif` ou `eifel07.jpeg`.

2.8. Les parenthèses

Elles permettent de regrouper certaines parties d'une expression pour former des sous-expressions. Si par exemple on enlève les parenthèses dans l'expression ci-dessus, on retrouve soit une suite de caractères alphanumériques suivie d'un point suivi de gif soit une des extensions png, jpg, jpeg.

Exp. : `[a-z0-9]+\.gif|png|jpg|jpeg` retrouve `image05.gif` ou `png` ou `jpg` ou `jpeg`.

Exp. : `([A-Z]\.)+` retrouve des sigles de n'importe quelle longueur : un caractère alphabétique suivi d'un point et le tout une ou plusieurs fois.

3. Motifs de substitution

A chaque sous-expression (expression entre parenthèses) correspond un numéro séquentiel à partir de 1. Il est possible de récupérer le résultat d'une sous-expression pour le réutiliser en utilisant le *backslash* suivi du numéro correspondant (dans la zone de recherche) ou un dollar suivi du numéro (dans la zone de remplacement). On crée ainsi un motif de substitution.

Exemple 1 : trouver des doublons

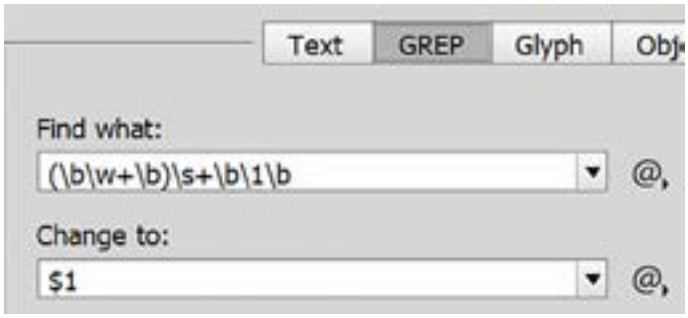
Deux occurrences successives d'un même mot forment un doublon.

Pour retrouver un mot, on utilise l'expression `\b\w+\b` qui recherche une séquence de caractères alphanumériques `w+` en bordure de mot `\b\b` .

Pour retrouver un doublon, il faut écrire `(\b\w+\b)\s+\b\1\b` :

- insérer l'expression entre parenthèses
- prévoir un ou plusieurs espaces `\s+` .
- Le `\1` représente le contenu de la sous-expression

Pour supprimer le doublon, on le remplace par la sous-expression en indiquant `$1` dans le champ Remplacer



Exemple 2 : Transformer des dates

On désire transformer des dates du mois de janvier en remplaçant janvier par-01-.Il faut bien sûr que le mot janvier soit utilisé dans une date, donc précédé de 1 ou 2 chiffres et d'un espace et suivi d'un espace et de 4 chiffres.On recherche ([0-9]{1,2})\s+janvier\s+([0-9$]{4}) On remplace par $1-01-$2

4. Exercices résolus

4.1. Intervertir des données

Un fichier contient une liste de paires {prénom tabulation nom} et on désire inverser l'ordre en {nom tabulation prénom}

On recherche `([a-zïéÿç ]+)\t([a-zïéÿç-]+)`

On remplace par `$2\t$1 (\w+-?\w*) \1`

4.2. Inverser les noms et les prénoms

Alphonse Daudet (1897) Joris-Karl Huysmans (1900-1907) Léon Hennique (1907-1912) Gustave Geffroy (1912-1926)

Construire une sous-expression formée d'une suite de 1 ou plusieurs caractères alphabétiques suivie d'une espace facultative suivie d'une suite de caractères alphabétiques de 0 ou plusieurs caractères. Le tout est suivie d'une virgule, espace puis d'une autre sous-expression contenant un tiret facultatif.Remplacer par l'inversion des deux sous-expressions.

4.3. Supprimer l'espace séparateur de milliers

4.4. Supprimer le mot Drapeau

suivi de la préposition et de l'article ou de l'article contracté Drapeau du Japon Drapeau des États-Unis Drapeau de la France Drapeau de l'Allemagne

4.5. Éliminer les doublons

4.6. Convertir des noms propres en minuscules

\u convertit le caractère suivant en majuscule \l convertit le caractère suivant en minuscule \U convertit toute l'expression en majuscule \L convertit toute l'expression en minuscule