

# Praktikum Programmieren - #3

---

## Die Bibliothek Turtle

---

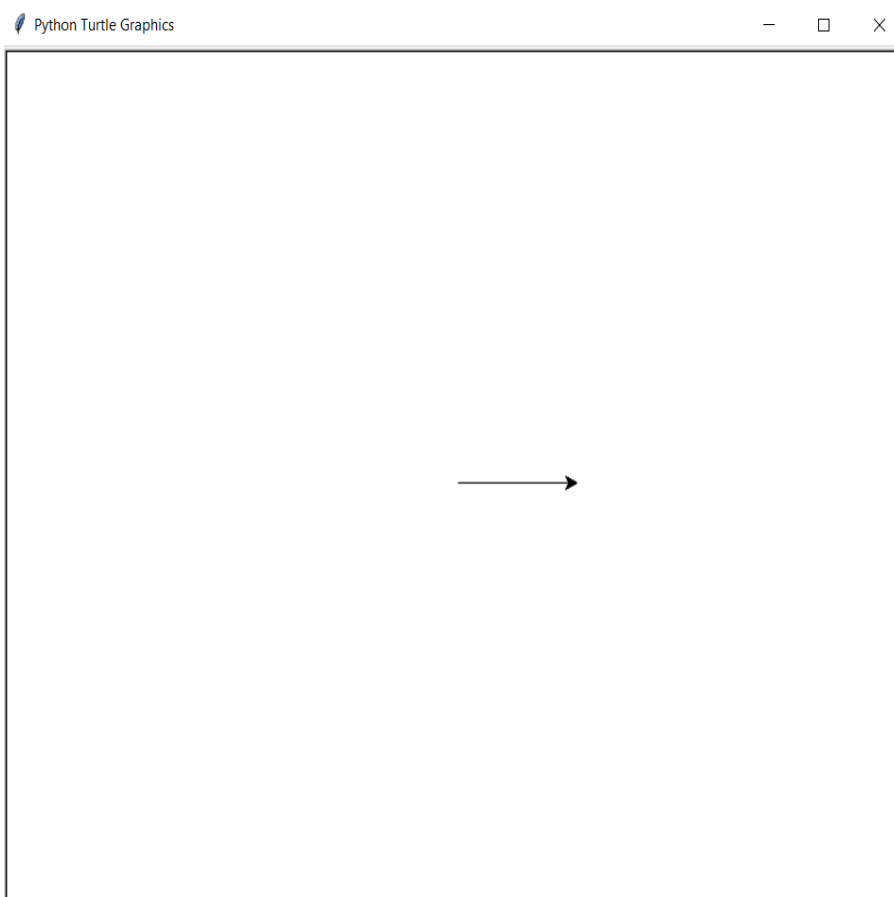
Anfangs wurde die Sprache Turtle zum Lernen einer Computersprache für Kinder entwickelt. Eine Schildkröte (Turtle) kann über den Bildschirm bewegt werden. Dabei hinterlässt die Schildkröte auf ihrer Bewegungsbahn eine Linie. So können Grafiken erstellt werden.

Minimalbeispiel:

```
import turtle

# Turtle "läuft" 100 Pixel vorwärts von Ausgangsposition
turtle.forward(100)
turtle.done()
```

Entspricht:



# 1. Geometrische Formen mit Turtle

---

Erkunde die [Dokumentation](#) (Aufruf 10.11.2022) von Turtle und nutze die Informationen für folgende Aufgaben

- zeichnen Sie ein Quadrat
- zeichnen Sie ein Rechteck
- zeichnen Sie ein gleichseitiges Dreieck
- zeichnen Sie ein gleichmäßiges 6-Eck

## 2. Geometrische Formen als Klassen

---

Erstellen Sie die Klassen *Square*, *Rectangle*, *Triangle* und *RegularPolygon* zur Erstellung von Objektinstanzen.

### 2.1 Klassen schreiben

- Alle Klassen sollen den Parameter `position` enthalten vom Typ `tuple`, sodass man Positionsangaben in der Form `(x, y)` übergeben kann. Setzen Sie als default-Wert die Position `(0, 0)`.
- Überlegen und implementieren Sie weitere geeignete Parameter für die `__init__()`-Methoden der jeweiligen Klassen.
- Implementieren Sie für jede Klasse eine `draw()`-Methode
- Implementieren Sie für jede Klasse die Methode `calculate_area()`, welche die Fläche des Objekts zurückgibt
- die Methode `turtle.goto((x, y))` lässt die Schildkröte zu der im Tupel `(x, y)` angegebenen Position wandern. Dabei zeichnet die Schildkröte jedoch eine Linie. Schreiben sie eine Klassenmethode `move_to_position(new_position)` welche zu der neuen Position wandert ohne eine Linie zu zeichnen (in der Dokumentation von Turtle recherchieren)
- Erweitern sie alle `__init__()`-Methoden um den Parameter `starting_angle` und alle Klassen um die Methode `set_starting_angle(angle)`, mit welchem die Ausrichtung des Objekts beim Zeichnen gesteuert werden kann (Anpassung der `draw()`-Methode nötig!)

## 2.2 Objekte erstellen und zeichnen

---

- Erstellen Sie unterhalb der Klassendefinitionen eine `main()` -Funktion
- Erstellen Sie mehrere Objekte mit Variationen in Größe, Position und Startwinkel und speichern Sie diese in einer Liste
- Durchlaufen Sie die Liste mit einer `for`-Schleife und rufen Sie für jedes Objekt die `draw` -Methode auf und geben Sie in der Konsole die Flächeninhalte der geometrischen Formen aus

## 3. Vererbung

---

Überprüfen Sie die Gemeinsamkeiten aller Klassen in Bezug auf Parameter und Methoden. Was ist bei allen gleich? Was ist unterschiedlich bzw. klassenspezifisch?

- Erstellen Sie eine Kopie des Scripts von Aufgabe 2.1, wenn diese vollständig gelöst ist.
- Erstellen Sie die Klasse `GeometricObject` in welcher die gefundenen Gemeinsamkeiten aller Klassen vereint werden, und *cleanen* Sie die bestehenden Klassen, indem diese von der Klasse `GeometricObject` erben.