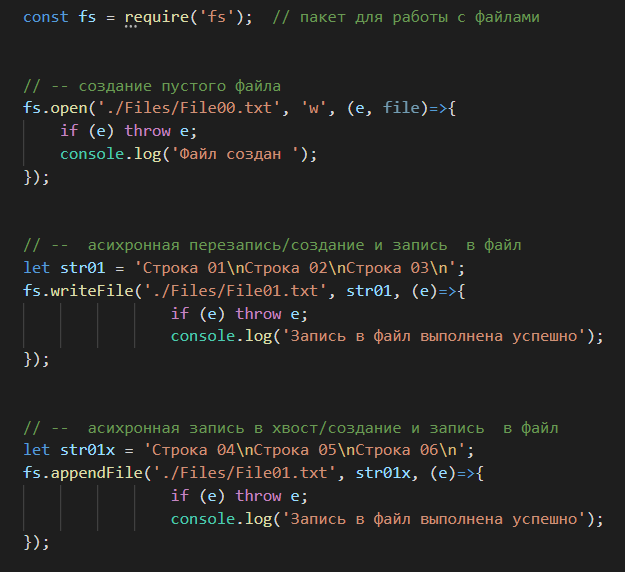
Лекция 09

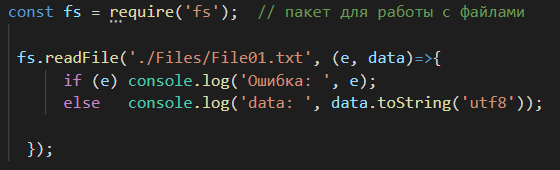
ПСКП, ПОИТ-3

**Node.js: File System**

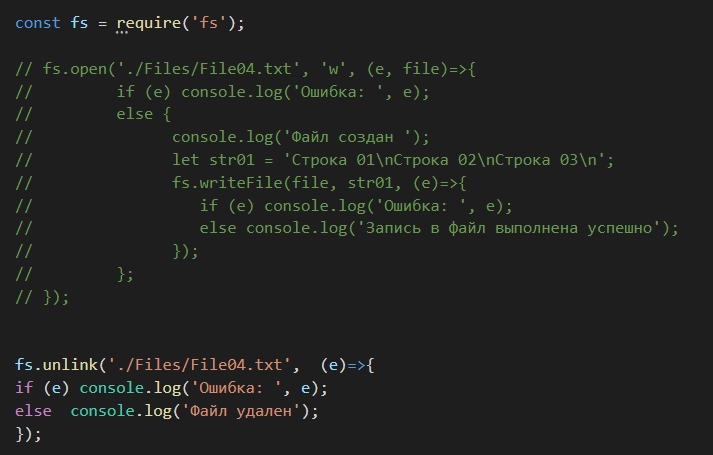
1. **File System: fs** пакет, входящий в **Node.js Core**
2. **File System:** создание файла и запись в файл

****

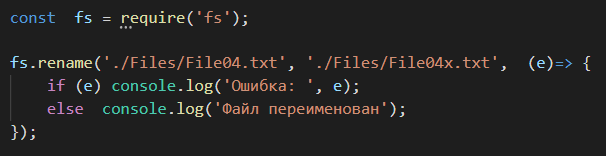
1. **File System:** чтение из файла



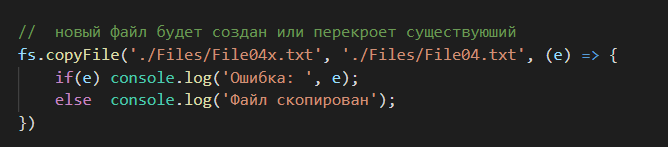
1. **File System:** удаление файла

****

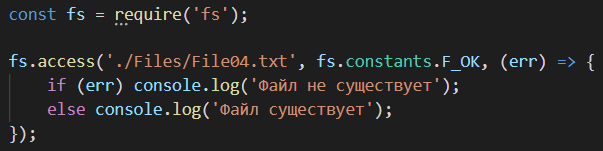
1. **File System:** переименование файла

****

1. **File System:** копирование файла

****

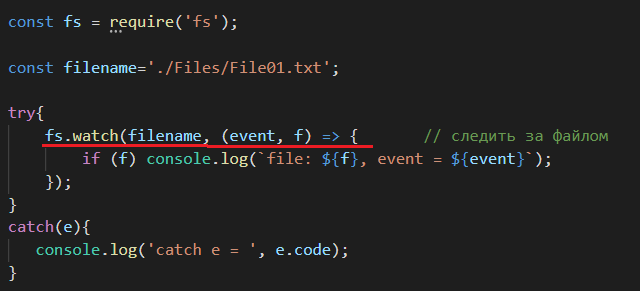
1. **File System:** наличие файла

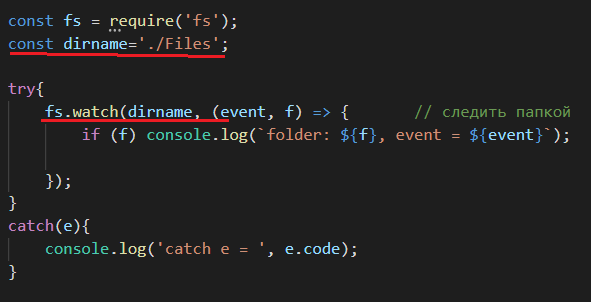


1. **File System:** работа с директориями (создать, удалить, переименовать, прочитать содержимое)

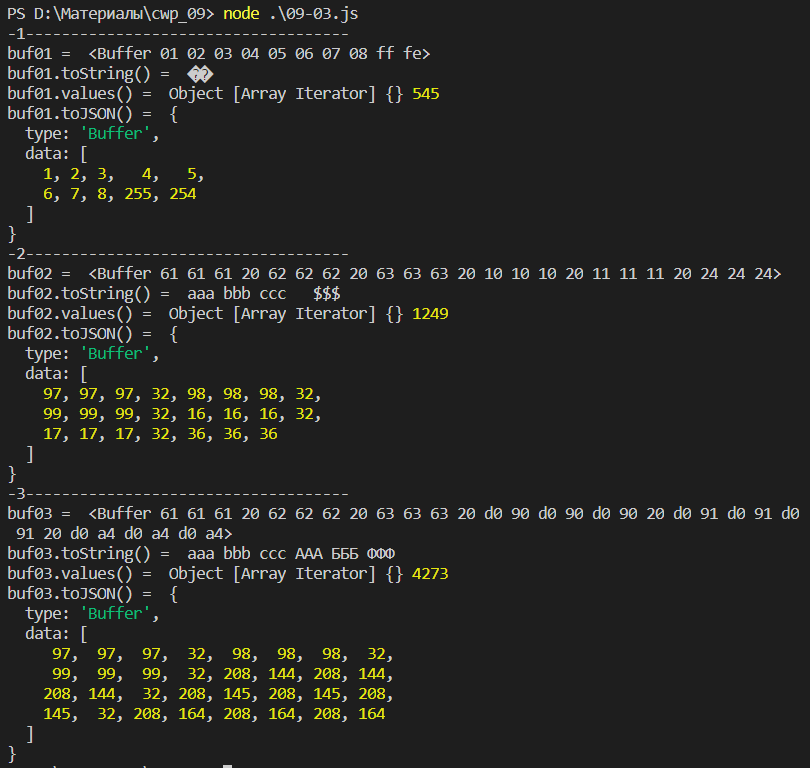


1. **File System:** слежение за файлом

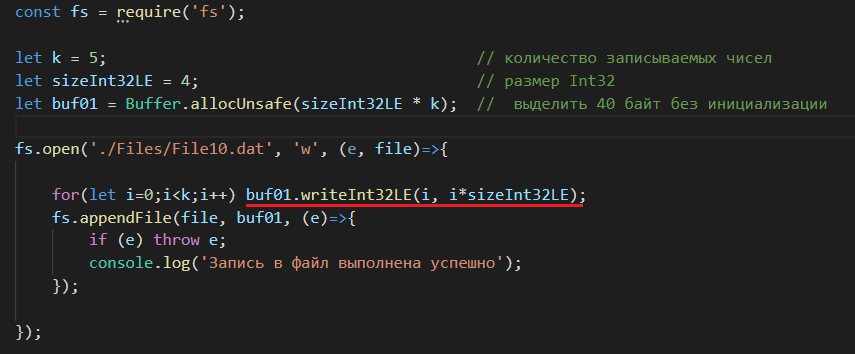
****

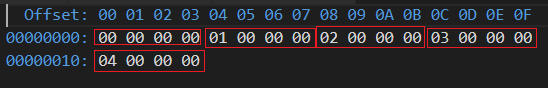
1. **File System:** слежение за директорием ****
2. **Buffer:** объект, предназначенный для работы с потоком октетов; применяется в функциях readFile, writeFile; 1GB (32-bit), 2GB (64-bit)

****

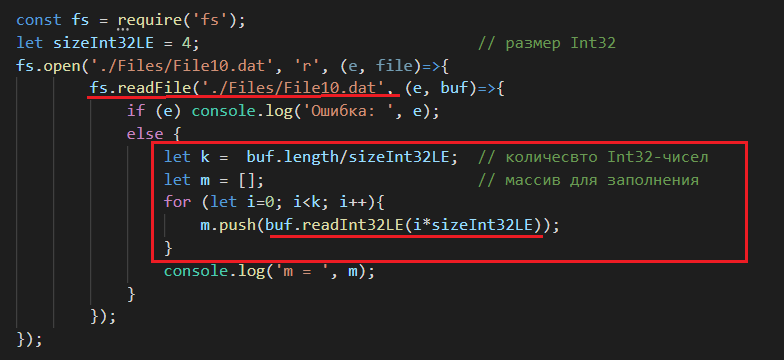


1. **File System:** Buffer, Buffer.write

****

****

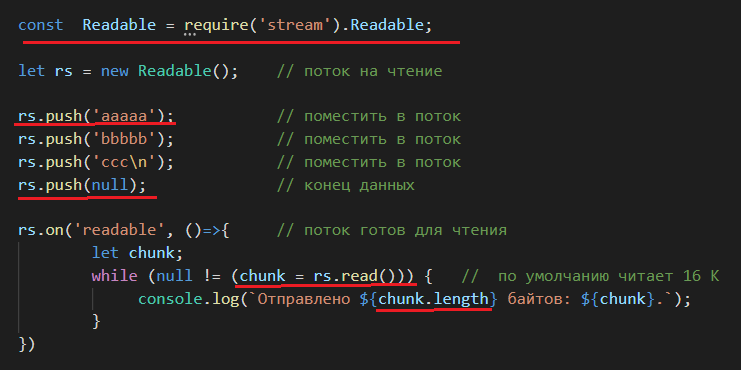
1. **File System:** Buffer, Buffer.read

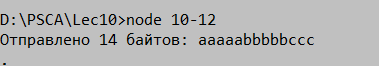
****

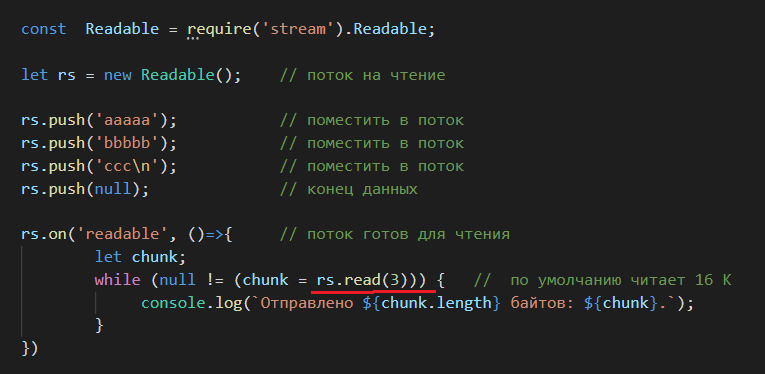
****

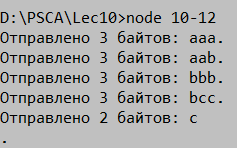
1. **Stream:** абстракция над данными.
2. **Stream:** Writable, Readable, Duplex, Transform.
3. **Stream:** Readable: HTTP request (на сервере), HTTP response (на клиенте), TCP Sockets, fs read, process.stdin.
4. **Stream:** Readable, push(помещает в буфер),read



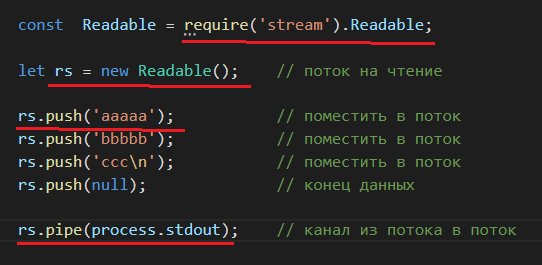


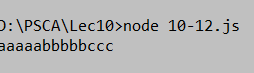






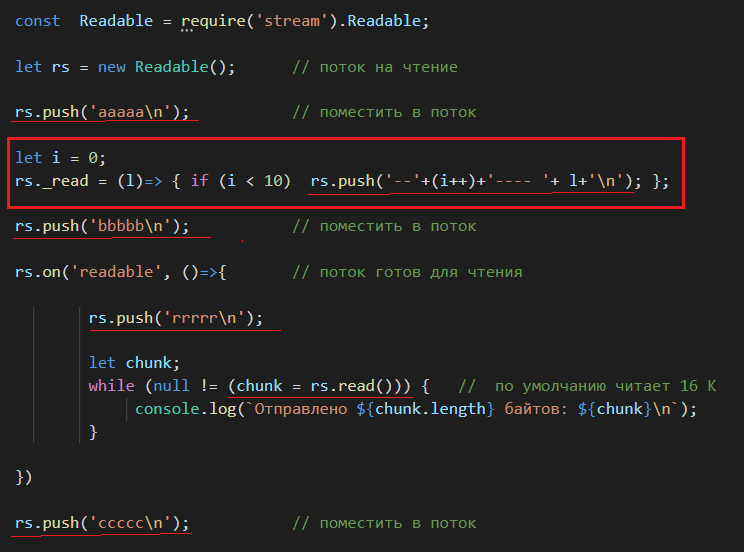
1. **Stream:** Readable, push, pipe

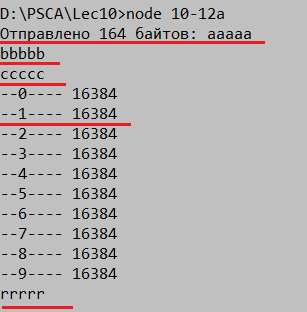
****

****

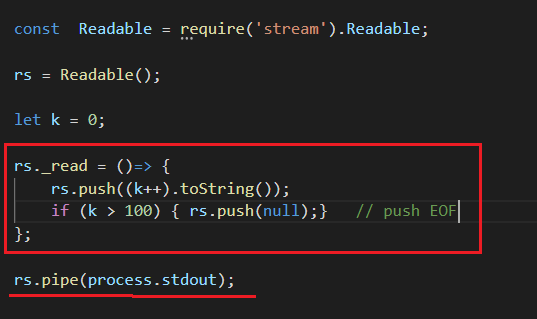
1. **Stream:** Readable, push(помещает в буфер),read, \_read

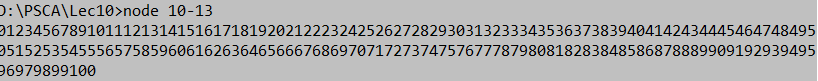




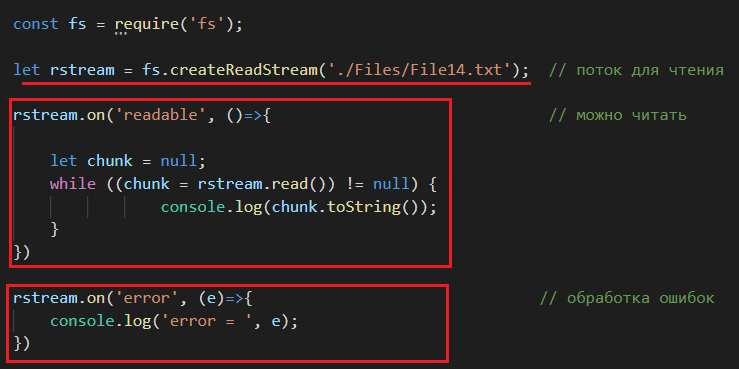
****

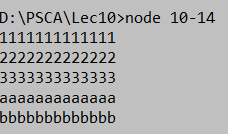
1. **Stream:** Readable, \_read, push, pipe

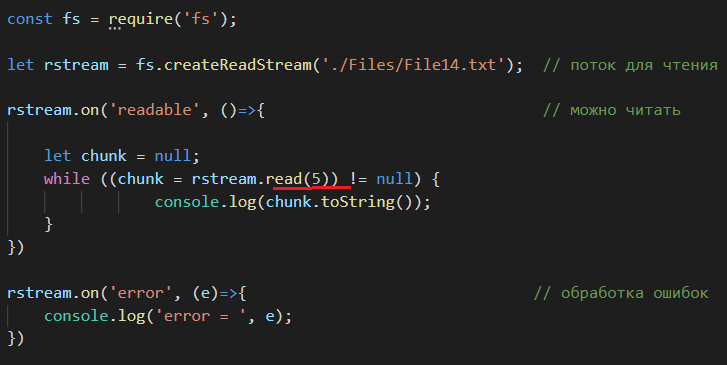
****

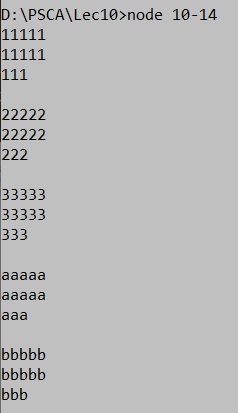
****

1. **Stream:** Readable, fs, paused mode

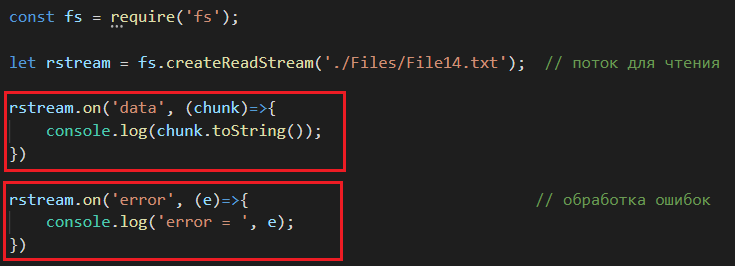
****

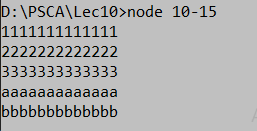
****

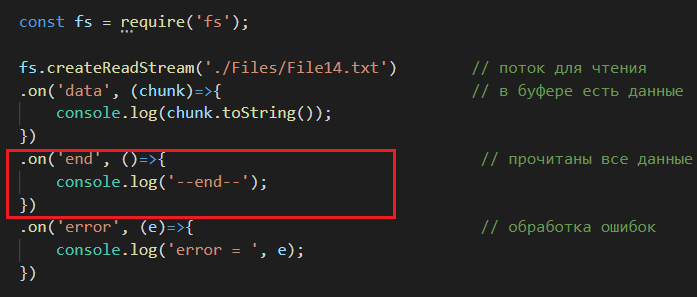
****

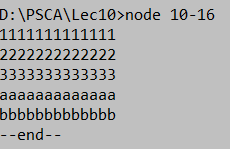
****

1. **Stream:** Readable, fs, flowing mode

****

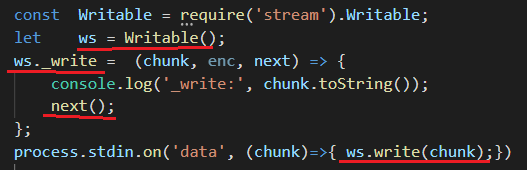
****

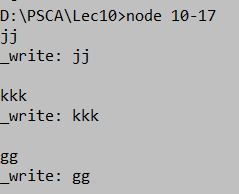
****

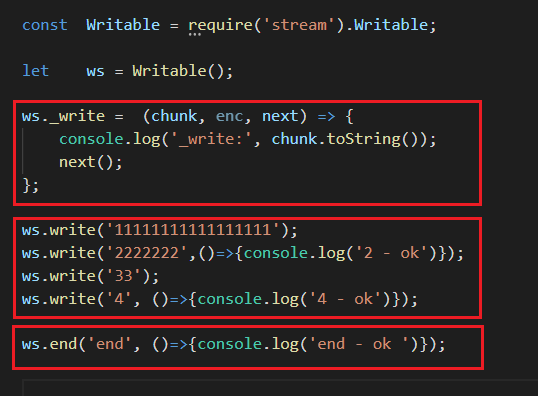
****

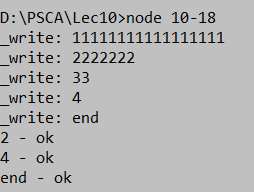
1. **Stream:** Writable: HTTP request (на клиенте), HTTP response (на сервере), TCP Sockets, fs write, process.stdout, process.stderr.
2. **Stream:** Writable, \_write, write



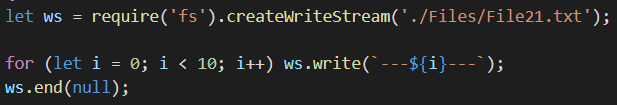
****

****

****

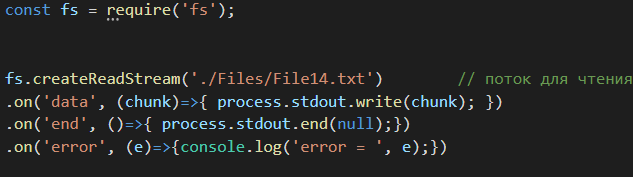
****

1. **Stream:** Writable, fs



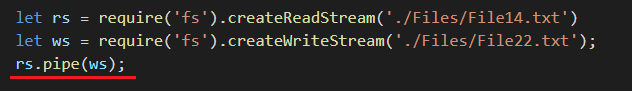
****

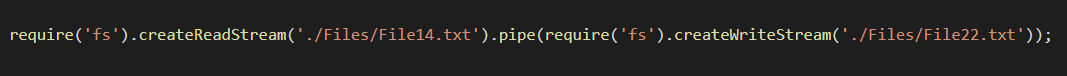
1. **Stream:** Writable, Readable, pipe/, file -> stdout

****

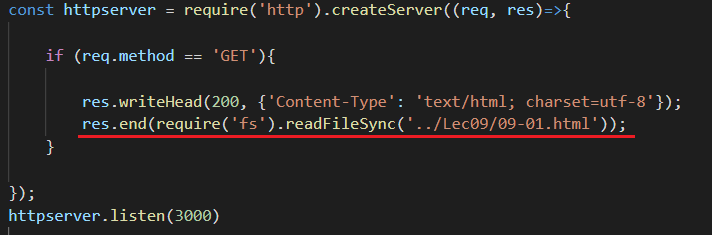
****

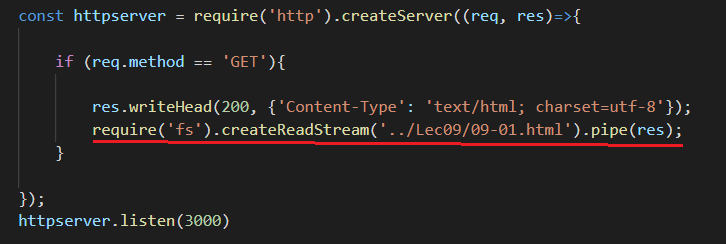
1. **Stream:** Writable, Readable, pipe, file -> file

****

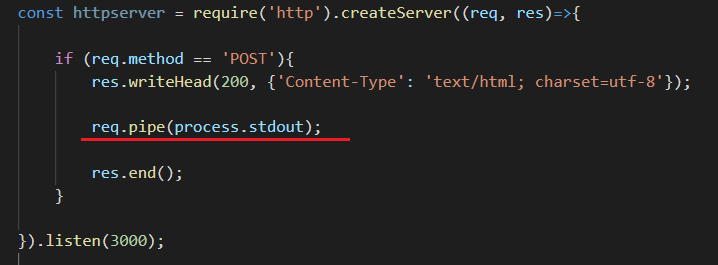
****

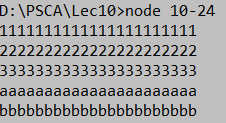
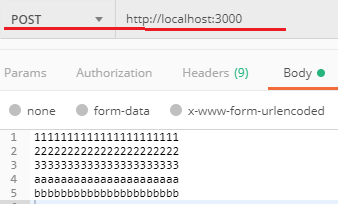
1. **Stream:** Writable, Readable, pipe, file -> response, download

****

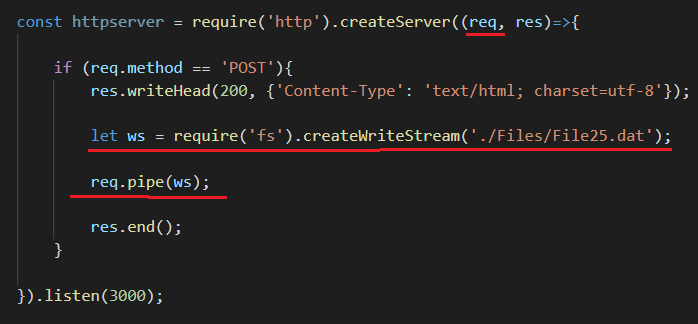
****

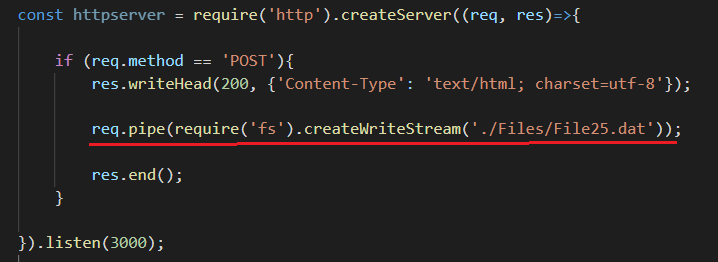
1. **Stream:** Writable, Readable, pipe, request -> process.stdout

****

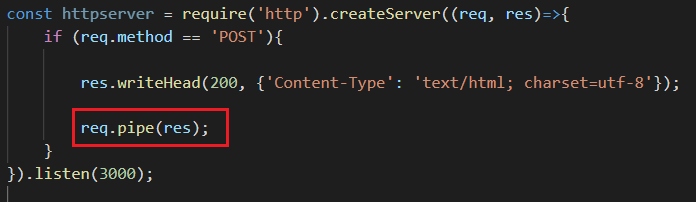
****

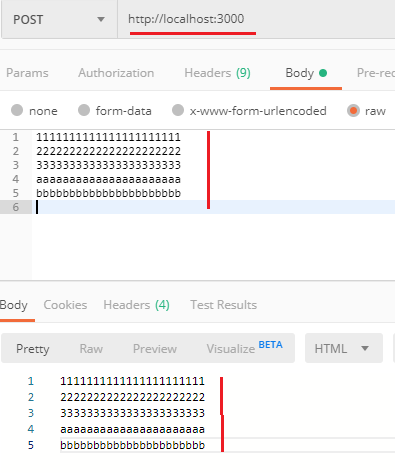
1. **Stream:** Writable, Readable, pipe, request -> file, upload

****

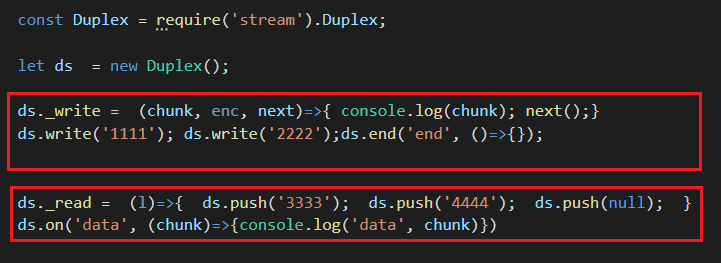
****

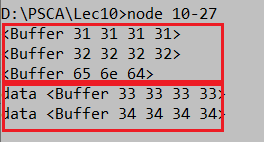
1. **Stream:** Writable, Readable, request -> response

****

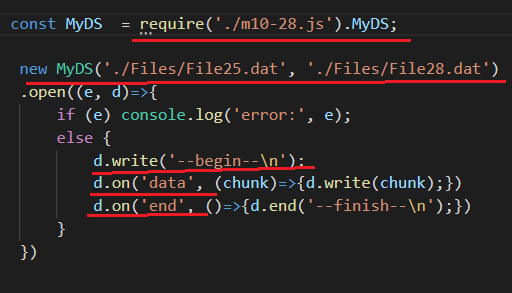
****

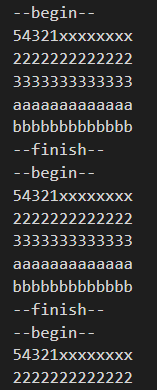
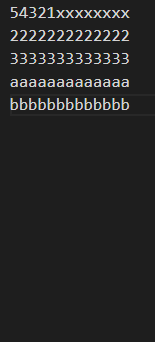
1. **Stream:** Duplex

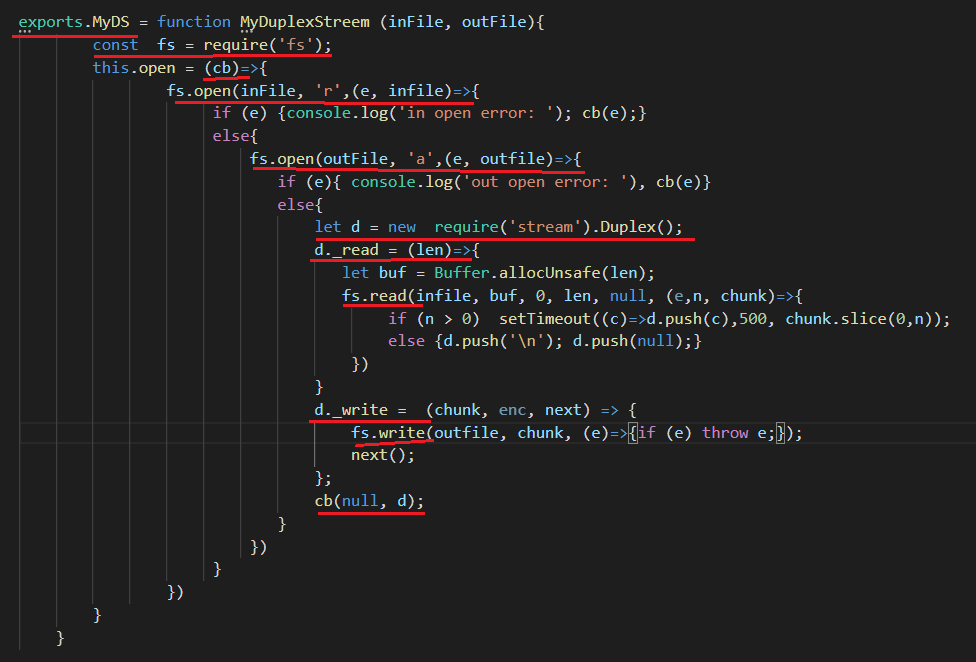
****

****

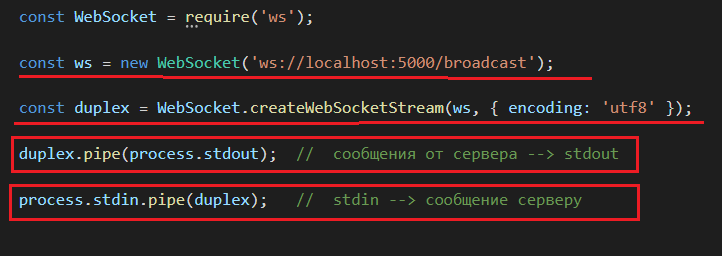
1. **Stream:** Duplex, fs

****

****

****

1. **Stream:** Duplex, websocket (раньше рассматривались).

****

1. <https://webdraftt.com/tutorial/nodejs/streams>
2. <https://medium.com/devschacht/stefan-baumgartner-the-definitive-guide-to-object-streams-in-nodejs-ab983b819eae>
3. <https://makeomatic.ru/blog/2016/08/22/Streams_handbook/#why>