

# Schulen in der digitalen Welt

Wer? Ugur Turhal e-Mail: [ugur.turhal@me.com](mailto:ugur.turhal@me.com)

Wann? 4. September 2018

# Inhaltsverzeichnis I

Antritt

Kurze Erläuterung

Realisierung

Welche Unterlagen wurden benötigt?

Überblick

Schaltjahrrechner in der Programmiersprache Python 2.7

Beginnen wir

Funktion - Programmieren - Kompilieren

Aufbau unseres  
Programms

Anforderungen an das Programm

Analyse des Codes

Fundamentale Funktionen und Befehle

Übersetzung

Compiler

Klassenstufen

An die Klassenstufen anpassen

Fragen

# Antritt

- Jörg Wilczek und Angelika Pulfer waren auf der Suche nach einer geeigneten Person, welche den Schülern und Schülerinnen das Fach Informatik anbieten konnte.
- Anfangs hatte ich Schwierigkeiten die Kompetenzen der SuS einzuschätzen.
- Zur Einschätzung erstellte ich eine Umfrage.
- Nach der Umfrage hatte ich einen Überblick von deren Kompetenzen.

## Realisierung

- Hilfe durch Angelika Pulfer beispielsweise: Vorschläge u.ä.

Das erste Jahr:	Das zweite Jahr:
alle Klassen arbeiten mit Scratch.	erste Klassen Scratch. Zweite/dritte Klassen: Python.

- Der Informatikunterricht ging über die Anforderungen des Lehrplans hinaus.  
(Unterschiede der Betriebssysteme, Vor- und Nachteile, Aufbau eines Computers u.v.m.)
- Welche Materialien wurden verwendet?  
Python: Computer<sup>1</sup>, Betriebssysteme: Film<sup>2</sup>, Unterlagen eigene<sup>3</sup>, Scratch <https://scratch.mit.edu/>

**In aller Kürze: SuS erhalten einen Einblick in die Informatik.**

<sup>1</sup><https://codeboard.io>

<sup>2</sup><https://www.youtube.com/watch?v=RJFUJQpu0FE>

<sup>3</sup><https://cloud.edubs.ch>

## Bezug zur Praxis

Welcher Aufgabe widmen Sie sich heute?

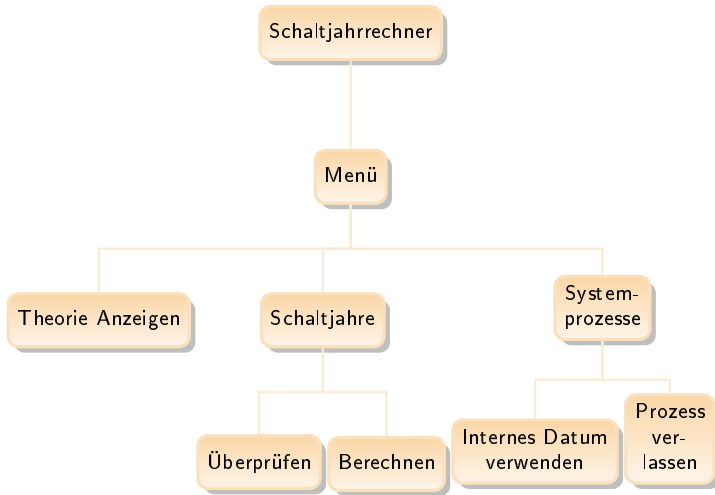
Wir analysieren kurz die Formel für die Schaltjahre. Nachfolgend werden Sie ein Programm schreiben, das weitere fünf Anforderungen erfüllen soll.

**In aller Kürze: Eine Formel in einer Programmiersprache ausdrücken und Einblick in die Programmiersprache Python 2.7 erlangen.**

## Die ersten Schritte

- Bitte registrieren Sie sich auf der Website:  
<https://codeboard.io>
- Klicken Sie auf den Button: **Create new**.
- Der Titel des Projekts ist Ihnen überlassen, ich verwende für den Titel: Schaltjahr.
- Die Beschreibung ist unwichtig (in unserem Fall).
- Bei Language wählen Sie **Python** aus. Alle anderen Sprachen sind nicht in unserem Interesse.
- Access control: **Private project** anwählen!
- Sie sind startbereit!
- Weshalb Python?
  - Python ist einsteigerfreundlich(+),
  - ist im Gegensatz zu anderen Programmiersprachen «einfacher»(+).
  - Einrücken(-).

# Aufbau



# Analyse des Codes

Befehl	Erläuterung
<code># -*-coding: utf-8<sup>4</sup> -*-</code>	Anzeige für die Umlaute auf der Konsole.
<code>import math, sys, time, datetime, calendar</code>	Import ist eine <b>Anweisung</b> , welche die fertigen Module: Mathematik, System, Zeit, Datum&Zeit und Kalender importiert. Da wir rechnen, das System auffordern das Programm zu schliessen und die interne Zeit und die Kalenderfunktion benutzen werden. <b>Kurz: Import bietet Zugriff auf Module.</b>

Tabelle: Erklärungen der Befehle

---

<sup>4</sup>Umlaute wie ä/ö/é/è



Befehl	Erläuterung
<code>eingabe = True</code> <code>while eingabe:</code>	<p><code>eingabe</code> ist eine <b>(Variable.)</b> Diese dient als Speicher für den «Wert» True. True ist ein boolescher Wert, dieser nimmt den Wert 1 an. Ein False würde den Wert 0 annehmen. Sprich: Unser Programm funktioniert solange, bis wir es beenden.</p> <p><b>Ein Programm speichert etwas in eine Variable von rechts nach links. Bspw. : unsereZahl=2. unsereZahl wird der Wert 2 zugewiesen.</b></p>

Befehl	Erläuterung
eingabe =raw_input()	Diese Funktion dient zur Eingabeaufforderung, diese liest die eingegebene Zeile ein, wertet die eingegebene Zeile nicht aus. Siehe Beispiel Test 1 von mir.
if, else, elif	Die <b>If-Bedingung</b> überprüft für uns einen Fall, falls dieser zutrifft, wird die Überprüfung terminiert. Darauf erfolgt die nächste Aktion(Bsp: print). Falls mehrere Bedingungen überprüft werden sollen, dann wird der Befehl: <b>elif</b> benutzt.
==, !=, and, or	Stellen logische Operatoren dar, diese Testen ob etwas <b>äquivalent</b> oder <b>different</b> ist ( <b>==</b> und <b>!=</b> ) falls, mehrere Bedingungen überprüft werden sollen, können and und or verwendet werden.

Befehl	Erläuterung
math.fmod(jahr,100), %	<p>Sowhl math.fmod() als auch % Rechnet mit dem Rest. Beispiel: 2424%100 hat 24 mal in 2424 Platz. Der Rest ist: 0.24 in unserer Überprüfung wird geschaut, ob der Rest == 0 ist. Doch was ist der Unterschied zwischen math.fmod() und %? Der Unterschied liegt in der Ausgabe und im Verbrauch des Speicherplatzes.</p> <p>% hat eine einfache Genauigkeit: <math>\log_{10}(2^{24})</math>, welche etwa einer Genauigkeit von 7 ~ 8 Dezimalstellen entspricht, =&gt; weniger Speicherplatz wird benötigt.</p> <p>Eine doppelte Genauigkeit hat eine Präzision von <math>\log_{10}(2^{53})</math>, was etwa einer Genauigkeit von 15 ~ 16 Dezimalstellen entspricht =&gt; mehr Speicher wird benötigt.</p>

Befehl	Erläuterung
<code>for x in range()</code>	Dies ist eine For-Schleife, x ist wie eingabe eine <b>Variable</b> , Zählvariable. Die range() ist die «Reichweite», in unserem Fall: eingetipptes Anfangsjahr und Endjahr. Die For-Schleife erhöht bei jeder Iteration (Durchlauf) x, die Zählvariable um + 1.
<code>sys.exit(0)</code>	Beendet den laufenden Prozess. Die 0 ist die Art des Verlassens. Theoretisch könnten wir auch sys.exit(1) benutzen, für uns ist das aber nicht relevant, weil kein interner Codefehler besteht.

# Kompilieren/Compile

Beim Kompilieren wird der geschriebene Programmcode (Python, Java, C++, C# etc.) in einen ausführbaren Code konvertiert.

**Der Compiler übersetzt den Code in einen Maschinencode, so dass er ausgeführt werden kann.**

## Aufteilung des Programms für Klassen

- Die Theorie ausgeben, Schuljahr: 1-9.
- Das Schaltjahr berechnen lassen, Schuljahr: 8-9.
- Das Schaltjahr über einen Zeitraum Schuljahr: 8-9.
- Das nächste Schaltjahr berechnen mit «interner» Zeit: 9. Klasse.
- Die Schliessaufforderung, Schuljahr 1-9.

# Fragen