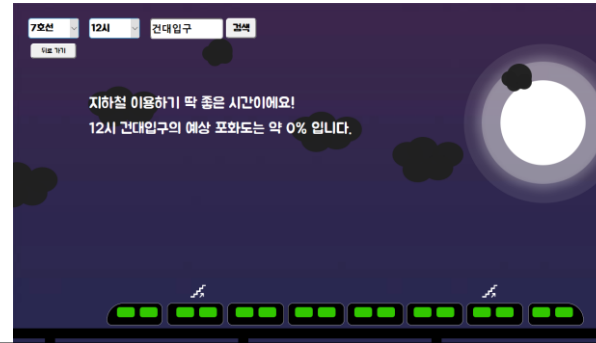


02

배열과 텍스트 데이터 파일의 활용





텍스트 데이터 파일(csv)을 활용하여 만든 지하철 포화도 예측 서비스 <앉을자리>



버스 내 혼잡도에 대한 데이터 시각화

미세먼지 농도 데이터를 활용한 미디어 아트 Webcam 영상



텍스트 데이터 파일

CSV
파일

JSON
파일

CSV 파일

(Comma-Separated Values)

몇가지 필드를 쉼표로 구분한 텍스트 파일

테이블 구조의 엑셀문서 파일에서 텍스트 파일 형식으로 저장

animals.csv

ID	CLASS	NAME
0	포유류	개
1	조류	앵무새
2	어류	금붕어

.....▶ header

```
loadTable(filename);
```

```
loadTable(filename, 'csv', 'header');
```



CSV 파일 데이터 출력해보기

```
var table;
function preload() {
  table = loadTable('animals.csv', 'csv', 'header'); //데이터를 외부에서 로드
}
function setup() {
  print(table.getColumn('name'));
  //["dog", "parrot", "goldfish"]
  //중첩반복문을 활용하여 열과 행을 순환하도록 함
  for (var r = 0; r < table.getRowCount(); r++)
    for (var c = 0; c < table.getColumnCount(); c++) {
      print(table.getString(r, c));
    }
}
function draw() {
}
```



JSON 파일

(JavaScript Object Notation)

네트워크를 통해 데이터를 주고받을 때
사용할 수 있는 데이터 형식

- **Name: Value** 형태가 한쌍으로 이루어짐
- 객체(Object)와 배열[]로 구성

```
{                                     paintings.json
  "title": "Golconda",
  "artist": "Rene Magritte",
  "year": 1953,
  "movement": ["초현실주의", "다다이즘"]
}
```



JSON 파일

(JavaScript Object Notation)

JSON 파일

- loadJSON(filename)
함수 활용하여 불러오기

예) paintings.json

JSON 데이터

- 외부파일로 정의하고 불러오기
- 코드 안에서 정의하기

JSON 파일 데이터 활용하여 그리기

```
let flower;  
function preload() {  
  flower = loadJSON("flower.json");  
}  
  
function setup() {  
  createCanvas(400, 400);  
}  
  
function draw() {  
  background(0);  
  fill(flower.red, flower.green, flower.blue);  
  text(flower.name, 10, 50);  
}
```

데이터를
외부에서 로드

flower.json
<pre>{ "name": "sunflower", "red": 255, "green": 255, "blue": 0 }</pre>

JSON 데이터 코드 안에서 정의하기

```
let flower;  
function setup() {  
  createCanvas(400, 400);  
  
  flower = {  
    name: "sunflower",  
    red: 255,  
    green: 255,  
    blue: 0  
  }  
  
}  
function draw() {  
  background(0);  
  fill(flower.red, flower.green, flower.blue);  
  text(flower.name, 10, 50);  
}
```

데이터를 sketch.js 안에 직접 정의