

Start in Python mit WebTigerPython

Was ist Python?

Python ist eine Programmiersprache. Mit ihr kann man dem Computer Schritt für Schritt sagen, **was er tun soll**.

Python wird sehr häufig verwendet, zum Beispiel:

- in der Informatik
- in der Wissenschaft
- in der Datenanalyse
- in der Webentwicklung
- für kleine Programme und grosse Anwendungen

Python ist besonders geeignet für Einsteigerinnen und Einsteiger, weil:

- die Sprache gut lesbar ist,
- sie der normalen Alltagssprache ähnelt,
- man schnell erste Resultate sieht.



Python Code Beispiel

Py

```

1 def fib(n):
2     a, b = 0, 1
3     while a < n:
4         print(a, end=' ')
5         a, b = b, a+b
6     print()
7     fib(1000)

```

Wir werden Python Code immer in diesen **Boxen** darstellen. Man erkennt sie am **grünen Label** oben mit dem **Py** Icon.

Die Zahlen am linken Rand geben die **Zeilennummern** an und helfen uns, den Überblick zu behalten (diese müssen nicht mit kopiert werden).

Was ist WebTigerPython?

WebTigerPython ist eine **webbasierte Umgebung**, mit der Sie Python **direkt im Browser** ausprobieren können.

Das bedeutet:

- Sie müssen **nichts installieren**
- Sie brauchen **nur einen Webbrowser**
- Sie können sofort mit Python arbeiten

WebTigerPython wurde speziell für den **Unterricht** entwickelt und eignet sich sehr gut für den Einstieg.



WebTigerPython

Wie funktioniert WebTigerPython?


1. **Öffnen Sie im Browser die Webseite:**
<https://webtigerpython.ethz.ch>
2. Geben Sie im **Editorfenster** folgenden Code ein:

1 x = 5
2 print(x)

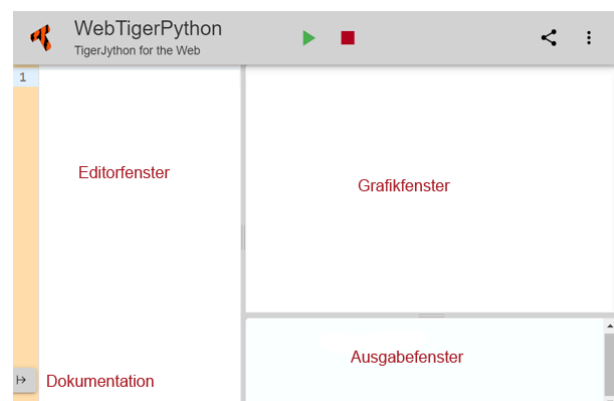
Py

Das sind die Anweisungen für den Computer

3. **Programm ausführen:**

Mit einem Klick auf **Ausführen-Knopf**  oben wird das Programm ausgeführt.

- Der Computer liest den Code von oben nach unten
- Jede Zeile wird nacheinander verarbeitet
- Das Ergebnis erscheint im **Ausgabefenster**



Wichtig für den Start

- Programmieren lernt man **durch Ausprobieren**
- Fehler sind **erwartet und erlaubt**
- Es geht nicht darum, alles sofort zu verstehen
- Ziel ist es, Schritt für Schritt sicherer zu werden

Ein **Computer denkt nicht selbst** und versteht keine Bedeutungen. Er führt nur **exakte Anweisungen** aus.

Schon kleine Änderungen im Code können ein anderes Resultat ergeben. Deshalb ist genaues Arbeiten wichtig – Fehler gehören aber ganz normal zum Lernen dazu.

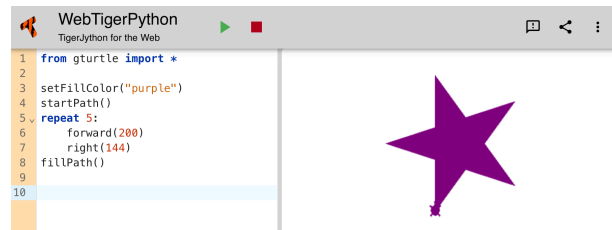
Einstieg mit Turtlegrafik

Für den Einstieg ins Programmieren verwenden wir **Turtlegrafik**.

Das ist eine einfache Art, mit Python **Zeichnungen zu erstellen**. Dabei steuert man eine kleine Schildkröte (engl. turtle) auf dem Bildschirm.

Die Schildkröte:

- kann sich vorwärts und rückwärts bewegen,
- kann sich drehen
- und hinterlässt beim Bewegen eine Linie.



```

1 from turtle import *
2
3 setFillColor("purple")
4 startPath()
5 repeat 5:
6     forward(200)
7     right(144)
8 fillPath()
9
10
  
```

Turtlegrafik ist **keine eigener Programmiersprache**, sondern ein sogenanntes **Package für Python**.

Ein Package ist eine für einen bestimmten Zweck gedachte **Erweiterung** für Python, d.h. eine Sammlung von **zusätzlichen Befehlen**.

Python selbst kann sehr viel und mit Packages kann man Python noch weiter ausbauen.

Aufgaben

Wir werden uns als Einstieg mit dem Lernprogramm **Turtlegrafik** beschäftigen.

1. Lesen Sie zuerst den Abschnitt zu **WebTigerPython** auf [dieser Seite](#) durch und machen Sie sich mit der Bedienung vertraut.
2. Nun werden wir die ersten Python Programme schreiben. Arbeiten Sie dazu die Seite [Turtle bewegen](#) durch. Probieren Sie das **Musterbeispiel** aus und lösen Sie die Aufgaben im Abschnitt **Zum selbst Lösen**.

Tipp

Speichern Sie Ihre Lösungen jeweils als Python-Dateien mit sinnvollen Namen auf Ihrem Computer ab, damit Sie sie wieder finden.

Alternativ können sie den Code von Ihren Lösungen ins OneNote kopieren.

Beginnen Sie für jede Aufgabe wieder mit einem leeren Editor-Fenster.

3. Nachdem wir nun wissen, wie wir den Turtle bewegen um Figuren zu malen, wollen wir uns mit den Farben beschäftigen. Arbeiten Sie dazu die Seite [Farben verwenden](#) inklusive **Musterbeispiel** und Aufgaben **Zum selbst Lösen** durch.
4. Nun fehlt uns noch ein hilfreiches Element: die [Wiederholungen](#). Arbeiten Sie auch diese Seite inklusive **Musterbeispiel** und Aufgaben **Zum selbst Lösen** durch.
5. Überlegen Sie sich eine **kleine Zeichnung**, die Sie mit dem Turtle programmieren möchten. Dies kann ein Buchstabe, ein Symbol oder eine kleine kreative Form sein.
 - **Implementieren** Sie das Programm in WebTigerPython.
 - Erstellen Sie einen **Screenshot** von der fertigen Zeichnung und senden diesen zusammen **mit Ihrem Code** via **Teams** an mich.

