Regular Expressions

Marion Fischerlehner

Anja Scheuchenstuhl

Überblick + Motivation

- Regular Expressions (auch regex, RE oder regex pattern)
- Zeichenfolgen suchen, extrahieren, ersetzen, validieren
- spezialisierte Programmiersprache in Python integriert
- import re

Literatur: https://docs.python.org/3/howto/regex.html#regex-howto

Funktionen

Funktion	Beschreibung
re.compile(*pattern*, *flags=0*)	erstellt aus Regex -Pattern ein Regex -Objekt, das wiederholt aufgerufen werden kann.
re.match(*pattern*, *string*, *flags=0*)	Überprüft, ob der Regex zu Beginn des Strings passt
re.search(*pattern*, *string*, *flags=0*)	Sucht den gesamten String nach einem Muster ab
re.findall(*pattern*, *string*, *flags=0*)	Gibt alle Vorkommen eines Musters im String als Liste zurück
re.sub(*pattern*, *repl*, *string*, *count=0*, *flags=0*)	Ersetzt Teile eines Strings, die einem Muster entsprechen
re.split(*pattern*, *string*, *maxsplit=0*, *flags=0*)	Teilt einen String an allen Stellen, die einem Muster entsprechen

Grundlegende REGEX Muster

Einfache Zeichen (literal characters)

- Meisten Zeichen entsprechen sich selbst
- Achtung: case sensitiv!

```
import re
beispiel = "Einführung in das Programmieren"
muster = "Einführung"

re.findall(muster, beispiel)
```

Metacharacters

Metacharacter	Bedeutung	Beispiel
`.` (Punkt)	Passt zu jedem Zeichen (außer Zeilenumbruch)	`a.c` passt zu `abc`, `aXc`, aber nicht zu `ac`
`^` (Zirkumflex)	Passt nur am Anfang eines Strings/Zeile	`^Hello` passt zu `Hello World`, aber nicht zu `Well Hello`
`$` (Dollar)	Passt nur am Ende eines Strings/Zeile	`end$` passt zu `the end`, aber nicht zu `ending soon`
`*` (Sternchen)	0 oder mehr Wiederholungen des vorherigen Zeichens	`ba*` passt zu `b`, `ba`, `baa`, `baaa`

Metacharacters

Metacharacter	Bedeutung	Beispiel
+ (Plus)	1 oder mehr Wiederholungen des vorherigen Zeichens	ba+ passt zu ba, baa, baaa, aber nicht zu b
? (Fragezeichen)	0 oder 1 Vorkommen des vorherigen Zeichens	ba? passt zu b und ba, aber nicht zu baa
{n}	Genau n Wiederholungen des vorherigen Zeichens	ba{2} passt zu baa, aber nicht zu ba oder baaa
{n,m}	Zwischen n und m Wiederholungen des vorherigen Zeichens	ba{1,3} passt zu ba, baa, baaa

Metacharacters

Metacharacter	Bedeutung	Beispiel
[] (eckige Klammern)	Definiert eine Zeichenklasse (Menge von Zeichen)	[aeiou] passt zu beliebigen Vokal, [A-Za-z] zu beliebigen Buchstaben
| (Pipe)	ODER-Verknüpfung für Auswahl zwischen Optionen	cat|dog passt zu cat oder dog.
() (runde Klammern)	Gruppiert Ausdrücke + ermöglicht wiederholte Anwendung	(ab)+ passt zu ab, abab, ababab
\ (Backslash)	Maskiert Sonderzeichen oder definiert spezielle Sequenzen	\. passt zu einem Punkt, \d zu einer Zahl (0-9)

Metacharacters

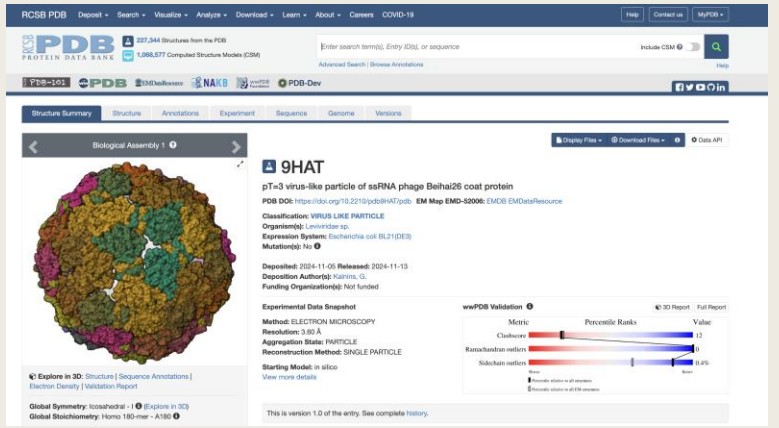
```
sequence = "AUGCGUUACGGAUGUGCCGCGUAAGCUA"
pattern = re.compile(r"^AUG")
match = pattern.findall(sequence)
```

```
sequence = "AUGCGAAAUACGGAUGUGCCGCGUAAGCUA"
pattern = re.compile(r"(UAA)$")
match = pattern.findall(sequence)
```

```
sequence = "AUGCGAAAUACGGAUGUGCCGCGUAAGCUA"
pattern = re.compile(r"(UAA|UAG|UGA)$")
match = pattern.findall(sequence)
```

Praktische Regex- Anwendungen

Regex in der Praxis



Erstelle ein Pattern, damit nur Proteine aus der die nach 4 Stellen mit „.pdb“ enden ausgegeben werden. Tipp: 1. Character ist immer numerisch

Lösung: re.compile(r“[1-9][A-Z0-9]{3}\.pdb\b“)

2MLB.pdb	EHEE_rd4_0172.pdb	r9_1163_TrROS_Hall.pdb
2MMP.pdb	EHEE_rd4_0195.pdb	r9_340_TrROS_Hall.pdb
2MP8.pdb	EHEE_rd4_0300.pdb	r9_366_TrROS_Hall.pdb
2MPK.pdb	EHEE_rd4_0325.pdb	r9_733_TrROS_Hall.pdb
2MR9.pdb	EHEE_rd4_0340.pdb	r9_814_TrROS_Hall.pdb
2MRL.pdb	EHEE_rd4_0394.pdb	v2K28S:R29S_2M7K.pdb
2MSG.pdb	EHEE_rd4_0463.pdb	v2K43S_2KVV.pdb
2MSJ.pdb	EHEE_rd4_0499.pdb	v2R14S:R16S_2L3X.pdb
2MUL.pdb	EHEE_rd4_0502.pdb	v2R31S:R32S_2N5D.pdb
2MW9.pdb	EHEE_rd4_0510.pdb	v2_1OGW.pdb
2MWA.pdb	EHEE_rd4_0625.pdb	v2_1PGY.pdb
2MWB.pdb	EHEE_rd4_0726.pdb	v2_1PV0.pdb
2MWF.pdb	EHEE_rd4_0840.pdb	v2_1WR7.pdb
2MWR.pdb	EHEE_rd4_0864.pdb	v2_2BN8.pdb
2MXD.pdb	EHEE_rd4_0877.pdb	v2_2HDZ.pdb
2MYX.pdb	EHEE_rd4_0929.pdb	v2_2J6K.pdb
2N2U.pdb	GG:run1_0874_0002.pdb	v2_2KIS.pdb
2N35.pdb	GG:run1_1113_0003.pdb	v2_2L15.pdb
2N3S.pdb	GG:run1_1390_0006.pdb	v2_2LC2.pdb

Zusammenfassung

- wichtiges Tool beim Arbeiten mit komplexen Texten und Daten
- Zeichenfolgen suchen, extrahieren, ersetzen oder validieren
- wichtig, sich mit den grundlegenden Metacharacters und der Regex-Syntax auseinanderzusetzen