Erstellt nach der DokuWiki-Syntax aus dem bereitgestellten Handbuch

P5.js

Einführung

p5.js ist eine JavaScript-Bibliothek für kreative Programmierung. Sie wurde so entwickelt, dass visuelle, interaktive und audiovisuelle Projekte leichter umgesetzt werden können. Besonders beliebt ist p5.js in den Bereichen Kunst, Design, Lehre und Prototyping.

Die Bibliothek baut auf JavaScript auf, vereinfacht aber viele typische Aufgaben wie das Zeichnen von Formen, das Arbeiten mit Farben, Animationen, Maus- und Tastatureingaben sowie Medien wie Bild, Ton und Video.

Hintergrund

p5.js ist von der Programmiersprache und Entwicklungsumgebung ‘‘Processing’’ inspiriert. Processing wurde geschaffen, um Künstlerinnen, Künstlern, Designerinnen, Designern und Einsteigerinnen und Einsteigern einen einfachen Zugang zur Programmierung zu ermöglichen.

Die Idee hinter p5.js ist, diese Prinzipien in den Webbrowser zu bringen. Dadurch können Programme direkt im Browser ausgeführt werden, ohne dass eine separate Installation einer Entwicklungsumgebung notwendig ist. Das macht p5.js besonders geeignet für:

- interaktive Webseiten
- digitale Kunst
- Animationen
- Lernprojekte
- kreative Experimente mit Code, etc.

Grundprinzipien

1. Niedrige Einstiegshürde

p5.js soll leicht zu erlernen sein. Viele Funktionen sind bewusst einfach benannt und direkt nutzbar. Dadurch können schon mit wenig Code erste sichtbare Ergebnisse erzeugt werden.

Beispielsweise reicht oft schon eine kleine Struktur aus, um eine Zeichenfläche zu erzeugen und etwas darauf darzustellen.

```
function setup() {
  createCanvas(400, 400);
}


function draw() {
background(220);
ellipse(200, 200, 100, 100);
}
```

2. Kreatives Programmieren

Im Mittelpunkt steht nicht nur die technische Lösung eines Problems, sondern der kreative Ausdruck. p5.js wird häufig verwendet, um mit Formen, Farben, Bewegung, Zufall und Interaktion zu experimentieren.

Mögliche Fragen in p5.js-Projekten sind zum Beispiel:

- Wie kann Bewegung interessant wirken?
- Wie reagieren Formen auf Maus oder Tastatur?
- Wie lassen sich Daten visuell darstellen?
- Wie können Klang und Bild verbunden werden?

3. Direkte visuelle Rückmeldung

Ein wichtiges Prinzip von p5.js ist die unmittelbare Rückmeldung. Änderungen am Code führen schnell zu sichtbaren Veränderungen auf der Zeichenfläche. Dadurch eignet sich die Bibliothek sehr gut zum Lernen durch Ausprobieren.

Dieses Vorgehen unterstützt ein experimentelles Arbeiten:

- Idee formulieren
- Code anpassen
- Ergebnis ansehen
- weiter verbessern

4. Der Zeichenzyklus

Viele p5.js-Skizzen beruhen auf zwei zentralen Funktionen:

1. ‘‘setup()’’ wird genau einmal zu Beginn ausgeführt
2. ‘‘draw()’’ wird danach fortlaufend wiederholt

Dieses Modell macht Animationen und interaktive Anwendungen besonders einfach.

```
function setup() {
  createCanvas(600, 400);
}


function draw() {
background(30);
circle(mouseX, mouseY, 40);
}
```

Hier wird in jedem Durchlauf der Hintergrund neu gezeichnet und ein Kreis an der aktuellen Mausposition dargestellt.

5. Interaktion

p5.js ist stark auf Interaktivität ausgelegt. Maus, Tastatur und andere Eingaben können direkt verarbeitet werden. Dadurch lassen sich Anwendungen erstellen, die auf Benutzeraktionen reagieren.

Beispiele für Interaktion sind:

- Zeichnen mit der Maus
- Reaktion auf Tastendruck
- Steuerung von Objekten
- audiovisuelle Installationen

```
function setup() {
  createCanvas(400, 400);
}


function draw() {
background(240);
if (mouseIsPressed) {
fill(255, 0, 0);
} else {
fill(0, 100, 255);
}
rect(150, 150, 100, 100);
}
```

6. Arbeiten mit dem Canvas

Die sichtbare Fläche in p5.js wird meist mit ‘‘createCanvas()’’ erzeugt. Auf diesem Canvas können Formen, Linien, Farben, Texte und Bilder dargestellt werden.

Wichtige Grundfunktionen sind zum Beispiel:

- ‘‘background()’’ für den Hintergrund
- ‘‘fill()’’ für Füllfarben
- ‘‘stroke()’’ für Linienfarben
- ‘‘line()’’, ‘‘rect()’’, ‘‘circle()’’ und ‘‘ellipse()’’ für Formen
- ‘‘text()’’ für Schrift

Diese klare Struktur unterstützt das gestalterische Arbeiten mit grafischen Elementen.

Zentrale Konzepte

Koordinatensystem

p5.js verwendet in der Regel ein zweidimensionales Koordinatensystem. Der Ursprung liegt oben links auf der Zeichenfläche.

- ‘‘x’’ wächst nach rechts
- ‘‘y’’ wächst nach unten

Dieses System ist wichtig, um Objekte gezielt zu platzieren und zu bewegen.

Farbe

Farben können in p5.js auf verschiedene Weise beschrieben werden, häufig über RGB-Werte. Damit lassen sich sehr gezielt visuelle Stimmungen erzeugen.

```
fill(255, 0, 0);
circle(100, 100, 50);
```

Obiger Code zeichnet einen roten Kreis.

Bewegung

Bewegung entsteht meist dadurch, dass Positionen oder andere Eigenschaften in jedem Durchlauf von ‘‘draw()’’ verändert werden.

```
let x = 0;


function setup() {
createCanvas(400, 200);
}

function draw() {
background(255);
circle(x, 100, 30);
x = x + 2;
}
```

So bewegt sich ein Kreis schrittweise nach rechts.

Zufall

Zufall ist ein zentrales Mittel im kreativen Programmieren. Mit zufälligen Werten können organische, überraschende oder generative Ergebnisse erzeugt werden.

Typische Einsatzbereiche sind:

- zufällige Positionen
- wechselnde Farben
- generative Muster
- simulierte natürliche Prozesse

Wiederholung und Muster

Durch Schleifen und systematische Veränderungen lassen sich komplexe Muster aus einfachen Regeln erzeugen. Das ist ein wesentliches Prinzip algorithmischer Gestaltung.

```
function setup() {
  createCanvas(400, 400);
  background(255);
```

```
for (let x = 20; x < width; x += 40) {
for (let y = 20; y < height; y += 40) {
circle(x, y, 20);
}
}
}
```

Pädagogische Bedeutung

p5.js wird häufig in der Lehre eingesetzt, weil es Programmierkonzepte anschaulich vermittelt. Lernende sehen schnell, wie sich Logik, Variablen, Bedingungen, Schleifen und Funktionen auf ein sichtbares Ergebnis auswirken.

Besonders hilfreich ist p5.js für das Verständnis von:

- Variablen und Zuständen
- Funktionen
- Bedingungen
- Schleifen
- Ereignissen
- algorithmischem Denken

Typische Anwendungsbereiche

Kunst und Design

p5.js eignet sich hervorragend für generative Kunst, visuelle Experimente und interaktive Installationen. Gestalterische Ideen können schnell als Prototyp umgesetzt werden.

Datenvisualisierung

Daten können mit Formen, Farben und Bewegung sichtbar gemacht werden. Dadurch lassen sich Zusammenhänge verständlich und ansprechend darstellen.

Interaktive Medien

Webbasierte Anwendungen, Lernumgebungen oder kreative Tools profitieren von der einfachen Integration interaktiver Elemente.

Prototyping

Ideen für Benutzeroberflächen, Animationen oder medienbasierte Experimente können schnell getestet werden, bevor eine umfassendere technische Umsetzung erfolgt.

Vorteile von p5.js

1. einfacher Einstieg in kreative Programmierung

2. direkte Ausführung im Browser
3. schnelle sichtbare Ergebnisse
4. gute Eignung für Lehre und Workshops
5. starke Ausrichtung auf Interaktion und Gestaltung
6. große Nähe zu künstlerischen und experimentellen Arbeitsweisen

Grenzen von p5.js

Trotz seiner Stärken ist p5.js nicht für jede Aufgabe die beste Wahl. Für sehr große, hochoptimierte oder spezialisierte Anwendungen können andere Werkzeuge besser geeignet sein.

Mögliche Grenzen sind:

1. geringere Performance bei sehr komplexen Projekten
3. weniger geeignet für große Softwarearchitekturen
4. Fokus liegt eher auf Kreativität und Prototyping als auf klassischen Geschäftsanwendungen

Fazit

p5.js ist ein zugängliches und leistungsfähiges Werkzeug für kreative Programmierung im Browser. Die Bibliothek verbindet technische Grundlagen mit gestalterischer Freiheit und fördert experimentelles Lernen. Gerade für den Einstieg in visuelles und interaktives Programmieren bietet p5.js einen besonders anschaulichen und motivierenden Zugang.

Siehe auch

- Offizielle p5.js-Webseite
- p5.js Web Editor
- Processing

From:
https://wiki.ct-lab.info/ - **Creative Technologies Lab | dokuWiki**

Permanent link:
https://wiki.ct-lab.info/doku.php/extras:codikon:p5js:einfuehrung_in_p5js?rev=1773992744

Last update: **2026/03/20 07:45**