

# Reguläre Ausdrücke REGEX

# Regex

## Definition

Beschreibungssyntax um bestimmte Teile aus Zeichenketten zu prüfen oder zu filtern

Zeichen entsprechen sich selbst

- `"Regex sind cool"` -match `"cool"` `#TRUE`

### Platzhalter

<code>.</code>	Beliebiges Zeichen
<code>\d</code>	Dezimalziffer
<code>\D</code>	keine Dezimalziffer
<code>\w</code>	Buchstabe, Zahl, Unterstrich
<code>\W</code>	kein Buchstabe, Zahl, Unterstrich
<code>\s</code>	Leerzeichen, Tab, Zeilenschaltung
<code>\S</code>	kein Leerzeichen, Tab, Zeilenschaltung

### Quantifizierer

<code>*</code>	Beliebige viele Ausdrücke, inkl. kein
<code>+</code>	1 und mehr Ausdrücke
<code>?</code>	Genau 1 oder kein Ausdruck
<code>{n}</code>	genau n Ausdrücke
<code>{n,}</code>	mindestens n Ausdrücke
<code>{n,m}</code>	mindestens n aber maximal m Ausdrücke

Zeichen entsprechen sich selbst

- "Regex sind cool" -match "cool"

## Beispiele

"Lorem ipsum 1234?" -match ".\*"

"abc1234?" -match "\d+"

"abcd123" -match "\D{4}"

"Lorem ipsum 1234?" -match "\w{6,}"

"Lorem ipsum 1234?" -match "\w{6,}"

"Lorem ipsum 1234?" -match "\s{1,2}"

"Lorem ipsum 1234?" -match "\s{6,9}"

Zeichen entsprechen sich selbst

- "Regex sind cool" -match "cool" #TRUE

## Beispiele

"lorem ipsum 1234?" -match ".\*" # TRUE

"abc1234?" -match "\d+" # TRUE

"abcd123" -match "\D{4}" # TRUE

"lorem ipsum 1234?" -match "\w{6,}" # FALSE

"lorem ipsum 1234?" -match "\w{6,}" # FALSE

"lorem ipsum 1234?" -match "\s{1,2}" # TRUE

"lorem ipsum 1234?" -match "\s{6,9}" # FALSE

Ausgabe der Treffer

`$Matches`

## Beispiele

`"lorem ipsum 1234?"`    `-match ".*"`    # `lorem ipsum 1234?`

`"abc1234?"`    `-match "\d+"`    # `1234`

`"abcd123"`    `-match "\D{4}"`    # `abcd`

`"lorem ipsum 1234?"`    `-match "\w{6,}"`    # `(FALSE)`

`"lorem ipsum 1234?"`    `-match "\w{6,}"`    # `(FALSE)`

`"lorem ipsum    1234?"`    `-match "\s{1,2}"`    # `" "`

`"lorem ipsum    1234?"`    `-match "\s{6,9}"`    # `" "`

### Klassen

- `[abc]` Nur a, b, oder c
- `[a-z]` Kleinbuchstaben
- `[^a-z]` keine Kleinbuchstaben
- `[A-Z]` Großbuchstaben
- `[A-Za-z]` Groß- und Kleinbuchstaben
- `[0-9]` Zahlen
- `[^0-9]` keine Zahlen

### Sonderzeichen maskieren

- `\.` Steht für den Punkt als Zeichen
- `\(\)` Stehen für Klammern als Zeichen
- `\?` Steht für Fragezeichen als Zeichen
- `\+` Steht für Plus als Zeichen
- `\*` Steht für Asterisk als Zeichen
- ...

### Anker

- `^` folgender Ausdruck muss am Anfang stehen
- `$` vorhergehender Ausdruck muss am Ende stehen

# Regex

Zeichenklassen, Maskierung & Anker

## # Matches



# Regex

## Zeichenklassen, Maskierung & Anker

"Glaspuppe", "Trinkgläser", "Glasflasche", "Plastik" -match "gl[aä]s"

# Regex

## Zeichenklassen, Maskierung & Anker

"Glaspuppe", "Trinkgläser", "Glasflasche", "Plastik" -match "gl[aä]s"

# Regex

## Zeichenklassen, Maskierung & Anker

"Glaspuppe", "Trinkgläser", "Glasflasche", "Plastik" -match "gl[aä]s"

# Glaspuppe Trinkgläser Glasflasche

# Regex

## Zeichenklassen, Maskierung & Anker

```
"Glaspuppe", "Trinkgläser", "Glasflasche", "Plastik" -match "gl[aä]s"  
# Glaspuppe Trinkgläser Glasflasche  
"ß", "ä", "s", "t" -match "[^äüöß]"
```

# Regex

## Zeichenklassen, Maskierung & Anker

```
"Glaspuppe", "Trinkgläser", "Glasflasche", "Plastik" -match "gl[aä]s"  
# Glaspuppe Trinkgläser Glasflasche  
"ß", "ä", "s", "t" -match "[^äüöß]" # s t
```

# Regex

## Zeichenklassen, Maskierung & Anker

```
"Glaspuppe", "Trinkgläser", "Glasflasche", "Plastik" -match "gl[aä]s"  
# Glaspuppe Trinkgläser Glasflasche  
"ß", "ä", "s", "t" -match "[^äüöß]" # s t  
"a", "b", "c", "x", "y", "z" -match "[a-m]"
```

# Regex

## Zeichenklassen, Maskierung & Anker

```
"Glaspuppe", "Trinkgläser", "Glasflasche", "Plastik" -match "gl[aä]s"
# Glaspuppe Trinkgläser Glasflasche

"ß", "ä", "s", "t" -match "[^äüöß]" # s t

"a", "b", "c", "x", "y", "z" -match "[a-m]" # a b c
```

# Regex

## Zeichenklassen, Maskierung & Anker

# TRUE vs. FALSE



# Regex

## Zeichenklassen, Maskierung & Anker

```
"Glaspuppe", "Trinkgläser", "Glasflasche", "Plastik" -match "gl[aä]s"
# Glaspuppe Trinkgläser Glasflasche

"ß", "ä", "s", "t" -match "[^äüöß]" # s t

"a", "b", "c", "x", "y", "z" -match "[a-m]" # a b c

"web.de" -match "web\\.de"
```

# Regex

## Zeichenklassen, Maskierung & Anker

```
"Glaspuppe", "Trinkgläser", "Glasflasche", "Plastik" -match "gl[aä]s"
# Glaspuppe Trinkgläser Glasflasche

"ß", "ä", "s", "t" -match "[^äüöß]" # s t

"a", "b", "c", "x", "y", "z" -match "[a-m]" # a b c

"web.de" -match "web\\.de" # TRUE
```

# Regex

## Zeichenklassen, Maskierung & Anker

```
"Glaspuppe", "Trinkgläser", "Glasflasche", "Plastik" -match "gl[aä]s"
                                     # Glaspuppe Trinkgläser Glasflasche

"ß", "ä", "s", "t" -match "[^äüöß]" # s t

"a", "b", "c", "x", "y", "z" -match "[a-m]" # a b c

"web.de" -match "web\\.de" # TRUE

"\windows\temp\test.txt" -match "\\windows\\temp\\test\\.txt"
```

# Regex

## Zeichenklassen, Maskierung & Anker

```
"Glaspuppe", "Trinkgläser", "Glasflasche", "Plastik" -match "gl[aä]s"
                                     # Glaspuppe Trinkgläser Glasflasche

"ß", "ä", "s", "t" -match "[^äüöß]" # s t

"a", "b", "c", "x", "y", "z" -match "[a-m]" # a b c

"web.de" -match "web\\.de" # TRUE

"\\windows\\temp\\test.txt" -match "\\windows\\temp\\test\\.txt" # TRUE
```

# Regex

## Zeichenklassen, Maskierung & Anker

```
"Glaspuppe", "Trinkgläser", "Glasflasche", "Plastik" -match "gl[aä]s"
# Glaspuppe Trinkgläser Glasflasche

"ß", "ä", "s", "t" -match "[^äüöß]" # s t

"a", "b", "c", "x", "y", "z" -match "[a-m]" # a b c

"web.de" -match "web\\.de" # TRUE

"\windows\temp\test.txt" -match "\\windows\\temp\\test\\.txt" # TRUE

"7543" -match "\s{6,9}"
```

# Regex

## Zeichenklassen, Maskierung & Anker

```
"Glaspuppe", "Trinkgläser", "Glasflasche", "Plastik" -match "gl[aä]s"
                                     # Glaspuppe Trinkgläser Glasflasche

"ß", "ä", "s", "t" -match "[^äüöß]" # s t

"a", "b", "c", "x", "y", "z" -match "[a-m]" # a b c

"web.de" -match "web\\.de" # TRUE

"\windows\temp\test.txt" -match "\\windows\\temp\\test\\.txt" # TRUE

"7543" -match "\S{6,9}" # FALSE
```

# Regex

## Zeichenklassen, Maskierung & Anker

```
"Glaspuppe", "Trinkgläser", "Glasflasche", "Plastik" -match "gl[aä]s"
# Glaspuppe Trinkgläser Glasflasche

"ß", "ä", "s", "t" -match "[^äüöß]" # s t

"a", "b", "c", "x", "y", "z" -match "[a-m]" # a b c

"web.de" -match "web\\.de" # TRUE

"\windows\temp\test.txt" -match "\\windows\\temp\\test\\.txt" # TRUE

"7543" -match "\s{6,9}" # FALSE

"7" -match "^[0-7]$" # TRUE
```

# Regex

## Zeichenklassen, Maskierung & Anker

```
"Glaspuppe", "Trinkgläser", "Glasflasche", "Plastik" -match "gl[aä]s"
# Glaspuppe Trinkgläser Glasflasche

"ß", "ä", "s", "t" -match "[^äüöß]" # s t

"a", "b", "c", "x", "y", "z" -match "[a-m]" # a b c

"web.de" -match "web\\.de" # TRUE

"\\windows\\temp\\test.txt" -match "\\windows\\temp\\test\\.txt" # TRUE

"7543" -match "\\S{6,9}" # FALSE

"7" -match "^([0-7])$" # TRUE
```



# Regex

## Zeichenklassen, Maskierung & Anker

```
"Glaspuppe", "Trinkgläser", "Glasflasche", "Plastik" -match "gl[aä]s"
                                     # Glaspuppe Trinkgläser Glasflasche

"ß", "ä", "s", "t" -match "[^äüöß]" # s t

"a", "b", "c", "x", "y", "z" -match "[a-m]" # a b c

"web.de" -match "web\\.de" # TRUE

"\windows\temp\test.txt" -match "\\windows\\temp\\test\\.txt" # TRUE

"7543" -match "\S{6,9}" # FALSE

"7" -match "^[0-7]$" # TRUE

"1A2B3C4D5E6F" -match "^[0-9A-F]*$"
```

# Regex

## Zeichenklassen, Maskierung & Anker

```
"Glaspuppe", "Trinkgläser", "Glasflasche", "Plastik" -match "gl[aä]s"
                                     # Glaspuppe Trinkgläser Glasflasche

"ß", "ä", "s", "t" -match "[^äüöß]" # s t

"a", "b", "c", "x", "y", "z" -match "[a-m]" # a b c

"web.de" -match "web\\.de" # TRUE

"\\windows\\temp\\test.txt" -match "\\windows\\temp\\test\\.txt" # TRUE

"7543" -match "\\S{6,9}" # FALSE

"7" -match "^[0-7]$" # TRUE

"1A2B3C4D5E6F" -match "^[0-9A-F]*$" # TRUE
```

# Regex

## Zum Weiterlesen und Üben

<https://www.script-example.com/regex>

<https://www.webmasterpro.de/coding/einfuehrung-in-regular-expressions/>

[https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about\\_regular\\_expressions](https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_regular_expressions)

<https://www.regular-expressions.info/>

<https://regexone.com/>