# 과제

# Node.js 를 활용한 Profiler 구현

2023년 1학기 자바웹애플리케이션

# 과제 개요 (1/2)

- 제목 : **Node.js 를 활용한 Profiler (Extension) 구현**
- 마감 : **23년 6월 9일(금) 밤 11:55 까지 (**시간 지나면 제출 안됨**)**
- 배점 : 과목 성적을 **100**점 만점으로 했을 때 **40**점 **(40%)**
  - ✓가장 우수한 프로젝트로 선정된 **1**팀은 최대 **10**점까지 추가 획득 가능
- 요구사항
  - ✓**[**입력**]** 웹브라우저를 통해 원본 데이터 파일을 서버로 업로드 **(inputFile.txt)**
  - ✓**[**로직**]**
    - 최소요건
      - 입력 파일의 데이터를 가공 처리하여**,** 각각의 데이터별 **MIN/MAX/AVG** 계산한다**.**
    - 확장 **(**옵션**)**
      - 입력 파일을 **DB**와 연동하여 데이터 가공처리**. (DB** 연동은 교과서 **7**장 또는 인터넷 검색 활용**)**
      - 입력데이터를 추가 가공하여 표준편차 분석등을 통해 그래프/메뉴 버튼 추가 확장 **(→**기능 확장**)**
      - 그래프 종류를 다양화하고 데이터 가동을 통해 **Visualization**을 강화한다**.**
      - 주어진 입력 데이터 이외에 자체 생산한 **(**대규모**)** 입력으로 강력한 프로파일링 기능을 추가한다**.**
  - ✓**[**출력**]** 웹브라우저에서 입력된 데이터를 클릭하여 그래프로 결과를 보여준다**.**
    - **Task** 번호 및 **Core** 번호에 대한 그래프를 따로 그려줄 수 있어야 한다**. (6~9** 페이지 참고**)**
    - 그래프의 형태는 자유롭게 사용 가능 **(**히스토그램**,** 파이차트 등등**)**

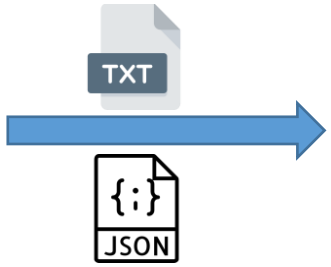
# 과제 개요 (2/2)

- 제출 항목
  - ✓**1. Source Code** 또는 **Source Code** 링크
  - ✓**2.** 프로그램에 대한 개요 설명 **(**문서 파일 또는 **URL)**
    - **1.** 프로그램 수행 절차
    - **2.** 프로그램 수행 과정에 대한 화면 캡쳐
  - ✓**3.** 제출 형식 **:** 파일 **1**개로 제출 **(**파일이 여러 개일 경우 압축해서 **1**개로 만들기**)**
    - 자바웹**-**학번**-**이름**.**파일확장자 **(**예**,** 자바웹**-2021xxxx-**홍길동**.pptx)**
    - 팀장만 제출하면 됨
    - **GitHub** 주소 하나만 제출해도 됨 **(GitHub** 내에 보고서도 포함**)**

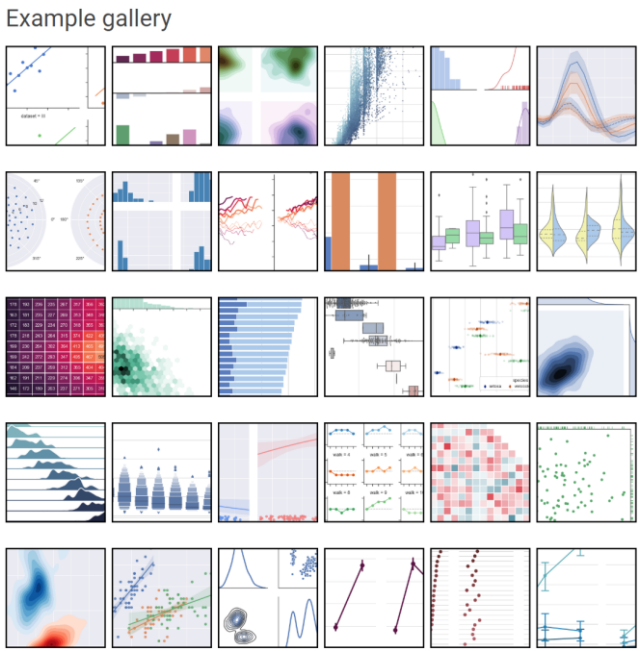
  - ✓중요 **:** 보고서에는 팀에 기여를 많이 한 순서대로 이름을 쓸 것
    - 이름 옆에 기여비율을 입력하는 것은 옵션 **(**단**,** 기여를 많이 한 순서는 표기 필요**)**
    - 예**1)** 홍길동**(25),** 전우치**(25),** 마당쇠**(20),** 장화**(20),** 홍련**(10)**
    - 예**2)** 홍길동**,** 전우치**,** 마당쇠**,** 장화**,** 홍련

# 전체 시스템 동작 파악 (Overall Flow)

**각 단계에서 기능을 확장할 수 있음**








## 프로파일링

- 수집 데이터 선정
- 프로파일러 On/Off
- 최소 부하 수집

## 데이터 저장

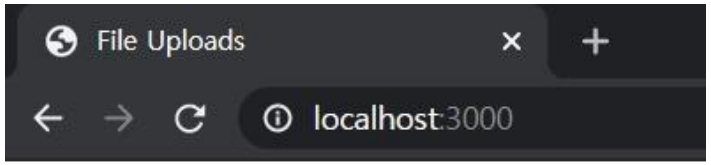
- 입력 포맷 (txt,JSON,)
- DB (SQL) 저장
- 데이터 알고리즘
- 웹서버 개발 필요

## 시각화

- 보고서 생성/관리
- 서버 관리
- 원본 데이터 관리

# 구현 예제

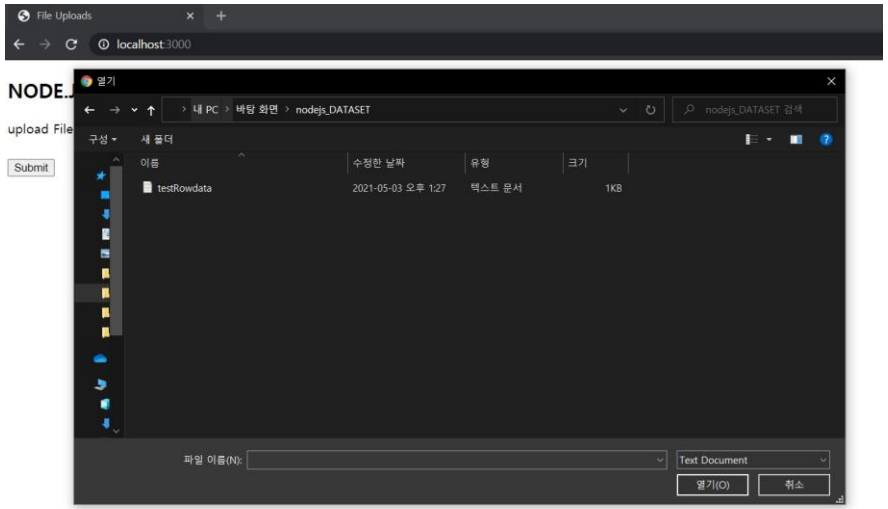
입력 화면

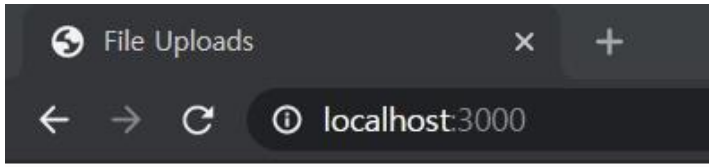



NODE.JS

upload File: 파일 선택 선택된 파일 없음

Submit




NODE.JS

upload File: 파일 선택 testRowdata.txt

Submit

**※ 예제와 동일한 형태로 출력하지 않아도 됨**

# 구현 예제

## 입력 파일

testRowdata - Windows 메모장

파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

| | task1 | task2 | task3 | task4 | task5 |
|---|---|---|---|---|---|
| core1 | 323 | 593 | 396 | 416 | 233 |
| core2 | 309 | 571 | 383 | 479 | 248 |
| core3 | 339 | 553 | 374 | 498 | 231 |
| core4 | 344 | 581 | 360 | 430 | 253 |
| core5 | 344 | 549 | 361 | 429 | 245 |

| | task1 | task2 | task3 | task4 | task5 |
|---|---|---|---|---|---|
| core1 | 307 | 515 | 370 | 448 | 254 |
| core2 | 337 | 525 | 376 | 493 | 245 |
| core3 | 317 | 1672 | 380 | 453 | 233 |
| core4 | 312 | 580 | 342 | 444 | 262 |
| core5 | 308 | 546 | 359 | 480 | 246 |

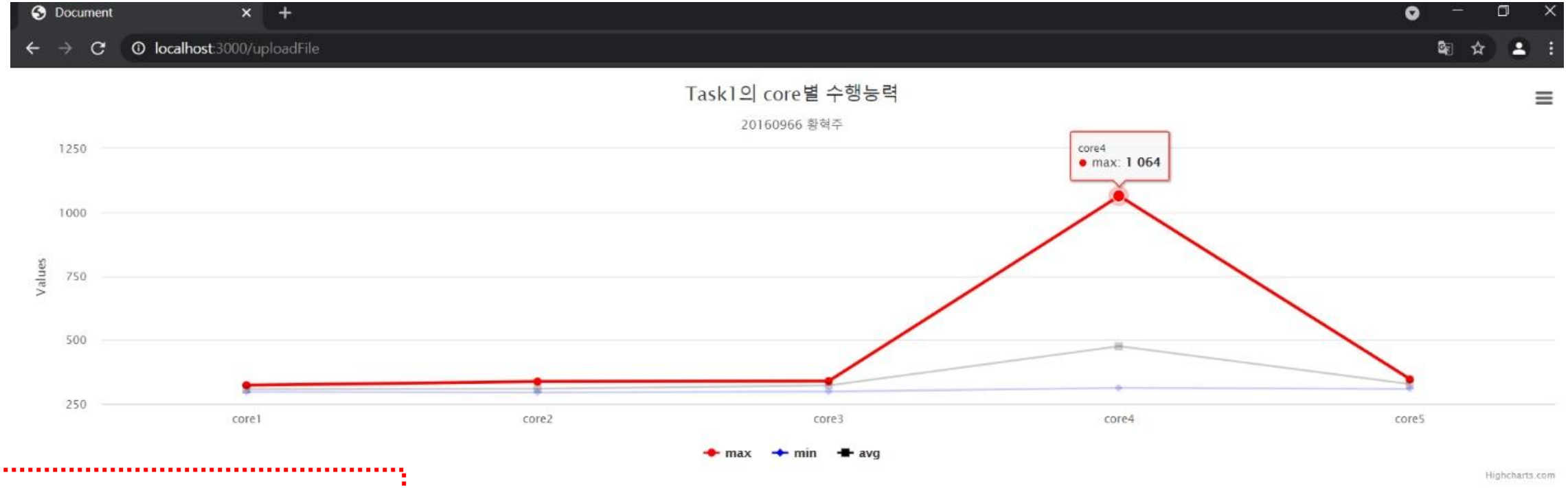
| | task1 | task2 | task3 | task4 | task5 |
|---|---|---|---|---|---|
| core1 | 305 | 543 | 345 | 451 | 230 |
| core2 | 308 | 591 | 381 | 468 | 271 |
| core3 | 327 | 572 | 348 | 437 | 242 |
| core4 | 1064 | 579 | 355 | 448 | 252 |
| core5 | 347 | 533 | 392 | 426 | 256 |

| | task1 | task2 | task3 | task4 | task5 |
|---|---|---|---|---|---|
| core1 | 306 | 558 | 392 | 484 | 235 |
| core2 | 296 | 534 | 355 | 427 | 248 |
| core3 | 298 | 559 | 385 | 476 | 245 |
| core4 | 325 | 582 | 1495 | 949 | 247 |
| core5 | 309 | 525 | 392 | 488 | 252 |

| | task1 | task2 | task3 | task4 | task5 |
|---|---|---|---|---|---|
| core1 | 297 | 563 | 344 | 429 | 250 |
| core2 | 295 | 535 | 341 | 425 | 231 |
| core3 | 326 | 520 | 344 | 499 | 264 |
| core4 | 332 | 1983 | 342 | 478 | 235 |
| core5 | 327 | 563 | 341 | 503 | 268 |

# 구현 예제

## 그래프 표현




**작업단위**

task1 task2 task3 task4 task5

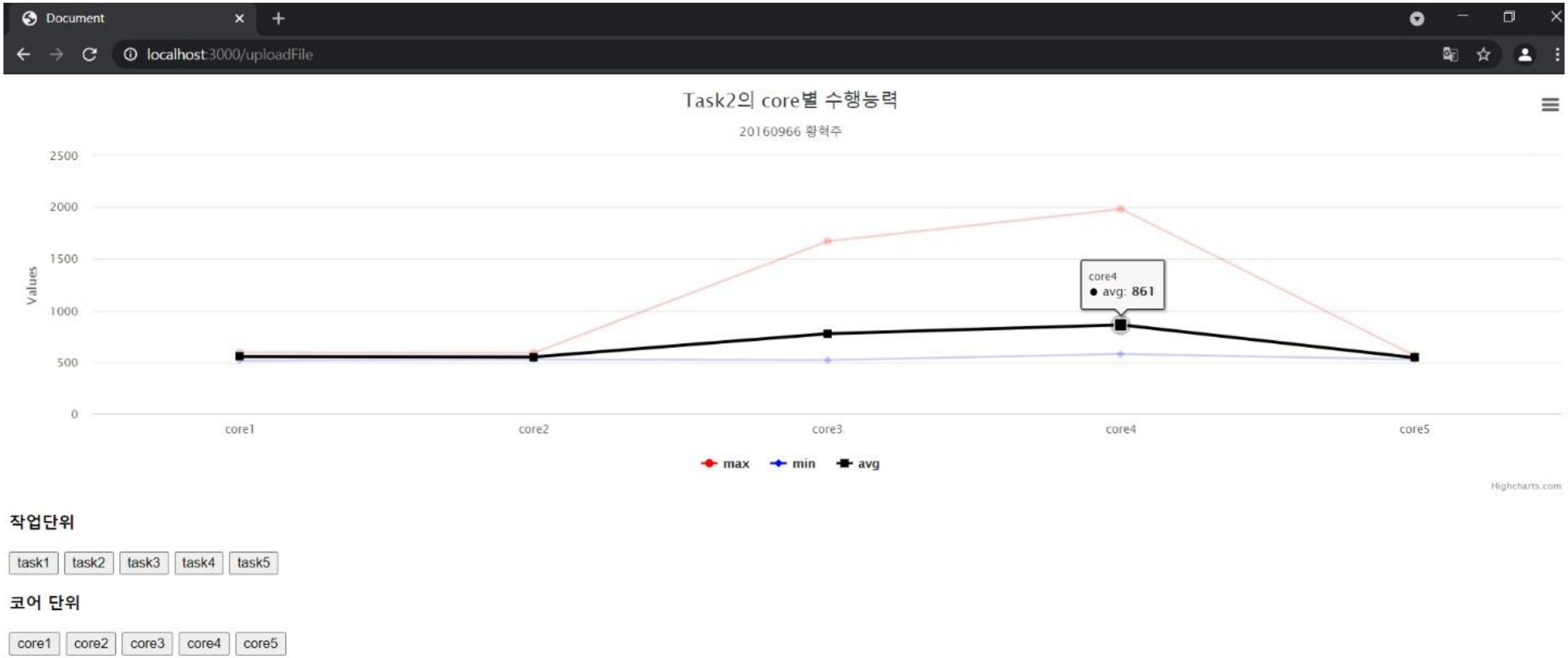
**코어 단위**

core1 core2 core3 core4 core5

**※ 예제와 동일한 형태로 출력하지 않아도 됨**

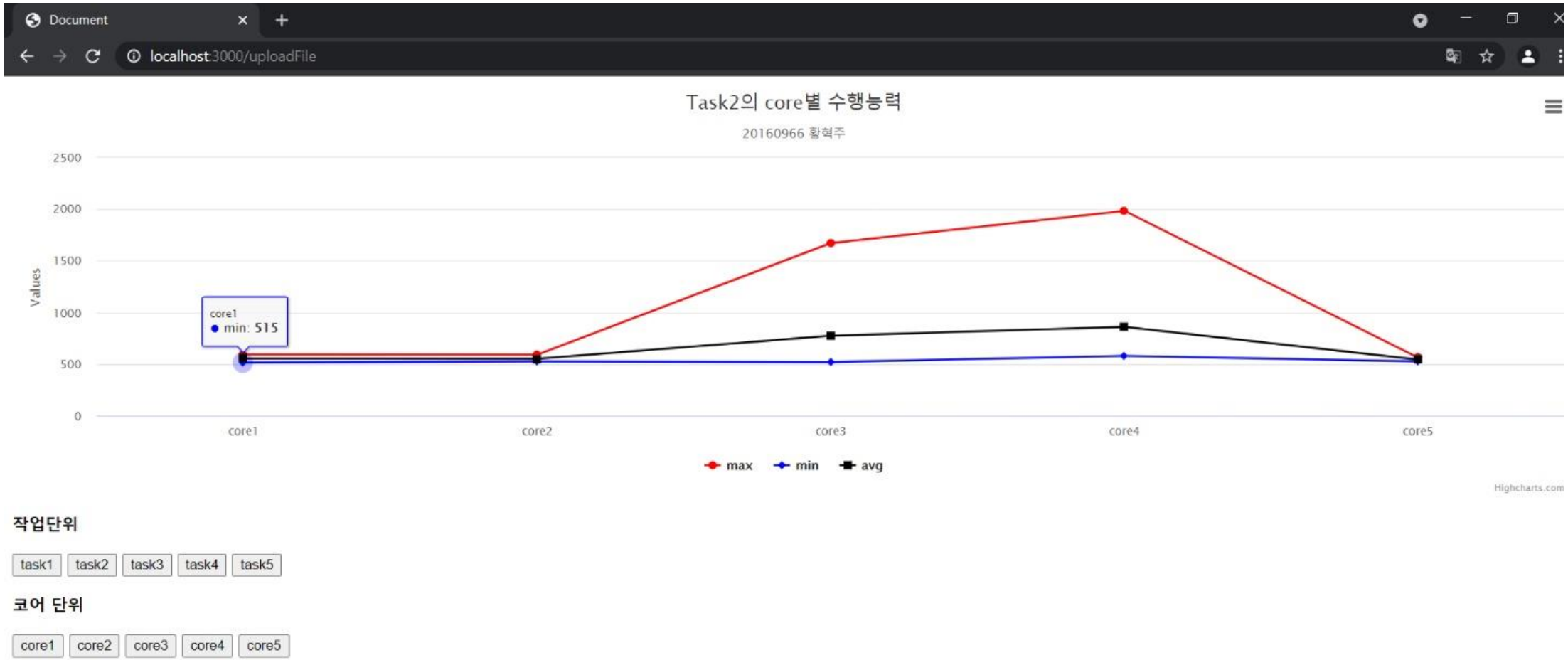
# 구현 예제

## 그래프 표현


Document
localhost:3000/uploadFile
Task2의 core별 수행능력
20160966 황혁주
2500
2000
1500
1000
500
0
Values
core1
core2
core3
core4
core5
core4
avg: 861
max
min
avg
Highcharts.com
작업단위
task1
task2
task3
task4
task5
코어 단위
core1
core2
core3
core4
core5

# 구현 예제

## 그래프 표현


Document
localhost:3000/uploadFile
Task2의 core별 수행능력
20160966 황혁주
2500
2000
1500
1000
500
0
Values
core1
min: 515
core1
core2
core3
core4
core5
max
min
avg
Highcharts.com
작업단위
task1
task2
task3
task4
task5
코어 단위
core1
core2
core3
core4
core5

# 구현 예제


Document
localhost:3000/uploadFile
core1의 task별 수행능력
20160966 황혁주
700
600
500
400
300
200
Values
task1
task2
task3
task4
task5
max
min
avg
Highcharts.com
작업단위
task1
task2
task3
task4
task5
코어 단위
core1
core2
core3
core4
core5

# 구현 예제 (Option)

Dashboard 형태