Musterlösung Übung Reguläre Ausdrücke

# Zusammenfassung reguläre Ausdrücke

|  |  |
| --- | --- |
| a | Das Zeichen a |
| . | Ein beliebiges Zeichen |
| [abc] | Ein beliebiges Zeichen aus der Menge {a, b, c} |
| \d | eine Ziffer [0-9] |
| \w | Buchstabe, Ziffer oder Unterstrich – [a-zA-Z\_0-9] |
|  |  |
| X|Y | X oder Y |
| X\*,X\*? | Eine beliebige Wiederholung von X, gierig bzw. genügsam |
| X+, X+? | Mindestens einmal X, gierig bzw. genügsam |
| X? | Höchstens einmal X |
| {n} | der vorangehende Ausdruck muss exakt n mal vorkommen |
| {m,n} | der vorangehende Ausdruck muss mindestens m mal vorkommen und darf höchstens n mal vorkommen |
| (xyz) | Gruppierung: xyz müssen miteinander vorkommen |
| [^xyz] | Nicht x, nicht y, nicht z |
| \X | X ein Sonderzeichen \()[]\*+?{}.|^$- |

# Aufgabe: Datumserkennung anpassen

In einem Programm, das Webseiten auf Daten hin durchsucht, finden Sie folgenden regulären Ausdruck: \d\d\.\d\d\.\d\d

## Beschreiben Sie in Worten, (1) welche Daten dieser Ausdruck findet und (2) welche Daten er nicht findet:

|  |
| --- |
| (1) Er findet nur Daten, bei denen Tage und Monate und Jahre exakt mit zwei Ziffern angegeben sind.  (2) Daten mit einstelligen Tag- oder Monatsangaben findet er nicht, ebensowenig Daten mit vierstelligen Jahresangaben. |

## Geben Sie Beispiele für alle Arten von Daten an, die gemäss folgender Beschreibung zulässig sind:

* sowohl zweistellige als auch vierstellige Jahreszahlen sind zulässig
* sowohl einstellige als auch zweistellige Tag- und Monatsangaben sind zulässig

|  |
| --- |
| Da es für drei Teile eines Datums zwei Möglichkeiten gibt, sind 2\*2\*2 Arten von Daten zulässig:  1.1.10  1.1.2010  1.10.10  1.10.2010  10.1.10  10.1.2010  10.10.10  10.10.2010 |

## Erweitern Sie den Ausdruck \d\d\.\d\d\.\d\d, so dass er Daten gemäss folgender Beschreibung erkennt:

* sowohl zweistellige als auch vierstellige Jahreszahlen sind zulässig
* sowohl einstellige als auch zweistellige Tag- und Monatsangaben sind zulässig

|  |
| --- |
| \d(\d)?\.\d(\d)?\.\d\d(\d\d)? |

# Aufgabe: Postleitzahlen erkennen

Postleitzahlen in der Schweiz sind vierstellig im Bereich von 1000..9999.

## Schreiben Sie einen regulären Ausdruck, der Postleitzahlen erkennt:

|  |
| --- |
| [1-9][0-9]{3} |

# Aufgabe: Zahlen mit Tausender-Apostroph erkennen

Zahlen gibt es in den unterschiedlichsten Formaten. Betrachten wir folgende Zahlen:

\* nur positive ganze Zahlen

\* Tausender-Apostroph zur Abtrennung von 1000 Bereichen

\* zum Beispiel 1, 42, 486, 9’386, 719’528, 783’748’894’846

Auch hier hat jemand von einen Anfang gemacht: Der Ausdruck \d{1,3}'\d{3} erkennt immerhin schon die Zahlen 1’000 bis 999’999.

## Erweitern Sie den Ausdruck, so dass die Zahlen wie eingangs beschrieben erkennt:

|  |
| --- |
| \d{1,3}('\d{3})\* |

# Aufgabe: Mailadressen erkennen

Sie finden in einem Programm folgenden regulären Ausdrück für die Überprüfung, ob eine Eingabe eine gültige Mailadresse darstellt:

([a-z]|[A-Z]|[0-9]|\_|-|\.)+@([a-z]|[A-Z]|[0-9]|\_|-)+\.([a-z]|[A-Z]|[0-9]|\_|-|\.)+

## Beschreiben Sie in Worten, was dieser Ausdruck erkennt:

|  |
| --- |
| Als gültige Zeichen werden Kleinbuchstaben, Grossbuchstaben, Ziffern, Unterstrich, Bindestrich und der Punkt akzeptiert.  Der Ausdruck erwartet mindestens ein gültiges Zeichen vor dem zwingenden Affenschwanz @. Nach dem Affenschwanz muss mindestens ein gültiges Zeichen (ausgenommen der Punkt) vor einem zwingenden Punkt kommen. Nach dem Punkt muss mindestens ein gültiges Zeichen kommen. |

Bei Mailadressen kommt nach dem letzten Punkt eine sog. Toplevel Domain wie .ch oder .com.

## Geben Sie vier unterschiedliche Beispiele von Mailadressen an, die der obige Ausdruck akzeptiert, die aber keine gültige Toplevel Domain enthalten:

|  |
| --- |
| r@r.007  r@r.\_  r@r.-  r@r.... |

# Aufgabe: HTML-Seiten

Webseiten werden textuell beschrieben. Eine einfache Webseite könnte sein:

<h1>Dies ist eine Überschrift ersten Grades</h1>

<h2>Dies ist eine Überschrift zweiten Grades</h2>

<p>Dies ist einfach nur ein ganz normaler Abschnitt. <u>Dieser Teil des Textes ist unterstrichen.</u></p>

<h1>Dies ist noch eine Überschrift ersten Grades</h1>

Die Befehle in spitzen Klammern geben dabei den Browsern an, welche Bedeutung welcher Abschnitt einer Webseite hat.

## Schreiben Sie einen regulären Ausdruck, der Überschriften ersten Grades erkennt.

|  |
| --- |
| <h1>.\*?</h1> |

## Schreiben Sie einen regulären Ausdruck plus einen ersetzenden Ausdruck, der unterstrichene Texte (<u>…</u>) durch kursive Texte (<i>…</i>) ersetzt.

|  |
| --- |
| Suchausdruck: <u>(.\*?)</u>  Ersetzender Ausdruck: <i>$1</i> |