

# 컴퓨터 네트워크 #HW1

소프트웨어학부 컴퓨터전공 2014037847 김진용

## 1. Give a high-level description of your server's design

### 1.1. 클래스

- 1.1.1. WebServer  
서버를 실행에 필요한 클래스를 로드하고 사용자에게 서버 관리 메소드를 제공합니다.
- 1.1.2. ServerSocket  
서버 소켓을 관리합니다.
- 1.1.3. Controller  
요청 정보를 보고 알맞은 응답을 생성합니다.
- 1.1.4. FileController  
요청 정보에 path 정보를 보고 web\_root 를 기준으로 해당 파일을 읽어 응답을 생성합니다.
- 1.1.5. Processor  
queue 에서 대기중인 connection 을 가져와 처리하며 queue 가 비면 작업을 종료합니다.
- 1.1.6. ProcessorPool  
Processor 의 실행을 담당합니다. 요청이 들어오면 Processor 를 실행합니다.
- 1.1.7. Request  
http request message 를 파싱해 Request 인스턴스를 생성합니다.
- 1.1.8. RequestAcceptor  
ServerSocket 으로부터 accept 를 기다리며 accept 되면 queue 에 추가합니다.
- 1.1.9. RequestQueue  
요청을 대기시키며 웹서버가 다중스레드로 돌아가므로 구현된 메소드는 thread-safe 합니다.
- 1.1.10. Response  
Controller 에 의해 생성되며 http message format 에 맞게 serialize 합니다.
- 1.1.11. NotFoundResponse  
404 Not Found 응답 클래스입니다.
- 1.1.12. SuccessResponse  
200 OK 응답 클래스로 body와 content-type 을 파라미터로 받습니다.

### 1.2. 처리 흐름

- 1.2.1. [main] WebServer.start() 호출
- 1.2.2. [WebServer] ServerSocket.bind() / ServerSocket.listen() 호출  
=> socket lib 의 bind 와 listen 함수와 동일
- 1.2.3. [WebServer] RequestAcceptor 쓰레드로 실행
- 1.2.4. [RequestAcceptor] accept 메소드로 요청 대기
- 1.2.5. [RequestAcceptor] 요청 발생
- 1.2.6. [RequestAcceptor] RequestQueue 에 추가
- 1.2.7. [RequestAcceptor] ProcessorPool.proceed() 호출  
=> pool 에 처리할 요청이 있다고 알려줌
- 1.2.8. [ProcessorPool] processor 배열에서 종료된 processor 1개 실행  
=> processor 가 전부 실행중이면 추가 실행 없음  
=> Processor 는 실행되면 새로운 쓰레드에서 동작함
- 1.2.4 ~ 1.2.8 반복
- 1.2.9. [Processor] client\_socket\_fd 에서 요청 메시지 읽어 Request 인스턴스 생성
- 1.2.10. [Request] 요청 메시지를 파싱해 Request 변수값 초기화
- 1.2.11. [Processor] Controller.handle() 호출
- 1.2.12. [Controller] Request 로부터 값을 읽어 Response 인스턴스 생성
- 1.2.13. [Processor] Response.serialize() 로 응답 메시지 생성 후 client 로 send
- 1.2.14. [Processor] client 연결 종료
- 1.2.9 ~ 1.2.14 반복
- 1.2.15. [main] 'Enter' 키 입력

1.2.16. [main] WebServer.stop() 호출

1.2.17. [main] 웹서버 종료

## 2. What difficulties did you face and how did you solve them?

### 2.1. 일부 요청에 빈 메시지가 들어오는 경우

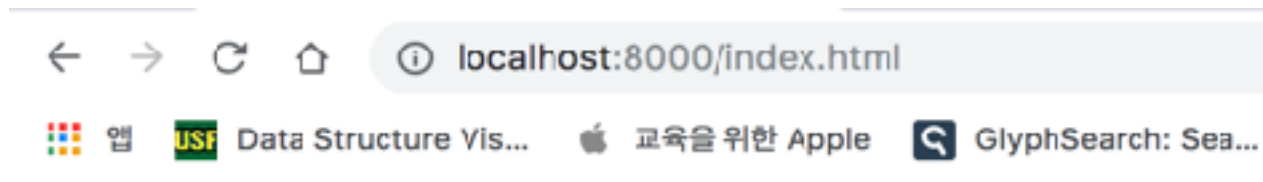
원인을 찾지 못해 요청메시지 길이를 확인해 예외처리했습니다.

### 2.2. 클라이언트가 연결돼 accept 가 넘어갔지만 read 에서 계속 blocking 된 경우

이 부분도 원인을 찾지 못했으나 다음 요청에서 해당 소켓으로 read 메시지가 들어와 동작에는 문제가 없습니다. 다만 웹서버를 종료할 때 join 으로 쓰레드의 처리를 기다렸으나 read 메시지로 인해 종료가 지연되어 종료할 때 join 하는 부분을 주석처리했습니다.

## 3. Include and briefly explain some sample outputs of your client-server (e.g. in Part A you should be able to see an HTTP request)

### 3.1. 웹브라우저



# Hello!! index.html



Thu Apr 04 2019 23:19:52 GMT+0900 (한국 표준시)

[Open Project1.pdf](#)

[Download Over the Horizon.mp3](#)

## 3.2. 콘솔화면

```
Use web_root : ../web_root
Create WebServer (processor_count : 6)
Starting WebServer (port : 8000)
Press 'Enter' key to shutdown

===== New Request =====
GET / HTTP/1.1
Host: localhost:8000
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.85 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/png,*/*;q=0.8,application/signed-exchange;v=b3
Accept-Encoding: gzip, deflate, br
Accept-Language: ko-KR,korq=0.9,en-US;q=0.8,en;q=0.7

Serve file : ../web_root/index.html
===== New Request =====
GET /sign.png HTTP/1.1
Host: localhost:8000
Connection: keep-alive
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.85 Safari/537.36
Accept: image/webp,image/png,image/*/*;q=0.8
Referers: http://localhost:8000/
Accept-Encoding: gzip, deflate, br
Accept-Language: ko-KR,korq=0.9,en-US;q=0.8,en;q=0.7

Serve file : ../web_root/sign.png
===== New Request =====
GET /index.css HTTP/1.1
Host: localhost:8000
Connection: keep-alive
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.85 Safari/537.36
Accept: text/css,*/*;q=0.1
Referers: http://localhost:8000/
Accept-Encoding: gzip, deflate, br
Accept-Language: ko-KR,korq=0.9,en-US;q=0.8,en;q=0.7

Serve file : ../web_root/index.css
===== New Request =====
GET /index.js HTTP/1.1
Host: localhost:8000
Connection: keep-alive
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.85 Safari/537.36
Accept: */*
Referers: http://localhost:8000/
Accept-Encoding: gzip, deflate, br
Accept-Language: ko-KR,korq=0.9,en-US;q=0.8,en;q=0.7

Serve file : ../web_root/index.js

Shutdown WebServer
Destroy WebServer
```

Use web\_root : ../web\_root

=> 파일을 찾을 디렉토리 경로

Create WebServer (processor\_count : 6)

=> processor\_count : 최대 실행할 Processor 개수 (기본값 4)

Starting WebServer (port : 8000)

=> port : 웹서버 포트번호 (기본값 8080)

이 하단 내용은 요청 메시지와 반환된 파일 경로입니다.