

웹 프로그래밍 풀스택 5기

대용량 파일을 읽어보자

Node.js 스트리밍



박지혜

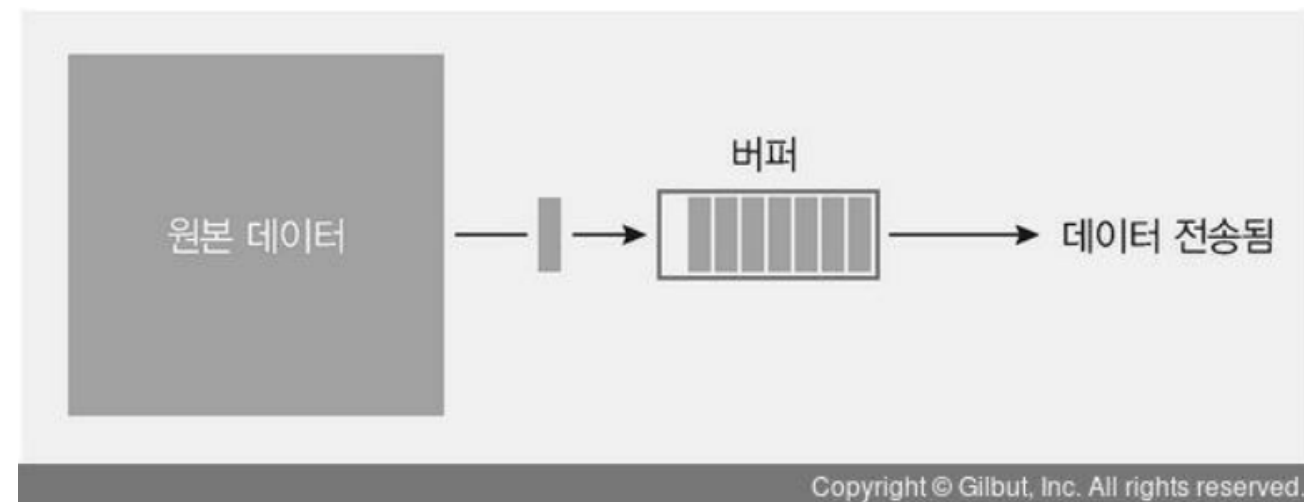
목차

- 파일 입출력 - 버퍼와 스트림
- Node.js에서의 스트리밍
- 비디오 스트리밍 예시

파일 입출력 - 버퍼와 스트림

✓ 버퍼 (Buffer)

파일의 크기만큼 메모리에 공간을 마련하여 데이터를 저장하는 방식

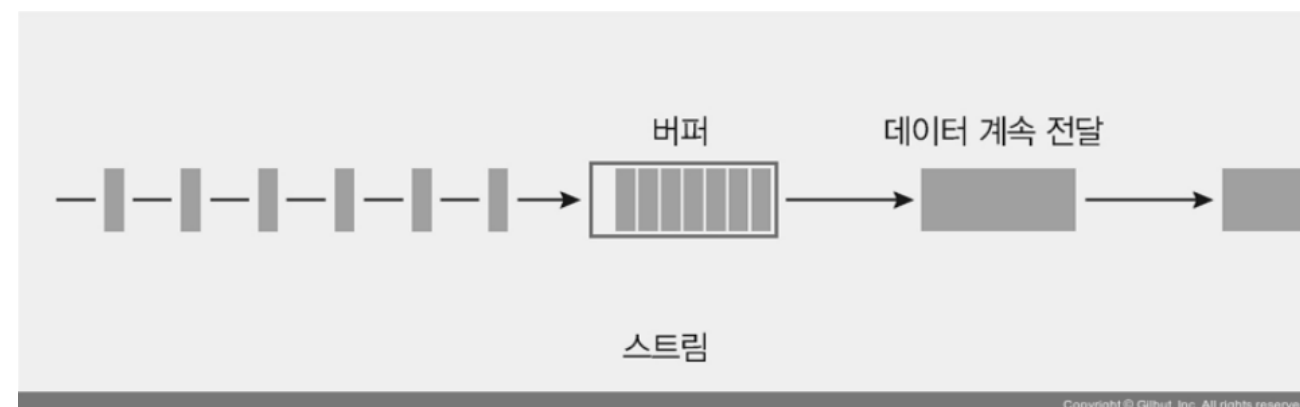


>> 데이터를 한번에 처리

>> 파일이 큰 경우 필요한 메모리 공간도 크다

✔ 스트림 (Stream)

데이터를 일정한 크기로 나눠서 연속으로 처리하는 방식



>> 나누어진 데이터 조각 : 청크(chunk)

>> 데이터의 크기가 클 때 주로 사용

Node.js에서의 스트리밍

✓ 파일 읽기 – createReadStream

```
fs.createReadStream(path[, options])
```



```
const fs = require("fs");

const readStream = fs.createReadStream("input.txt", { highWaterMark: 2 });

readStream.on("data", function (fileData) {
  console.log(fileData.toString());
});
```

✓ 파일 쓰기 – createWriteStream

```
fs.createWriteStream(path[, options])
```



```
const fs = require("fs");  
  
const writeStream = fs.createWriteStream("output.txt");  
  
writeStream.write("Hello Node.js!");  
writeStream.end();
```


✓ 파이프 - Pipe

버퍼들을 파이프처럼 연결하는 것



```
const fs = require("fs");  
  
const readStream = fs.createReadStream("input.txt");  
const writeStream = fs.createWriteStream("output.txt");  
  
readStream.pipe(writeStream);
```

비디오 스트리밍 예시

Q&A

감사합니다
