

Programmzeilen mit der IDLE Shell testen

Gib die Programmzeilen nach dem Prompt (>>>) ein. Ist das Ergebnis OK oder bekommst du eine Fehlermeldung?

Tipp: Mit ALT + p kannst du die letzte Zeile wiederholen.

Zeichenketten (strings) testen

Programmzeile	OK	Fehler
Hallo Welt		
"Hallo Welt"		
"Hallo Welt'		
'Hallo Welt'		
print("Hallo Welt")		
pirnt("Hallo Welt")		

Zeichenketten in Variablen schreiben

Programmzeile	OK	Fehler
nachricht = Hallo Welt		
nachricht = "Hallo Welt"		
nachricht		
print(nachricht)		
len(nachricht)		
type(nachricht)		
nachricht.upper()		

Zahlen (integer und float) testen

Programmzeile	OK	Fehler
3		
3.14		
3,14		
print(3)		
print(3.14)		
print(3,14)		

Zahlen in Variablen schreiben

Programmzeile	OK	Fehler
zahl = 3		
zahl = 3.14		
zahl		
print(zahl)		
len(zahl)		
type(zahl)		
zahl.upper()		

Die Farbe kennzeichnet einen Text mit besonderer Bedeutung

Farbe	Bedeutung
rot	
grün	
violett	

Mit der IDLE Shell rechnen

Gib die Programmzeilen nach dem Prompt (>>>) ein. Ist das Ergebnis OK oder bekommst du eine Fehlermeldung?

Rechnen mit Zeichenketten

Programmzeile	OK	Fehler
"Hallo" + "Welt"		
"Hallo" + " " + "Welt"		
"Hallo " + "Welt"		
"Hallo " - "Welt"		
print("Hallo " + "Welt")		
3 * "Hallo Welt"		
3 * "Hallo Welt "		
3 / "Hallo Welt "		
print(3 * "Hallo Welt ")		
nachricht = "Hallo Welt "		
print(3 * nachricht)		

Rechnen mit Zahlen

Programmzeile	OK	Fehler
3 + 4		
3 - 4		
3 * 4		
3/4		
30//4		
30%4		
ergebnis = 3 * 4		
ergebnis/5		
ergebnis//5		
print(ergebnis/5)		

Punktrechnung vor Strichrechnung

Programmzeile	OK	Fehler
3 + 4 * 2		
(3 + 4) * 2		
12 - 3 / 2		
(12 - 3) / 2		
12 - 3 // 2		
(12 - 3) // 2		

Zahl in Zeichenkette umwandeln – Zeichenkette in Zahl umwandeln

Programmzeile	OK	Fehler
zahl = 3		
str(zahl)		
zahl = 3.14		
str(zahl)		
zeichenkette = "3.14"		
int(zeichenkette)		
float(zeichenkette)		

Mit dem IDLE Editor ein Programm schreiben

Mit dem Menüpunkt File/ New File erzeugst du ein leeres Editor-Fenster mit dem Titel "untitled".

Gib die Programmzeilen im Editor-Fenster ein. Die Farbe dort kennzeichnet einen Text mit besonderer Bedeutung.

Speichere den Inhalt mit dem Menüpunkt File/ Save unter dem Dateinamen "Editor_test" im Ordner Python/03_Rechnen_mit_Zeichenketten_und_Zahlen

Starte das Programm mit dem Menüpunkt Run/ Run Module.

Notiere die print-Ausgaben in der Tabelle.

Rechnen

Programmzeile	print-Ausgabe
# Das ist ein Kommentar	
# Rechnen	
nachricht = "Hallo Welt "	
print(3 * nachricht)	
ergebnis = 3 * 4	
print(ergebnis/5)	

Umwandeln

Programmzeile	print-Ausgabe
# Umwandeln	
zahl = 3.14	
print(zahl)	
print(str(zahl))	
zeichenkette = "3.14 "	
print(zeichenkette)	
print(3 * zeichenkette)	
print(3 * float(zeichenkette))	

Variablen ausgeben

Programmzeile	print-Ausgabe
# Variablen ausgeben	
print("nachricht =", nachricht)	
print("ergebnis =", ergebnis)	
print("zahl =", zahl)	
print("zeichenkette =", zeichenkette)	

Aufgabe: Ziffern mit Punkten darstellen

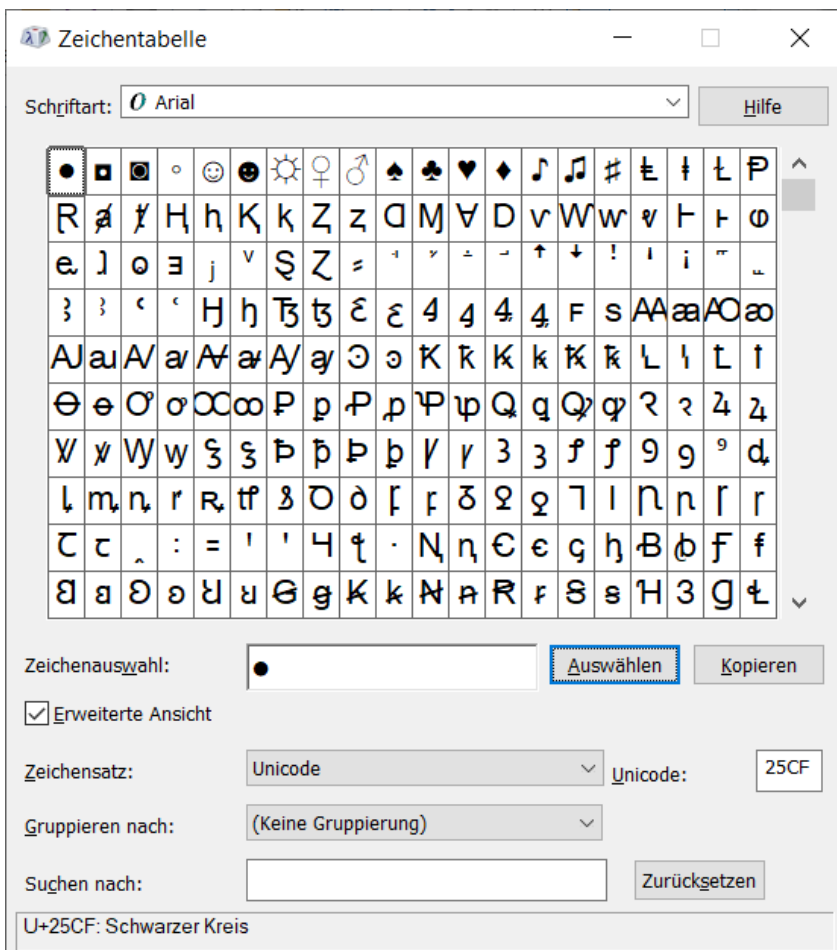
Die Ziffern auf einem Kassenzettel sind aus Punkten zusammengesetzt. Mit 5x7 Punkten können die Ziffern 0 bis 9 gut lesbar dargestellt werden.

	•	•	•	
•				•
•				•
•				•
•				•
•				•
	•	•	•	

Schreibe ein Programm, das mit print-Befehlen und dem Zeichen * oder • eine Ziffer in ein 5x7 Raster druckt:

- Drucke eine Null
- Drucke eine Acht
- Drucke eine Eins
- Drucke eine Ziffer deiner Wahl

Tipp: Das Zeichen • hat den Unicode 25CF. Es kann über die Windows-Zeichentabelle eingegeben werden.



Eine Liste speichert viele änderbare Elemente

Gib die Programmzeilen nach dem Prompt (>>>) ein.

Merke: Der Index steht in eckigen Klammern. Der Index beginnt mit Null!

Eine Liste anlegen – mit eckigen Klammern!

Programmzeile	print-Ausgabe
<code>vornamen = ["Axel", "Elke", "Martin"]</code>	
<code>print(vornamen)</code>	
<code>print(vornamen[0])</code>	
<code>print(vornamen[0:2])</code>	
<code>print(vornamen[-1])</code>	
<code>vornamen[2] = "Fritz"</code>	
<code>print(vornamen)</code>	

Eine Liste erweitern

Programmzeile	print-Ausgabe
<code>vornamen = vornamen + ["Heike", "Sabine"]</code>	
<code>print(vornamen)</code>	
<code>vornamen += ["Markus"]</code>	
<code>print(vornamen)</code>	

Eine leere Liste anlegen und füllen

Programmzeile	print-Ausgabe
<code>buchstaben = []</code>	
<code>buchstaben.append("a")</code>	
<code>print(buchstaben)</code>	
<code>buchstaben.append("b")</code>	
<code>print(buchstaben)</code>	

Elemente einer Liste löschen

Programmzeile	print-Ausgabe
<code>print(vornamen)</code>	
<code>vornamen.remove("Heike")</code>	
<code>print(vornamen)</code>	
<code>del vornamen[0]</code>	
<code>print(vornamen)</code>	
<code>del vornamen</code>	
<code>print(vornamen)</code>	

Ein zufälliges Element aus einer Liste auswählen

Programmzeile	print-Ausgabe
<code>import random</code>	
<code>handzeichen = ["Schere", "Stein", "Papier"]</code>	
<code>print(random.choice(handzeichen))</code>	

Ein Tupel speichert viele nicht änderbare Elemente

Gib die Programmzeilen nach dem Prompt (>>>) ein.

Merke: Der Index steht in eckigen Klammern. Der Index beginnt mit Null!

Ein Tupel anlegen – mit runden Klammern!

Programmzeile	print-Ausgabe
punkt = (-10, 5, 7)	
print(punkt)	
print(punkt[0])	
print(punkt[0:2])	
print(punkt[-1])	
punkt[2] = 15	
punkt = (-10, 5, 15)	
print(punkt)	

Ein Element im Tupel suchen

Programmzeile	print-Ausgabe
print(punkt)	
print(punkt.count(7))	
print(punkt.index(7))	
print(punkt[0:2])	
print(punkt[-1])	
punkt[2] = 15	
punkt = (-10, 5, 15)	
print(punkt)	