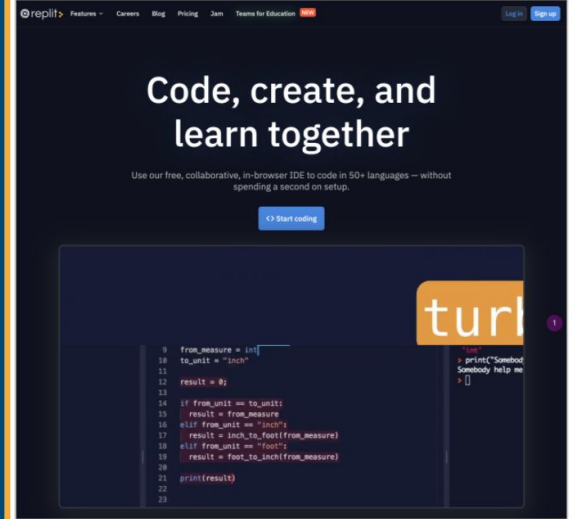




Programmieren online unterrichten

Kerstin Reese & Nadine Kirsch

Bitte Account erstellen!



repl.it



UNIVERSITÄT
DES
SAARLANDES

SIC

Saarland Informatics
Campus

Wer sind Sie?

Welchem Schulfach würden Sie Ihren Bereich zuordnen?

 Mentimeter



Haben Sie Erfahrung in Programmierung?

Mentimeter

0
Ja

0
Ein wenig

0
Nein







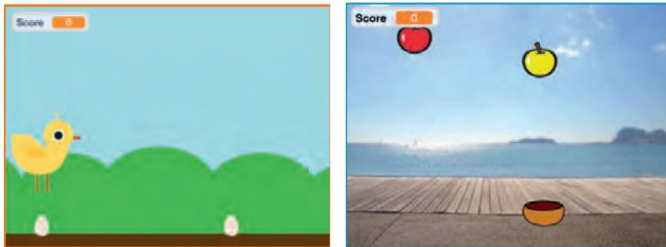
Scratch



Ferienkurs: Spieleprogrammierung mit Scratch

Im Moment kann das InfoLab Saar nicht besucht werden? Kein Problem. Wir bieten mit [InfoLabSaar@home](#) ein Ferienangebot, das online stattfindet. Für den Kurs nutzen wir die Videokonferenz-Software Zoom.

Wir bieten am **Dienstag, den 14.04.** und **Mittwoch, den 15.04.**, jeweils von **10 – 12 Uhr** eine Einführung in die grafische Programmiersprache [Scratch](#) an. Ziel ist es in Scratch ein eigenes Spiel zu programmieren.



Teilnehmen kann jedes Kind **ab 10 Jahren**. Um teilzunehmen wird ein **Computer** (Laptop oder Desktop-PC) mit **Internetzugang** benötigt. Vorkenntnisse sind keine notwendig.

Anmeldungen bitte per Mail an kerstin.reese@uni-saarland.de. Die Anzahl der Plätze ist beschränkt.

Einladung über den Newsletter

InfoLab Saar

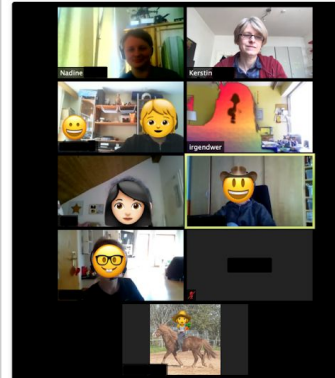
Schülerlabor Informatik an der Universität des Saarlandes

STARTSEITE EINLADUNGEN ALUMNI ONLINE-ANGEBOTE MODULE ANMELDUNG NEWSLETTER KONTAKT



Erstes Mal InfoLab Saar @home: Es hat sehr viel Spaß gemacht!

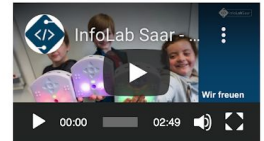
Am 14. und 15.04. hat der erste Online-Kurs des InfoLab Saar stattgefunden. Sieben Kinder konnten mit uns Scratch lernen.



Aus dem ganzen Saarland haben die Kinder von zu Hause aus am Kurs teilgenommen.

Die Kinder im Alter von 9 – 13 Jahren hatten kaum Vorerfahrung in Programmierung.

Über die Videokonferenzsoftware Zoom konnten die Teilnehmer im Team, in Paaren und auch mit 1:1-Betreuung graphische Programmierung lernen.



Das Schülerlabor Informatik ist Teil der [Informatik Didaktik der Universität des Saarlandes am Saarland Informatics Campus](#).

Newsletter-Anmeldung

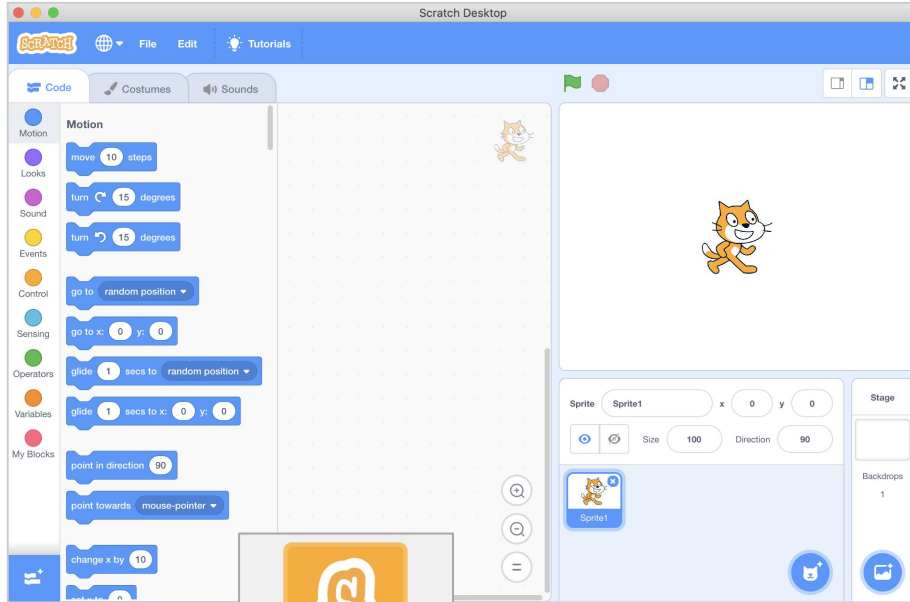
Bericht auf der Webseite

Was ist Scratch?

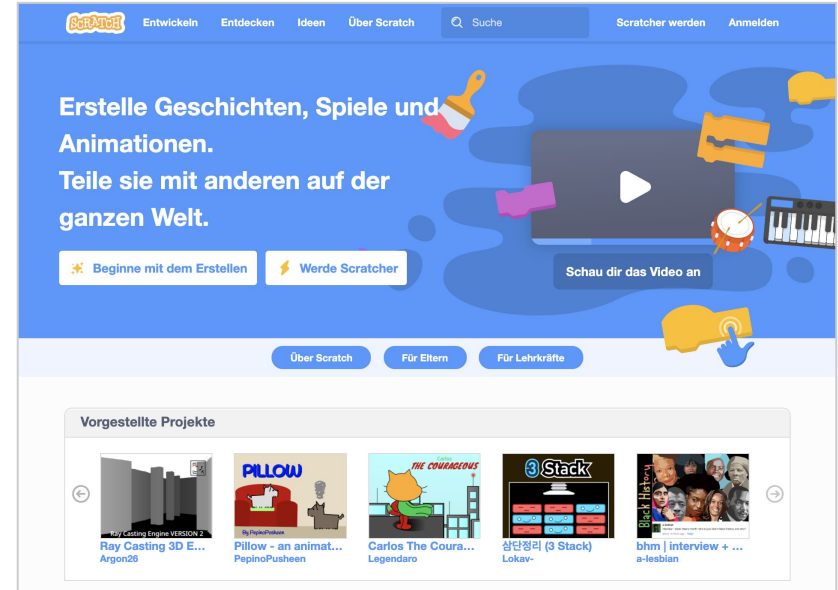


- Grafische Programmiersprache
- Befehle wie Puzzleteile/Bausteine
- Bausteine sind in natürlicher Sprache beschriftet
- Bausteine werden per Drag-and-Drop aneinandergefügt
- Programm ist lesbar ohne Vorkenntnisse
- Zielgruppe: 3. - 6. Klasse

Wo kann man Scratch programmieren?



lokal installiert



scratch.mit.edu

Webseite

Live Coding: Scratch

Scratch! | Datei | Bearbeiten | Tutorien | labyrinth remix | Veröffentlichen | Zeige Projektseite | Jetzt speichern | infolab-saar

Skripte | Kostüme | Klänge

Fühlen

- Bewegung
 - wird Mauszeiger berührt?
- Aussehen
 - wird Farbe berührt?
 - Farbe berührt ?
- Klang
- Ereignisse
- Steuerung
 - frage Wie heißt du? und warte
 - Antwort
 - Taste Leertaste gedrückt?
 - Maustaste gedrückt?
 - Maus x-Position
 - Maus y-Position
- Fühlen
- Operatoren
- Variablen
- Meine Blöcke

setze Ziehbarkeit auf ziehbar

Lautstärke

Stoppuhr

setze Stoppuhr zurück

Bühnenbildnummer von Bühne

Wenn Flagge angeklickt wird

wiederhole fortlaufend

gehe zu Mauszeiger

falls wird Ziel berührt? , dann

sage Gewonnen für 2 Sekunden

stoppe alles

falls wird Farbe berührt? , dann

sage Verloren für 2 Sekunden

stoppe alles

Figur

Hippo1

Zeige dich

Größe 20

Richtung 95

Hippo1

Ziel

Bühne

Bühnenbilder 3

Lager

Steuerung des Nilpferds, Ziel- und Kollisionserkennung.

Wenn man Scratch online unterrichtet

- Lieber im Browser programmieren, dann kann man den Code der Kinder ansehen.
- Gruppengröße 12 - 15 Kinder ab 10 Jahren. Jede Frage bremst die ganze Gruppe aus.
- Anmelden muss mit Schülerkonto Schritt-für-Schritt begleitet werden.
- Login-Namen und Passwörter einfach halten. ("Nein, die Zahlen hinter den Buchstaben habe ich nicht eingegeben.")
- Fehlermeldungen werden nicht immer mitgeteilt, z.B. bei erfolgloser Anmeldung.
- Bei jedem Seitenwechsel kann man Kinder verlieren. Junge Kinder merken nicht unbedingt, wenn sie auf der falschen Seite sind.
- Scratch programmieren ist kein Problem.

Fazit: Wenn man ausreichend Zeit einplant, lässt sich Scratch online sehr gut unterrichten!

Python



Ferienkurs: Lerne Python und programmiere ein Quiz

Im Moment kann das InfoLab Saar nicht besucht werden? Kein Problem. Wir bieten mit [InfoLabSaar@home](#) ein Ferienangebot, das online stattfindet. Für den Kurs nutzen wir die Videokonferenz-Software Zoom.

Wir bieten am **Donnerstag, den 23.04.** und **Freitag, den 24.04.**, jeweils von **14 – 16 Uhr** eine Einführung in die Programmiersprache [Python](#) an. Ziel ist es, in Python ein Quiz zu programmieren.



Tierquiz
Welcher Bär lebt am Nordpol?

Teilnehmen kann jeder Jugendliche **ab 13 Jahren**. Um teilzunehmen wird ein **Computer** (Laptop oder Desktop-PC) mit **Internetzugang** benötigt. Vorkenntnisse sind keine notwendig.

Anmeldungen bitte per Mail an kerstin.reese@uni-saarland.de.

Die Anzahl der Plätze ist beschränkt.

```
main.py  saved
1 def pruefe_antwort(antwort, loesung):
2     global punkte
3     loesung_gefunden = False
4     anzahl_versuche = 0
5     while not loesung_gefunden and anzahl_versuche < 3:
6         if antwort.lower() == loesung.lower():
7             print('Richtige Antwort')
```

Einladung über den Newsletter



InfoLab Saar @home: Bericht vom Python-Kurs

Am Anfang des Kurses haben sich alle Teilnehmer einen Account bei [replit](#) erstellt. Auf der Webseite kann man Python direkt im Browser programmieren. Es muss nichts auf dem eigenen Computer installiert werden.



Das Schülerlabor Informatik ist Teil der Informatik Didaktik der Universität des Saarlandes am Saarland-Informatics Campus.

Suche

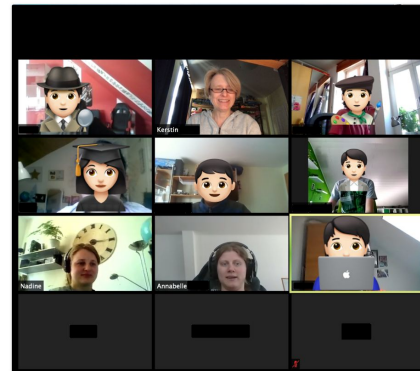
Newsletter-Anmeldung

Email Adresse

[Kontakt](#)
[Datenschutzerklärung](#)
[Impressum](#)

ARCHIVE

Februar 2021
Januar 2021
Dezember 2020
November 2020
Oktober 2020
September 2020
August 2020
Juli 2020
Juni 2020
Mai 2020



Schritt für Schritt haben wir gemeinsam die Programmierung für das Quiz begonnen. Über [Bildschirmteilen](#) konnten sich die Teilnehmer meist gegenseitig helfen, wenn jemand „alles genauso gemacht hat“ es aber nicht funktioniert. Am Ende des ersten Tages hat eine einfache Version des Quiz bei jedem funktioniert.

Das Ergebnis nach dem ersten Kurstag

```
1 def frage_stellen(frage, richtige_antwort):
2     print(frage)
3     antwort = input()
4     if (antwort == richtige_antwort):
5         print("Richtig!")
```

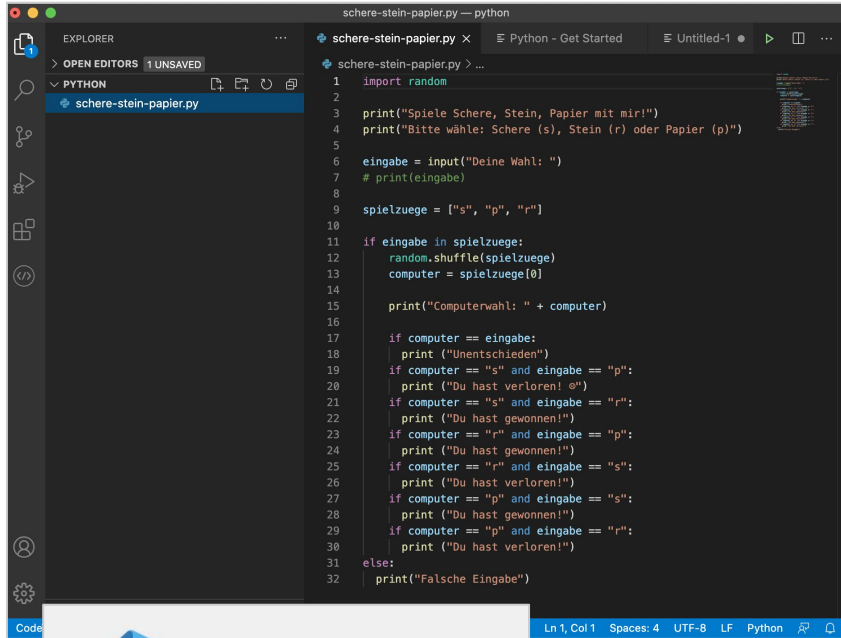
Bericht auf der Webseite

Was ist Python?

```
main.py
1  import random
2
3  print("Spiele Schere, Stein, Papier mit mir!")
4  print("Bitte wähle: Schere (s), Stein (r) oder Papier (p)")
5
6  eingabe = input("Deine Wahl: ")
7  # print(eingabe)
8
9  spielzuege = ["s", "p", "r"]
10
11 if eingabe in spielzuege:
12     random.shuffle(spielzuege)
13     computer = spielzuege[0]
14
15     print("Computerwahl: " + computer)
16
17     if computer == eingabe:
18         print ("Unentschieden")
19     if computer == "s" and eingabe == "p":
20         print ("Du hast verloren! ☹")
21     if computer == "s" and eingabe == "r":
22         print ("Du hast gewonnen!")
23     if computer == "r" and eingabe == "p":
24         print ("Du hast gewonnen!")
25     if computer == "r" and eingabe == "s":
26         print ("Du hast verloren!")
27     if computer == "p" and eingabe == "s":
28         print ("Du hast gewonnen!")
29     if computer == "p" and eingabe == "r":
30         print ("Du hast verloren!")
31 else:
32     print("Falsche Eingabe")
```

- Textuelle Programmiersprache
- Code muss getippt werden
- Schlüsselwörter sind Englisch
- Fehlermeldungen müssen kennengelernt werden
- Wer mehr lernen will, muss Dokumentation lesen.
- Zielgruppe: ab 6. Klasse

Wo kann man Python programmieren?



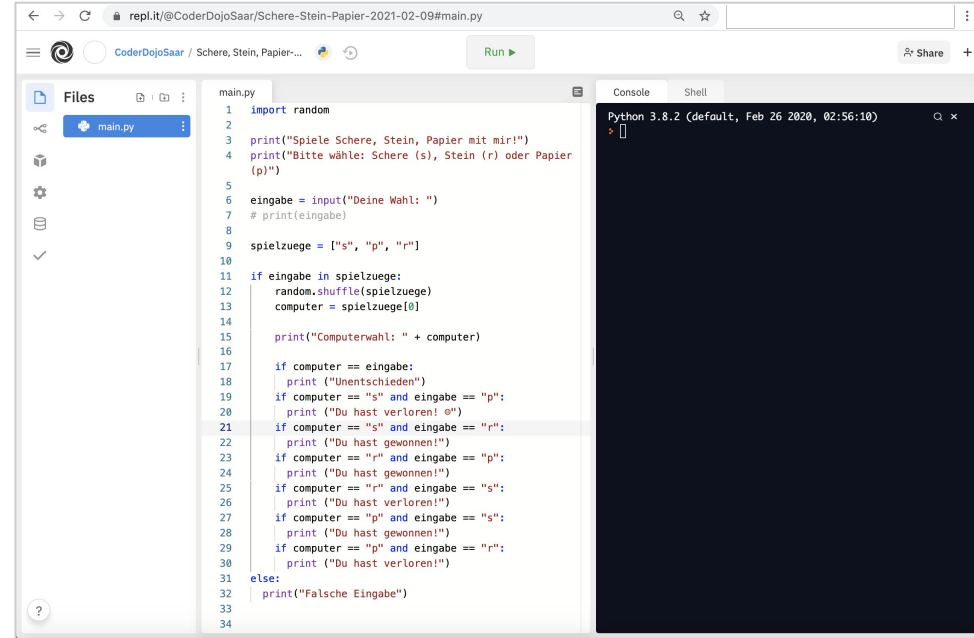
The screenshot shows the Visual Studio Code editor interface. The Explorer panel on the left shows a file named 'schere-stein-papier.py'. The main editor area displays the code for a game. The code is as follows:

```
1 import random
2
3 print("Spiele Schere, Stein, Papier mit mir!")
4 print("Bitte wähle: Schere (s), Stein (r) oder Papier (p)")
5
6 eingabe = input("Deine Wahl: ")
7 # print(eingabe)
8
9 spielzuege = ["s", "p", "r"]
10
11 if eingabe in spielzuege:
12     random.shuffle(spielzuege)
13     computer = spielzuege[0]
14
15     print("Computerwahl: " + computer)
16
17     if computer == eingabe:
18         print("Unentschieden")
19     if computer == "s" and eingabe == "p":
20         print("Du hast verloren! o")
21     if computer == "s" and eingabe == "r":
22         print("Du hast gewonnen!")
23     if computer == "r" and eingabe == "p":
24         print("Du hast gewonnen!")
25     if computer == "r" and eingabe == "s":
26         print("Du hast verloren!")
27     if computer == "p" and eingabe == "s":
28         print("Du hast gewonnen!")
29     if computer == "p" and eingabe == "r":
30         print("Du hast verloren!")
31 else:
32     print("Falsche Eingabe")
```



Visual Studio Code.app
10 Recent Documents

als Beispiel für einen lokal installierten Editor



The screenshot shows the repl.it online development environment. The browser address bar shows the URL 'repl.it/@CoderDojoSaar/Schere-Stein-Papier-2021-02-09#main.py'. The main editor area displays the same code as the Visual Studio Code editor. The console on the right shows the output of the code, which is empty.

repl.it

als Beispiel für einen Online-Entwicklungsumgebung

Interaktives H5P-Element als niederschwelliger Einstieg in Python

Python ist eine Programmiersprache, wir müssen alle Buchstaben und Leerzeichen richtig eintippen. Wenn wir Buchstaben, Punkte oder Leerzeichen vergessen, funktioniert das Programm nicht.

Unser erstes Pythonprogramm:
Wir wollen "Hallo Welt" auf der Konsole ausgeben.

Wie schreiben wir das in Python? 

`print("Hallo Welt")`

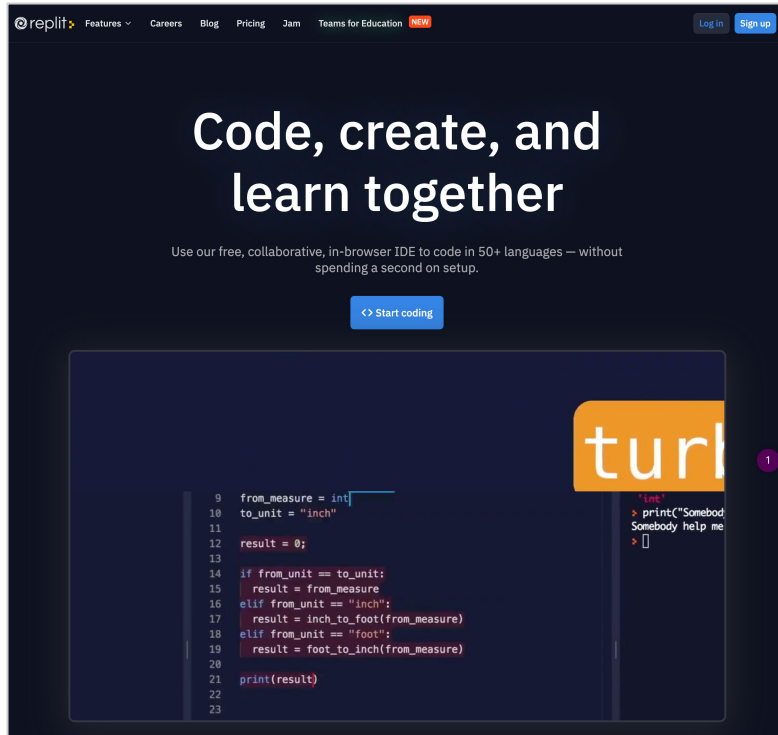
`print("Welt Hallo")`

`Hallo Welt`

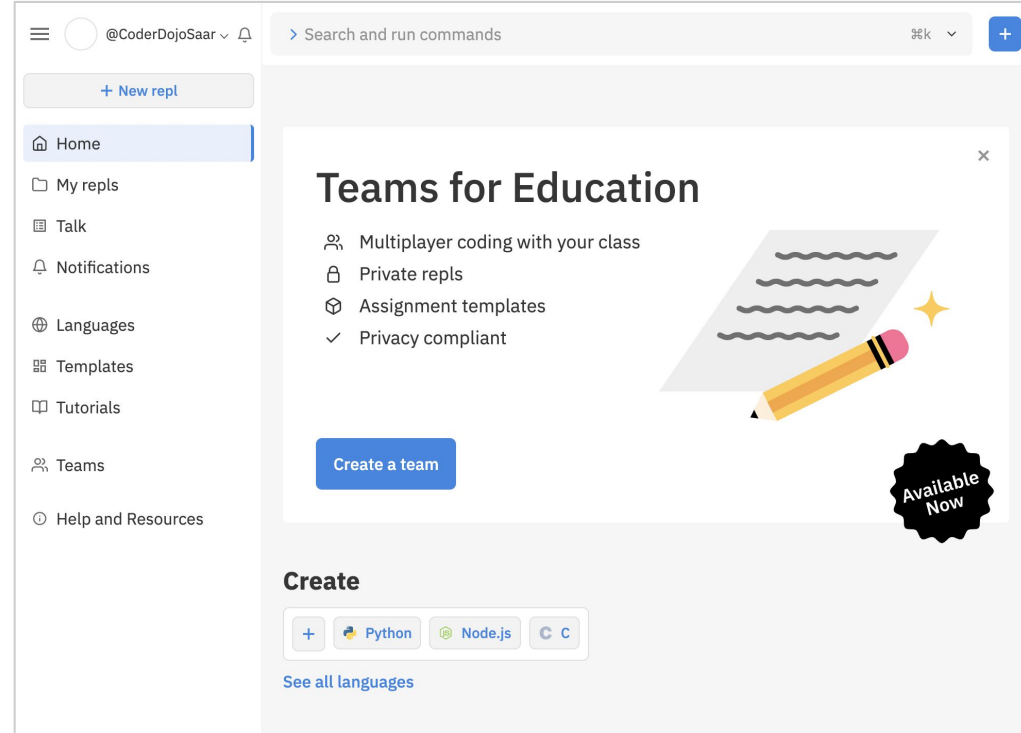
◀ 2 / 12 ▶

Reuse Embed H5P

Online-Programmierplattform repl.it



repl.it



Nach Anmeldung

Live Coding: Python

main.py

```
1  # Einlesen des Namens
2  print("Hallo wie heißt Du?")
3  name = input()
4
5  # Einlesen des Alters
6  print("Wie alt bist Du?")
7  alter = input()
8
9  # Die Ausgabe zusammenbauen
10 text = name + " ist "
11 text = text + str(alter) + " Jahre alt."
12
13 # Die Ausgabe
14 print(text)
```

Wenn man Python online unterrichtet

- Man sollte in den Code der Jugendlichen schauen können
- Gruppengröße ca. 12 Jugendliche ab 12 Jahren. Jede Frage bremst die ganze Gruppe aus.
- Erfolgreiches Anmelden muss nur abgefragt werden.
- Navigieren im Internet ist meist kein Problem
- Springen im Programmcode hängt Jugendliche ab. Das Programm lieber linear von oben nach unten entwickeln.
- Fehlerfrei tippen ist für die Jugendlichen ungewohnt. Es wird Fehler geben. Es werden alle Kommentare mitgetippt. Das kann dauern.
- Das Beispielprojekt sollte gut ausgesucht sein, damit die Jugendlichen den Überblick behalten und Konzepte sichtbar werden, z.B. EVA-Prinzip.

Fazit: Python lässt sich online gut unterrichten. Am besten ist eine zweite Person dabei, die den Code der Jugendlichen live verfolgt.

Weitere Online-Kurse des InfoLab Saar

Was haben wir noch online gemacht

Künstliche Intelligenz:

- Entscheidungsbäume
- Fairness von Algorithmen
- Erklärbare KI (10.03.2021)

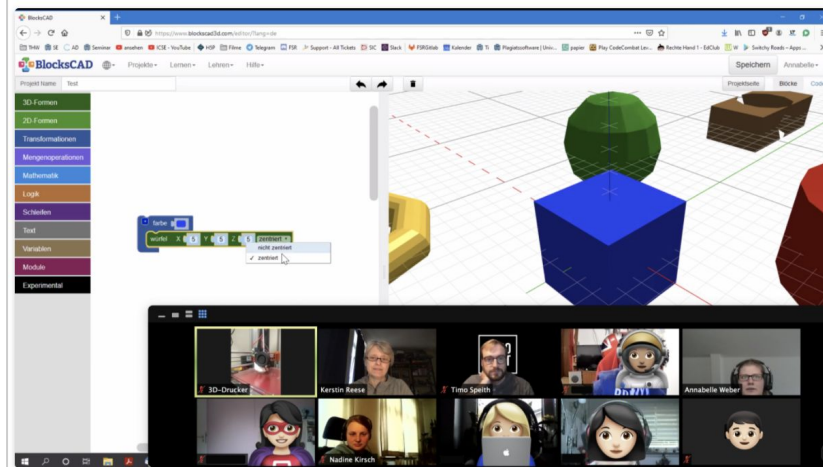
Informatik-AG:

- Phaser.io selber unterrichtet



CoderDojo Saar

- Unity
- Godot



Demnächst

- Minecraft-Plugins mit Java

infolab.cs.uni-saarland.de

