Strings

In [ ]:
```
# Erstellen von Strings
'Um einen String in Python zu erstellen, benutzen wir entweder einfache oder dop
```

In [ ]:
```
# Einzelnes Wort
'hallo'
```

In [2]:
```
# Gesamter Satz
"Dies ist ein Beispiel für einen ganzen Satz"
```

Out[2]:
```
'Dies ist ein Beispiel für einen ganzen Satz'
```

In [3]:
```
# Wir können auch doppelte Anführungszeichen benutzen
"String mit doppelten Anführungszeichen"
```

Out[3]:
```
'String mit doppelten Anführungszeichen'
```

In [3]:
```
print('Hallo Erde')
print('Hallo Mond')
print('Nutze \n um in einer neuen Zeile auszugeben')
print('\n')
print('Sehen Sie, was ich meine?')
```

```
Hallo Erde
Hallo Mond
Nutze
 um in einer neuen Zeile auszugeben


Sehen Sie, was ich meine?
```

In [5]:
```
# Länge des Strings ermitteln
len('Hallo Welt11')
```

Out[5]:
```
12
```

In [7]:
```
# Indizierung - Indizierung beginnt bei 0!
der_string='FH-Aachen!'
```

In [8]:
```
der_string[0]
```

Out[8]:
```
'F'
```

In [9]:
```
der_string[9]
```

Out[9]:
```
'!'
```

In [ ]:
```
# Der Slicing-Operator:
```

In [11]:
```
# Ausgabe des Strings nach dem einschließlich fünften Zeichen
der_string[5:]
```

Out[11]:
```
'chen!'
```

In [6]:
```
# Wie sieht das Original aus?
der_string
```

Out[6]:
```
'FH-Aachen!'
```

In [7]:
```
# Bis zum dritten Index
der_string[:3]
```

Out[7]:
```
'FH-'
```

In [13]:
```
# Letzes Zeichen
der_string[-3]
```

Out[13]:
```
'e'
```

In [14]:
```
# Strings sind unveränderlich!
der_string[0]='x'
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[14], line 2
      1 # Strings sind unveränderlich!
----> 2 der_string[0]='x'

TypeError: 'str' object does not support item assignment
```

In [15]:
```
# Strings "erweitern"
der_string + ' ist nicht nur in Aachen'
```

Out[15]:
```
'FH-Aachen! ist nicht nur in Aachen'
```

In [15]:
```
der_string
```

Out[15]:
```
'FH-Aachen!'
```

In [ ]:
```
# Grundlegende String Methoden
```

In [17]:
```
neuer_string='FH-Aachen!'
```

In [18]:
```
# Nur Großbuchstaben
neuer_string.upper()
```

Out[18]:
```
'FH-AACHEN!'
```

Einblenden der Kontexthilfe

In [19]:
```
# Nur Kleinbuchstaben
neuer_string.lower()
```

Out[19]:
```
'fh-aachen!'
```

In [20]:
```
neuer_superstring='Bald ist Ostern'
```

In [21]:
```
# Splitten der Zeichenkette, Rückgabe als Liste
neuer_superstring.split()
```

Out[21]:
```
['Bald', 'ist', 'Ostern']
```

In [18]:
```
# Slicing in Python ermöglicht es, Teile eines Sequenztyps (wie Strings, Listen
mein_string = "Der Winter ist großartig!"

# Extraktion eines Teils des Strings
teil_string = mein_string[11:14]

print(teil_string)
```

```
ist
```

Übungsaufgaben

1. Geben Sie bitte bei dem String 'Hallo' das "a" mittels Indizierung aus

2. Schreiben Sie ein Python-Programm, das einen beliebigen Satz als Zeichenkette entgegen nimmt. Verwenden Sie dann das Slicing-Konzept, um Teile des Satzes zu extrahieren und auszugeben.

- Extrahiere die ersten fünf Zeichen des Satzes.
- Extrahiere einen Teil des Satzes von Index 7 bis 10.
- Extrahiere den Rest des Satzes ab Index 11 bis zum Ende.

Stelle sicher, dass du die extrahierten Teile des Satzes in geeigneter Weise ausgibst.

3. Schreiben Sie bitte den String "Hallo" rückwärts mittels Slicing

4. Gegeben ist folgender String 'Hallo'. Zeigen Sie bitte zwei unterschiedliche Methoden, um den Buchstaben "o" durch Indizierung zu erhalten