

Build your Robot

Handout 4.1 Aufgabenblatt 2. Olivia Schwarz und Mirco Leali, Mai 2022

«Build your Bot»

Start

- Bildet Zweiergruppen.
- Alle Schüler:innen loggen sich mit ihrem Account auf P5js ein.

Einführungen

- Die Augen verfolgen die Maus (mouseX, mouseY)
- Der Mund bewegt sich beim Sprechen (mic.Level = mic.getLevel())

«Build your Bot 2.0»

- Die Tandems bilden zwei Teams mit jeweils drei Bots (1 Bot pro Tandem). Beide Teams erarbeiten mit einer vorgegebenen Situation (Wald oder Stadt) eine Geschichte, in der die drei Bots unterschiedlich reagieren.
- Die Adjektive der letzter Woche helfen, drei verschiedene Bots zu gestalten und den dazugehörigen Charakter zu entwickeln. Beantwortet dazu im Padlet die Fragen.
- Öffnet im Tandem den vorbereiteten Code. Dupliziert und speichert diesen:
File -> Duplicate
File -> Save
- Auf Teams haben wir euch einen Ordner gesendet mit sechs Bildern, je drei pro Situation. Importiert das gewünschte Bild in die p5js Umgebung (function preload())
- Unter «Sketch Files», links oben auf P5js findet ihr die Text-Dateien. In diese Dateien könnt ihr die Fragen des Bots schreiben und so die Interaktion zwischen Menschen und Bot gestalten.
Ihr könnt pro Textfile mehrere Varianten einer Frage schreiben, aber jede Variante muss auf einer neuen Zeile beginnen. Jeweils eine der Varianten wird dann zufällig ausgewählt und angezeigt werden.

Build your Robot

Handout 4.2 Aufgabenblatt 2. Mai 2022

p5.js

<https://p5js.org/>

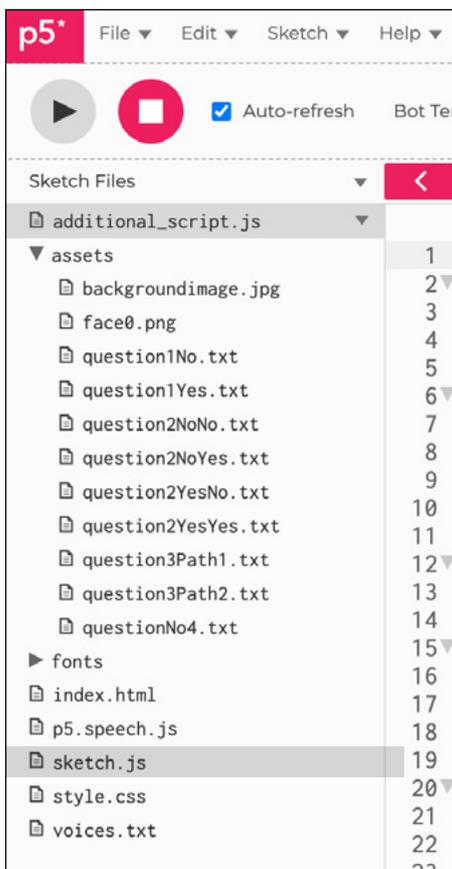
Vorbereiteter Code für Bot

Unter der Adresse <https://editor.p5js.org/CreativeCodingZHdK/sketches/GKaSGrPv6> ist das vorbereitete Code-Template für den Bot abrufbar.

Unter «Sketch Files», links oben auf P5js, befindet sich das Dokument «sketch.js». Um den umfangreichen Code übersichtlicher zu halten wurde ein Teil des Codes zusätzlich im Dokument «additional_script.js» gespeichert.

Zwei Schritte sind nötig, um den Bot fertig zu stellen:

- 1) Öffne den Code deines fertigen Bots. Kopiere den Code, der sich innerhalb der Funktion `function draw()` befindet.
Öffne jetzt das Bot-Template, öffne dann das Dokument «sketch.js» und setze deinen Code hier innerhalb der Funktion `function draw()` ein.
- 2) Ebenfalls unter «Sketch Files» im Folder «assets» findet ihr 9 Text-Dateien mit den Namen «question... .txt». In diese 9 Dateien könnt ihr eure vorbereiteten Fragen des Bots einsetzen.
(Siehe auch folgende Seite).



Build your Robot

Handout 4.3 Aufgabenblatt 2. Mai 2022

p5.js

<https://p5js.org/>

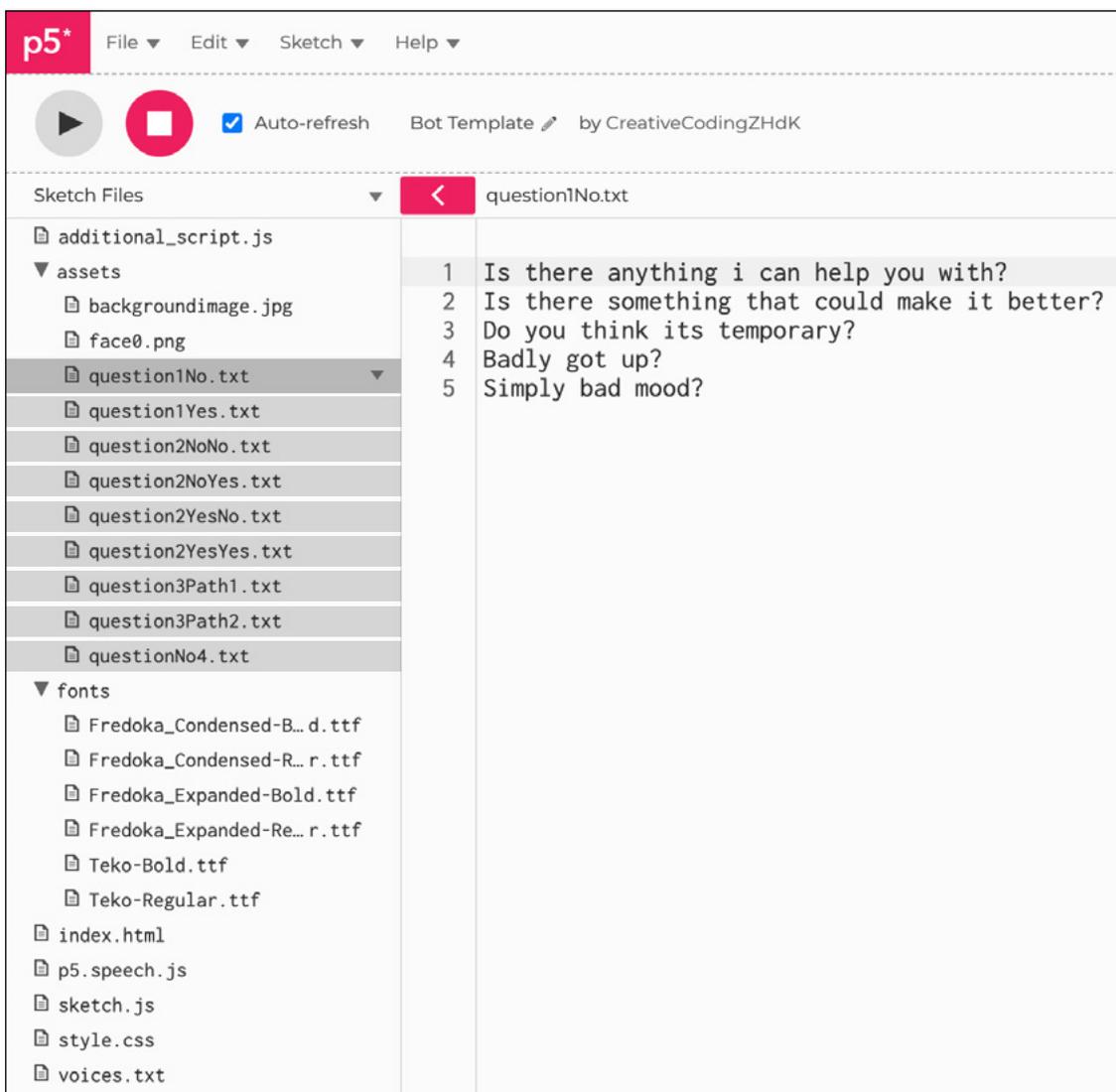
Hypertext Text-Inhalte einfüllen

Unter «Sketch Files», links oben auf P5js findet ihr die 9 Text-Dateien im Folder «assets»:

- question1No.txt
- question1Yes.txt
- question2NoNo.txt
- question2NoYes.txt
- question2YesNo.txt
- question2YesYes.txt
- question3Path1.txt
- question3Path2.txt
- questionNo4.txt

In diese 9 Dateien könnt ihr die Fragen des Bots einsetzen und so die Interaktion zwischen Menschen und Bot gestalten.

Ihr könnt pro Textfile mehrere Varianten einer Frage schreiben, aber jede Variante muss auf einer neuen Zeile beginnen. Jeweils eine der Varianten wird dann zufällig ausgewählt und angezeigt werden.



Build your Robot

Handout 4.4 Aufgabenblatt 2. Mai 2022



<https://p5js.org/>

Einblick in den Code

Der Code des Bot-Templates möchte die Logik möglichst nachvollziehbar abbilden und ist deshalb sehr ausführlich verfasst. Das Template ist so gedacht, dass der Entwurf des eigenen Bots und die vorbereiteten Texte einfach einkopiert werden können, ohne den vollständigen Code verstehen zu müssen.

Für diejenigen Schüler:innen, die das Template-Script verstehen wollen sind hier die wesentlichen Funktionen kurz beschrieben.

```
let q1No, q1Yes;

function loadTextfiles() {
  q1No = loadStrings("assets/question1No.txt");
  q1Yes = loadStrings("assets/question1Yes.txt");
}

function botQuestion() {
  if (answer == „Yes“) {
    let r = floor(random(0, q1Yes.length));
    question = q1Yes[r];
  }
  if (answer == „No“) {
    let r = floor(random(0, q1No.length));
    question = q1No[r];
  }
}

function testAnswer() {
  if (mouseY > 550 && mouseX < width / 2 &&
      mouseY > 550)
    answer = answer + „No“;
}
```

function testAnswer()

`answer = answer + „No“;`

Je nach Antwort wird am Ende des Strings „answer“ ein „Yes“ oder „No“ hinzugefügt.

let q1No;

Erstellt die Variablen, in die die Fragen abgefüllt werden.

function loadTextfiles()

`q1No = loadStrings(„assets/question1No.txt“)`

Füllt den Inhalt des Textfiles „question1No.txt“ in die Variable «q1No» ein.

function botQuestion()

Wählt zufällig eine Zeile des Textes aus und füllt diese Zeile in die Variable „question“ ab.

if (answer == „No“)

Prüft ob „Yes“ oder auf „No“ geklickt wurde.

q1Yes.length

Zählt die Zeilen des Texts.

let r = floor(random(0, q1No.length));

r ist eine zufällige ausgewählte Zeile des Texts.

question = q1No[r];

Füllt die ausgewählte Zeile in die Variable «question» ein