Enkelvoudige variabelen - Strings

Een String is een enkelvoudig datatype voor textuele content.

String variabelen aanmaken en afdrukken

Om een String (tijdelijk) in het gegeugen op te slaan, kun je een variabele aanmaken.

In [15]:
```
planet = "Mercury"
```

De variabele planet heeft nu de waarde "Mercury" en is krijgt daardoor het type String

In [16]:
```
type(planet)
```

Out[16]:
```
str
```

Gebruik camelCase voor de naam van een variabele

In [17]:
```
myHomePlanet = "Earth"
```

Gebruik haakjes voor het afdrukken van een string

In [18]:
```
print(planet)
```

```
Mercury
```

Zowel enkele als dubbele quotes zijn toegestaan.

In [19]:
```
star = 'Sun'
type(star)
```

Out[19]:
```
str
```

Wanneer gebruik je enkele en wanneer dubbele quotes? Wees vooral praktisch..

In [20]:
```
print("Merury doesn't have an atmosphere because it is so close to the Sun")
print('"We choose to go to the moon in this decade and do the other things,
```

```
Merury doesn't have an atmosphere because it is so close to the Sun
"We choose to go to the moon in this decade and do the other things, not be
cause they are easy, but because they are hard" – JFK
```

Variabelen kunnen als onderdeel van een omvattende String worden afgedrukt

met de methode **format()**

In [21]:
```
formatStr = "Being the planet closest to the {0}, {1} guards its secrets ver
```

door gebruik van een **f-string**

In [22]:
```
fStr = f"Being the planet closest to the {star}, {planet} guards its secrets
```

met de **String Formatting Operator**

In [23]:
```
formatOperatorStr = "Being the planet closest to the %s, %s guards its secre
```

In [24]:
```
sentence = formatStr
print(formatStr)
```

```
Being the planet closest to the Sun, Mercury guards its secrets very carefu
lly
```

Gebruik comparators om de waarden van variabelen met elkaar te vergelijken

In [25]:
```
formatStr == fStr == formatOperatorStr
```

Out[25]:
```
True
```

Exercise 1

Print de volgende String op drie verschillende manieren en met gebruik van de variabelen *planet*, *star* en *myHomePlanet*

"Since Mercury is so much closer to the Sun than we are, in the Mercurian sky the Sun appears three times as big as from here on Earth"

In [26]:
```
formatStr = "Since {0} is so much closer to the {1} than we are, in the Merc

planet = 'Mercury'
star = 'Sun'
myHomePlanet = 'Earth'

fStr = f"Since {planet} is so much closer to the {star} than we are, in the

formatOperatorStr = "Since %s is so much closer to the %s than we are, in th

print (formatStr)
print (fStr)
print (formatOperatorStr)
```

```
Since Mercury is so much closer to the Sun than we are, in the Mercurian sk
y the Sun appears three times as big as from here on myHomePlanet
Since Mercury is so much closer to the Sun than we are, in the Mercurian sk
y the Sun appears three times as big as from here on Earth
Since Mercury is so much closer to the Sun than we are, in the Mercurian sk
y the Sun appears three times as big as from here on Earth
```

De variabelen zijn niet aan de methode format() meegegeven

String operaties

Ookal verander je de waarde van de *geinjecteerde variabele*, de samengestelde variabele behoudt zijn oorspronkelijke waarde

In [27]:
```
planet = "Venus"
print(planet)
print(formatStr)
```

```
Venus
Since Mercury is so much closer to the Sun than we are, in the Mercurian sk
y the Sun appears three times as big as from here on myHomePlanet
```

Ook na de operatie *replace()* variabele sentence nog zijn initiele waarde

In [28]:
```
print(sentence.replace("Mercury","Venus"))
print(sentence)
```

```
Being the planet closest to the Sun, Venus guards its secrets very carefull
y
Being the planet closest to the Sun, Mercury guards its secrets very carefu
lly
```

Gebruik functies voor het omzetten van en naar hoofdletters in Strings

In [29]:
```
print("The NEAR Mission".upper())
print("Dwarf Planets".lower())

cosmonaut = "alexi leonov"
print("Cosmonaut {}".format(cosmonaut.title()))
print(f"Cosmonaut {cosmonaut.title()}")
```

```
THE NEAR MISSION
dwarf planets
Cosmonaut Alexi Leonov
Cosmonaut Alexi Leonov
```

Exercise 2

Print de volgende String met gebruik van de gegeven variabelen, een String Formatting Operator en String operaties, zonder de variabelen zelf te veranderen

"So far, KBOs have been seen from distances ranging from 30AU to 50AU from the Sun (for reference, Pluto's average orbit is at about 39AU)."

In [30]:
```
from turtle import title

object = "kbo"
distanceUnit = "au"
star = "SUN"

formatOperatorStr = "So far, %ss have been seen from distances ranging from
print (formatOperatorStr)
```

```
So far, KBOs have been seen from distances ranging from 30AU to 50AU from t
he Sun (for reference, Pluto's average orbit is at about 39AU).
```

Correct

Gebruik split() om een zin op te splitsen in woorden

In [31]:
```
rockyPlanets = "Mercury, Venus, Earth and Mars".replace(',','').split()
rockyPlanets.remove("and")
print(rockyPlanets)
```

```
['Mercury', 'Venus', 'Earth', 'Mars']
```

Inhoud van een String over meerdere regels

In [32]:
```
multiLineStr = """
               Standing on Mercury's surface at its closest approach to the
               our star would appear more than three times larger than it d
               """
print(multiLineStr)
```

```

               Standing on Mercury's surface at its closest approach to th
e Sun,
               our star would appear more than three times larger than it
does on Earth.
```