

①

w / f: $j * j$ liefert das erwartete Ergebnis von -1 und hat den Datentyp `complex`.

w / f: Die Funktion `len()` gibt für `[[1, 3, [4, 5], 6], 7]` den Wert 2 aus.

w / f: Ein Set, welches aus der Liste `[1, 2, 5, 1, 0, 2, 3, 1, 1, (1, 2, 3)]` erzeugt wird, umfasst 10 Elemente

w / f: Das Ergebnis von `len([[1, 2]] * 3)` ist 3.

w / f: Von den drei Anweisungen a) `min(["a", "b", "c"])`, b) `max([1, 2, "three"])`, c) `[1, 2, 3].count("one")` verursacht nur die Anweisung `[1, 2, 3].count("one")` einen Fehler.

w / f: Bei der Anweisung `new_x = [i for i in x if i >= 0]` handelt es sich um eine List Comprehension, welche alle negativen Werte der Liste `x` entfernt.

w / f: Die `pass`-Anweisung in einer Kontrollstruktur entspricht einer `break`-Anweisung in Java.

w / f: Die Ausgabe für den Ausdruck `x = "%(a).2f" % {'a': 1.1111}` lautet: `1.11110000`.

w / f: Die Zahlen `0`, `0.0`, und `0 + 0j` geben den boolschen Wert `False` zurück, wohingegen alle anderen Zahlen den boolschen Wert `True` zurückliefern.

w / f: Der Ausdruck `f'{97:c}'` formatiert den Integer gemäß der ASCII Tabelle als dazugehörigen Character und liefert das Zeichen `'a'`.

②

1. `bar()`,
2. `varName`,
3. `VERYLONGVARNAME`,
4. `foobar`,
5. `longvarname`,
6. `foo_bar()`,
7. `really_very_long_var_name`

1. keine Klammern
2. Wähle ein guter Klassenname wenn es im UpperCamelCase wähle
3. Einzelne Wörter müssen durch Unterstriche getrennt werden damit man es als Konstanten benutzen kann

A: Nach Python Konvention sind 4 und 7 gute Variablen.

③

1. 1.6 Float

2. Syntax Error: unexpected character after line continuation character $i \backslash \% 3_1 \mid i \% 3$
 $= 1$

3. 'gaganene' String

4. 7 int

Name Error: name 's' is not defined $\mid \text{print}(a)$

5. ZeroDivisionError: float division by Zero $\mid 1.0/1.0$

6. ValueError: invalid literal for int() with base 10: '2.0' $\mid b = "2"$

7. '3' string

8. 27.0 float