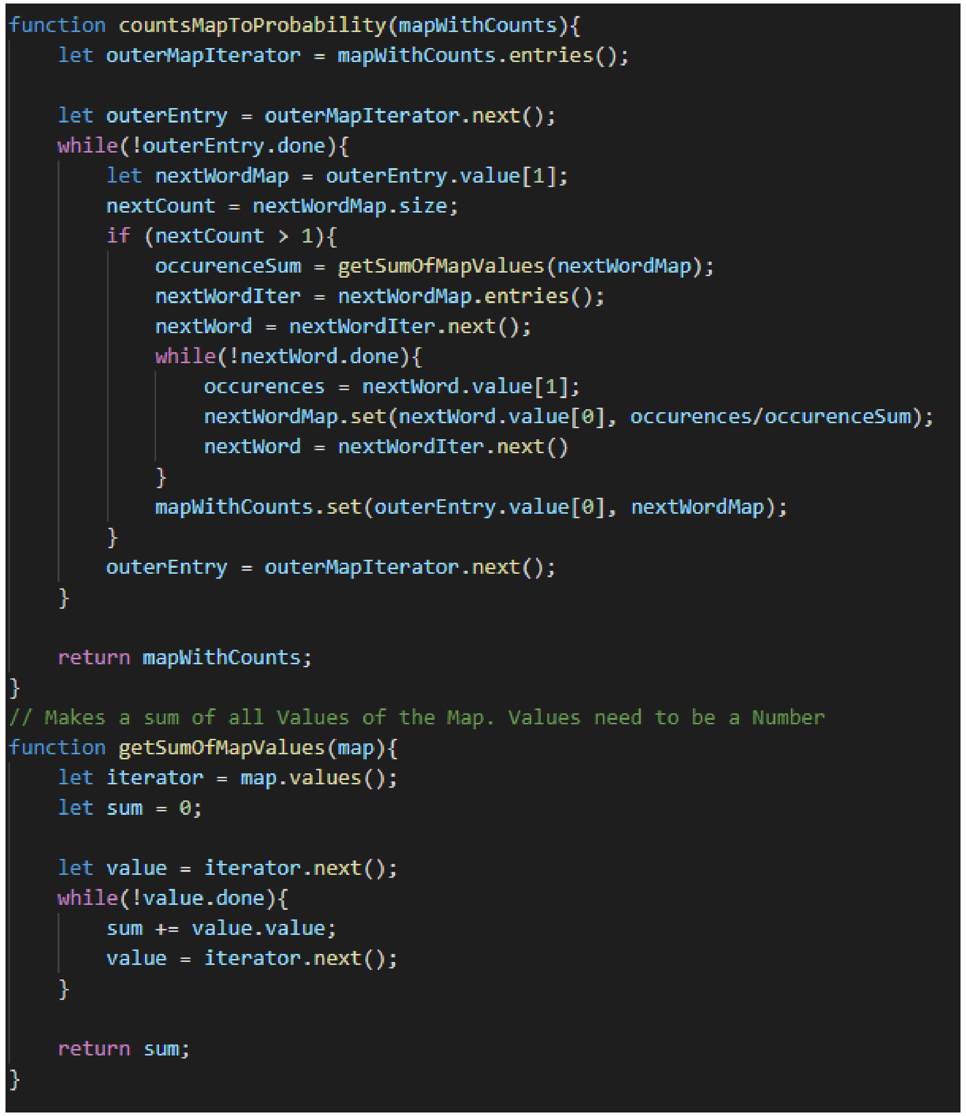
MATH2 Woche 2
Markov Chain
Simon Gisler

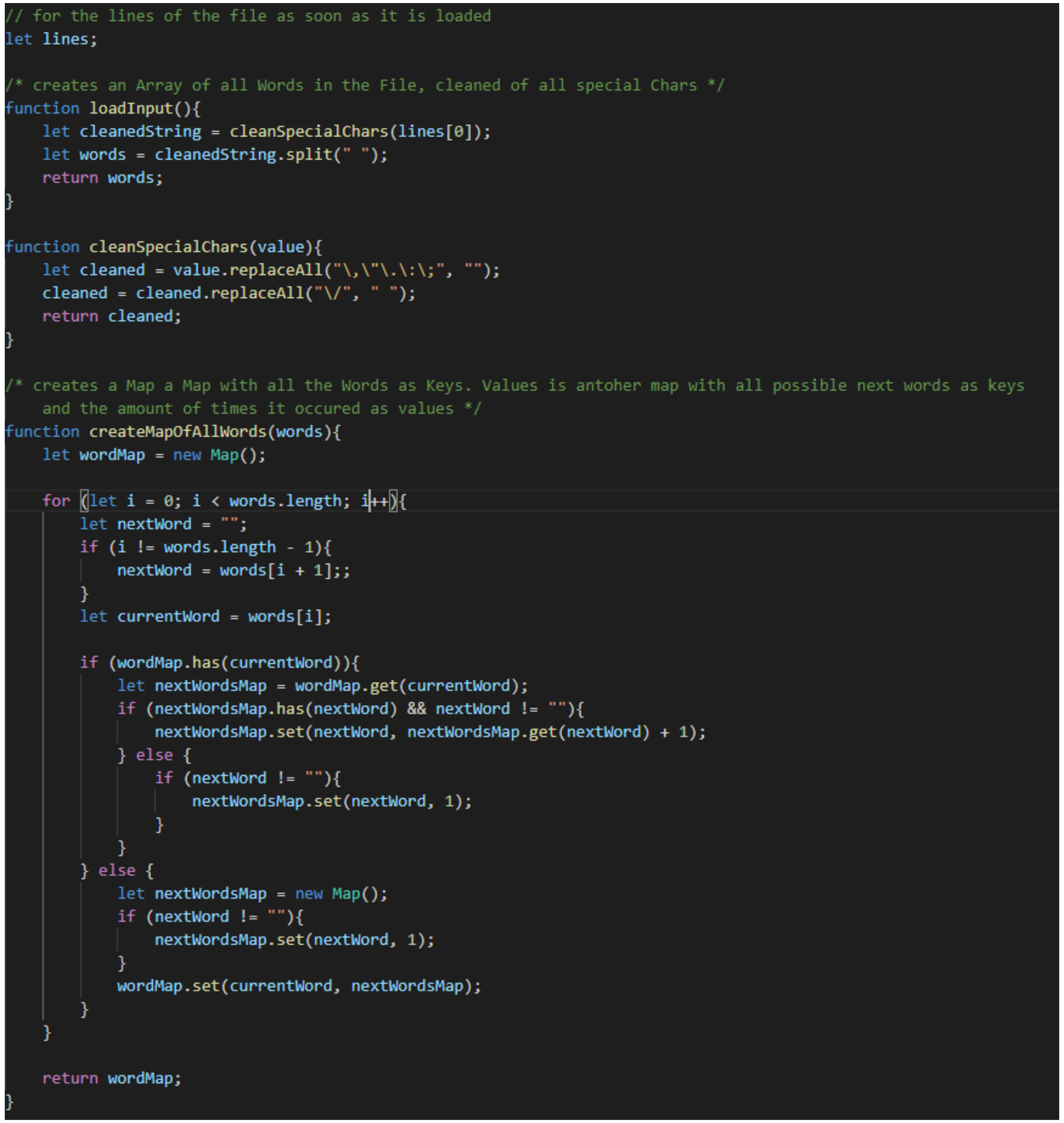
Texte

- Modulbeschreibungen Digital Ideation 1.Jahr
- «Kritik der Gegenwart» von Hermann Bahr
  - https://www.projekt-gutenberg.org/bahr/kritik/kritik.html

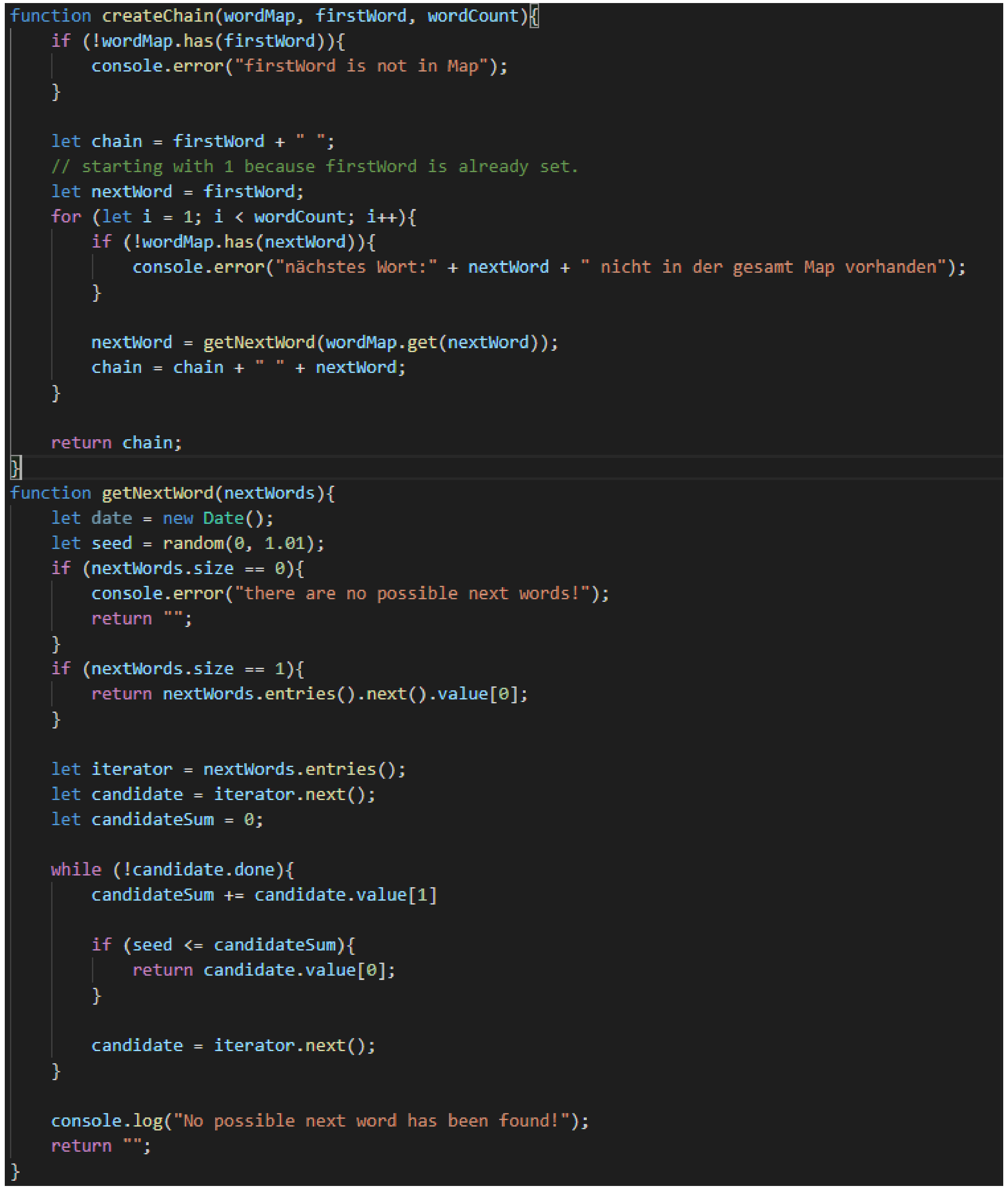
Code



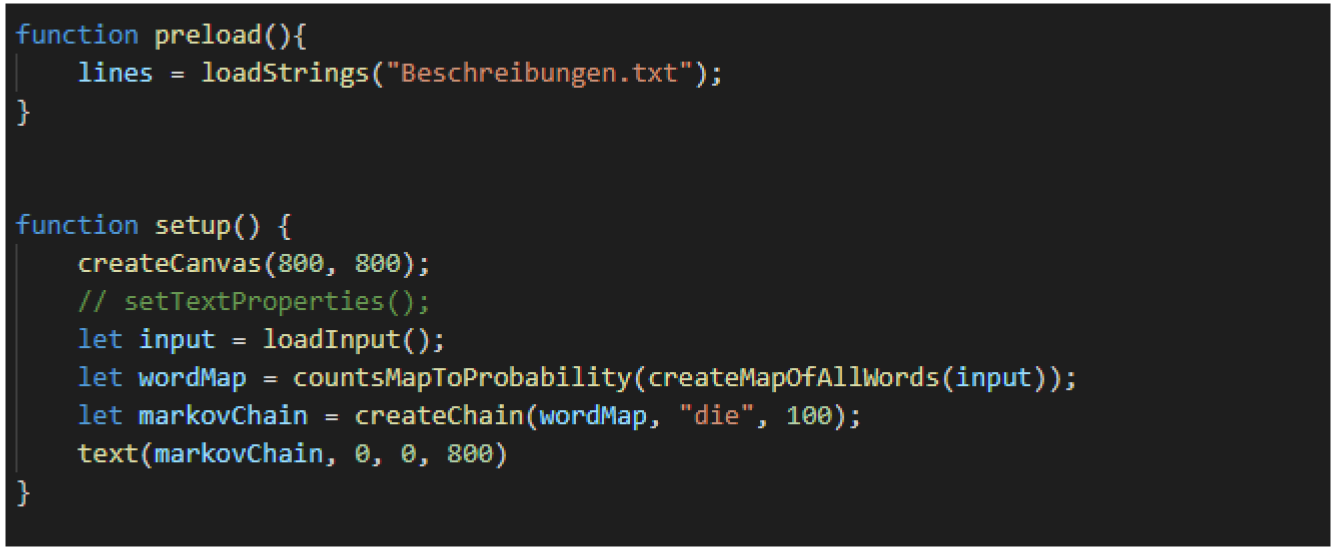
```
function countsMapToProbability(mapWithCounts){
    let outerMapIterator = mapWithCounts.entries();

    let outerEntry = outerMapIterator.next();
    while(!outerEntry.done){
        let nextWordMap = outerEntry.value[1];
        nextCount = nextWordMap.size;
        if (nextCount > 1){
            occurenceSum = getSumOfMapValues(nextWordMap);
            nextWordIter = nextWordMap.entries();
            nextWord = nextWordIter.next();
            while(!nextWord.done){
                occurences = nextWord.value[1];
                nextWordMap.set(nextWord.value[0], occurences/occurenceSum);
                nextWord = nextWordIter.next()
            }
            mapWithCounts.set(outerEntry.value[0], nextWordMap);
        }
        outerEntry = outerMapIterator.next();
    }

    return mapWithCounts;
}
// Makes a sum of all Values of the Map. Values need to be a Number
function getSumOfMapValues(map){
    let iterator = map.values();
    let sum = 0;

    let value = iterator.next();
    while(!value.done){
        sum += value.value;
        value = iterator.next();
    }

    return sum;
}
```



```
// for the lines of the file as soon as it is loaded
let lines;

/* creates an Array of all Words in the File, cleaned of all special Chars */
function loadInput(){
    let cleanedString = cleanSpecialChars(lines[0]);
    let words = cleanedString.split(" ");
    return words;
}

function cleanSpecialChars(value){
    let cleaned = value.replaceAll("\,\"\.\:\;", "");
    cleaned = cleaned.replaceAll("\/", " ");
    return cleaned;
}

/* creates a Map a Map with all the Words as Keys. Values is antoher map with all possible next words as keys
    and the amount of times it occured as values */
function createMapOfAllWords(words){
    let wordMap = new Map();

    for (let i = 0; i < words.length; i++){
        let nextWord = "";
        if (i != words.length - 1){
            nextWord = words[i + 1];;
        }
        let currentWord = words[i];

        if (wordMap.has(currentWord)){
            let nextWordsMap = wordMap.get(currentWord);
            if (nextWordsMap.has(nextWord) && nextWord != ""){
                nextWordsMap.set(nextWord, nextWordsMap.get(nextWord) + 1);
            } else {
                if (nextWord != ""){
                    nextWordsMap.set(nextWord, 1);
                }
            }
        } else {
            let nextWordsMap = new Map();
            if (nextWord != ""){
                nextWordsMap.set(nextWord, 1);
            }
            wordMap.set(currentWord, nextWordsMap);
        }
    }

    return wordMap;
}
```



```
function createChain(wordMap, firstWord, wordCount){
    if (!wordMap.has(firstWord)){
        console.error("firstWord is not in Map");
    }

    let chain = firstWord + " ";
    // starting with 1 because firstWord is already set.
    let nextWord = firstWord;
    for (let i = 1; i < wordCount; i++){
        if (!wordMap.has(nextWord)){
            console.error("nächstes Wort:" + nextWord + " nicht in der gesamt Map vorhanden");
        }

        nextWord = getNextWord(wordMap.get(nextWord));
        chain = chain + " " + nextWord;
    }

    return chain;
}
function getNextWord(nextWords){
    let date = new Date();
    let seed = random(0, 1.01);
    if (nextWords.size == 0){
        console.error("there are no possible next words!");
        return "";
    }
    if (nextWords.size == 1){
        return nextWords.entries().next().value[0];
    }

    let iterator = nextWords.entries();
    let candidate = iterator.next();
    let candidateSum = 0;

    while (!candidate.done){
        candidateSum += candidate.value[1]

        if (seed <= candidateSum){
            return candidate.value[0];
        }

        candidate = iterator.next();
    }

    console.log("No possible next word has been found!");
    return "";
}
```



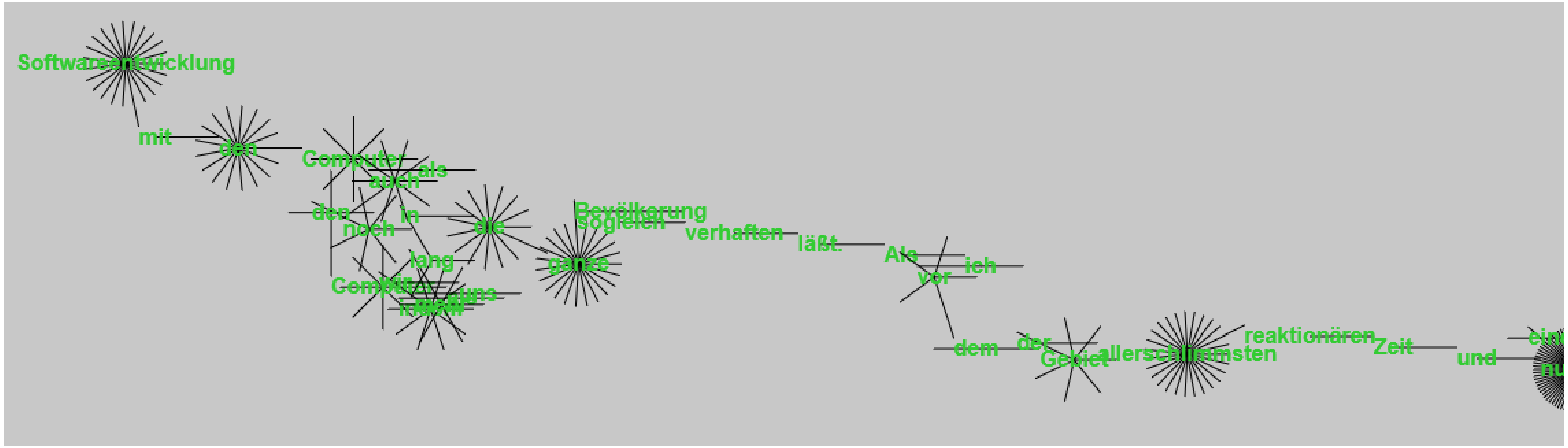
```
function preload(){
    lines = loadStrings("Beschreibungen.txt");
}


function setup() {
    createCanvas(800, 800);
    // setTextProperties();
    let input = loadInput();
    let wordMap = countsMapToProbability(createMapOfAllWords(input));
    let markovChain = createChain(wordMap, "die", 100);
    text(markovChain, 0, 0, 800)
}
```

Beispiel Texte

die Softwareentwicklung mit ein Tracking-Konzept erarbeitet. Die Studierenden lernen die Grundstimmung seines Lebens und: wie Computerspiele und Prototypen (Geräte, Anwendungen und Datentransfer (z.B. JSON). Überblick über technologischen, ökonomischen und Bedürfnisse in Processing und Computerspieleengines aufgebaut sind, wie mir ja noch

die Seelen frißt, daß sie persönlich aus meiner Studentenzeit, an Brandes: »Ueberhaupt gibt es also für Flächenmuster (Tesselation), Agenten, Formen und -Storyboards aufgrund von Anforderungen (Requirements). Selbständige Recherche von Computern, sowie eine unermessliche Formenvielfalt hervorbringt. Die Studierenden interaktive Objekte zu

Darstellungs Experiment


Softwareentwicklung
mit
den
Computer
als
auch
den
in
noch
die
lang
Bevölkerung
sogleich
verhaften
läßt.
ganze
Als
ich
vor
uns
dem
der
Gebiet
allerschlimmsten
reaktionären
Zeit
und