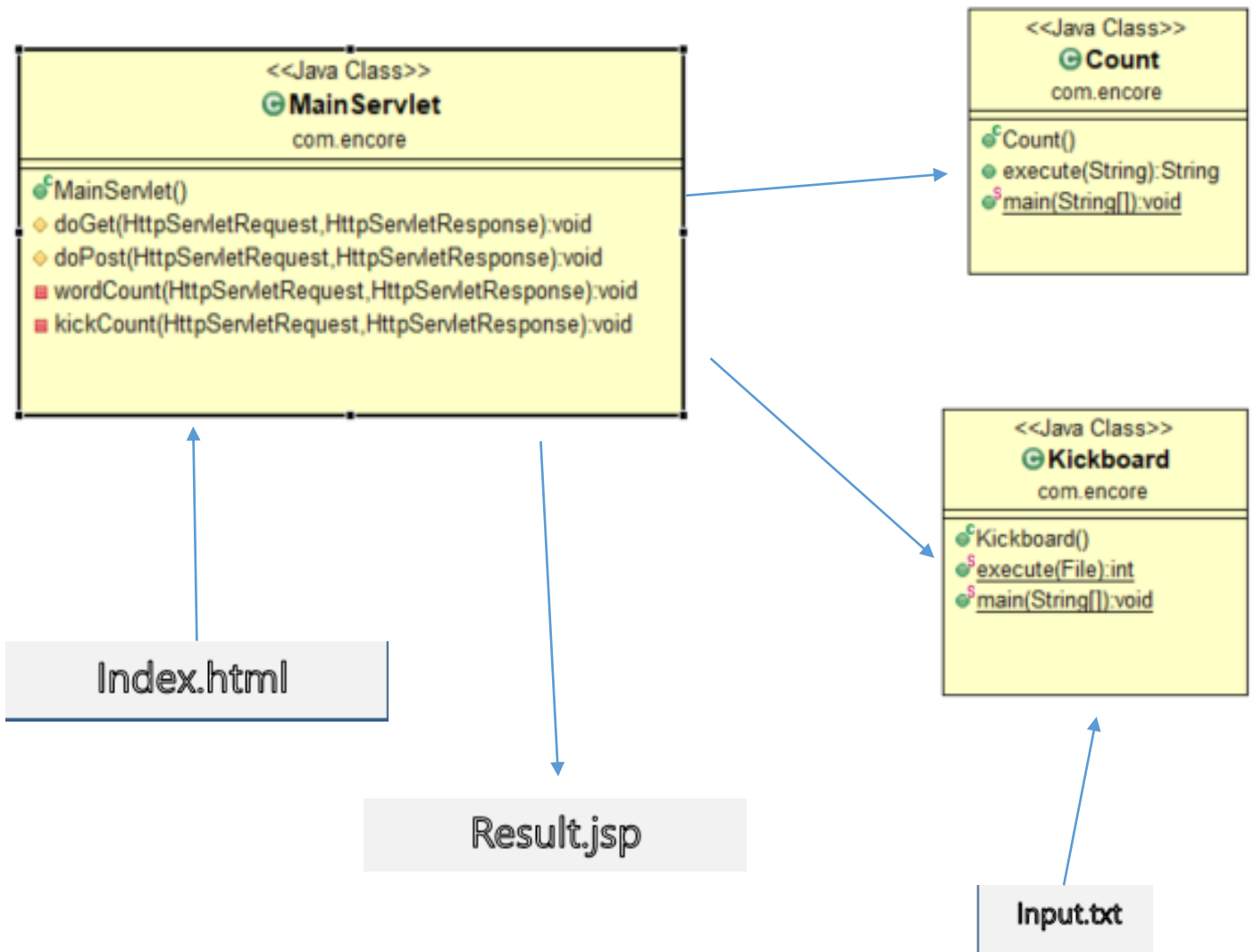


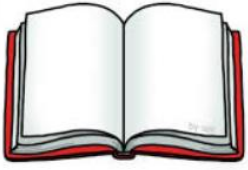
# Web + Ajax + Algorithm Project



- ✓ 알고리즘을 이용해서 문제를 해결하고 프로그램 결과로 표현할 수 있어야 한다.
- ✓ 웹 페이지의 요청 메커니즘을 이해해야 한다.
- ✓ 비동기 통신의 요청, 응답을 핸들링 할 수 있어야 한다.

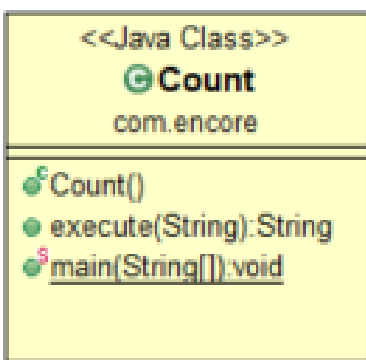


# Web + Ajax + Algorithm Project



- ✓ 알고리즘을 이용해서 문제를 해결하고 프로그램 결과로 표현할 수 있어야 한다.
- ✓ 웹 페이지의 요청 메커니즘을 이해해야 한다.
- ✓ 비동기 통신의 요청, 응답을 핸들링 할 수 있어야 한다.

## | 구현 해야 할 내용 1. 단어 세기 (Count.java)



위 Count 클래스의 execute()를 구현하세요.

execute()는 파라미터로 문장이 입력된다. 그 문장내에 가장 많이 등장하는 단어를 찾아 리턴하세요.

<조건>

1. 가장 많이 등장하는 단어의 빈도수가 같을 경우 알파벳순으로 앞선 단어를 출력한다.
2. 모두 소문자로 변경하여 처리한다.

파라미터 예)

Can Danny and his father outsmart the villainous Mr. Hazell? Danny has a life any boy would love - his home is a gypsy caravan, he's the youngest master car mechanic around, and his best friend is his dad, who never runs out of wonderful stories to tell. But one night Danny discovers a shocking secret that his father has kept hidden for years.

리턴 예)

his

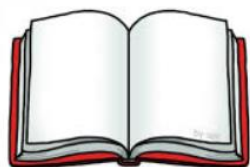
파라미터 예)

I like cat. I like cat. I like cat.

리턴 예)

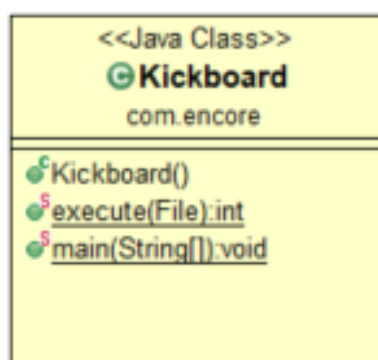
cat.

# Web + Ajax + Algorithm Project



- ✓ 알고리즘을 이용해서 문제를 해결하고 프로그램 결과로 표현할 수 있어야 한다.
- ✓ 웹 페이지의 요청 메커니즘을 이해해야 한다.
- ✓ 비동기 통신의 요청, 응답을 핸들링 할 수 있어야 한다.

## | 구현 해야 할 내용 2.세준이의 킥보드 (Kickboard.java)



위 Kickboard 클래스의 execute( )를 구현하세요.

execute( )는 파라미터로 데이터 파일명이 입력된다. 그 파일에 있는 데이터를 읽어서 아래의 내용에 맞게 처리하도록 구현하세요. (추가 메소드 구현 가능)

### >> 문제설명

킥보드가 갖고 싶었던 세준이는 평화나라에서 저렴하게 킥보드를 구매했다  
엔코아 아카데미 인근을 나타내는 지도는 각 칸마다 지대의 높이가 적혀있고 각 지점은 상하좌우로 이동이 가능하다.

지도에서 가장 왼쪽 위 칸의 남부터미널역 에서 출발하여 가장 오른쪽 아래인 엔코아 아카데미로 이동하고자 한다. 그런데 평화나라에서 구매한 킥보드는 힘이 약해서 현재 위치보다 높이가 낮은 위치로만 이동이 가능하다.

50	45	37	32	30
35	50	40	20	25
30	30	25	17	28
27	24	22	15	10

# Web + Ajax + Algorithm Project



- ✓ 알고리즘을 이용해서 문제를 해결하고 프로그램 결과로 표현할 수 있어야 한다.
- ✓ 웹 페이지의 요청 메커니즘을 이해해야 한다.
- ✓ 비동기 통신의 요청, 응답을 핸들링 할 수 있어야 한다.

앞 페이지 지도에서는 아래와 같이 세 가지 이동이 가능하다.  
출근지역 지도가 주어질 때 세준이가 남부터미널에서 엔코아 아카데미까지 구매한 킥보드를 이용해 이동할 수 있는 경로의 수를 구하여라.

50	45	37	32	30
35	50	40	20	25
30	30	25	17	28
27	24	22	15	10

50	45	37	32	30
35	50	40	20	25
30	30	25	17	28
27	24	22	15	10

50	45	37	32	30
35	50	40	20	25
30	30	25	17	28
27	24	22	15	10

## 입력 조건

첫째 줄에는 지도의 세로의 크기 H와 가로 크기 W가 빈칸을 사이에 두고 주어진다. 이어 다음 H개 줄에 걸쳐 한 줄에 W개씩 위에서부터 차례로 각 지점의 높이가 빈 칸을 사이에 두고 주어진다. H와 W는 각각 500이하의 자연수이고, 각 지점의 높이는 10000이하의 자연수이다.

## 출력 조건

첫째 줄에 이동 가능한 경로의 수 K를 출력한다. 모든 입력에 대하여 K는 10억 이하의 음이 아닌 정수이다.

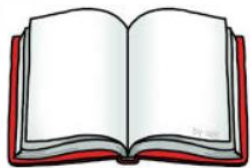
## 입력 예

```
4 5
50 45 37 32 30
35 50 40 20 25
30 30 25 17 28
27 24 22 15 10
```

## 출력 예

3

# Web + Ajax + Algorithm Project



- ✓ 알고리즘을 이용해서 문제를 해결하고 프로그램 결과로 표현할 수 있어야 한다.
- ✓ 웹 페이지의 요청 메커니즘을 이해해야 한다.
- ✓ 비동기 통신의 요청, 응답을 핸들링 할 수 있어야 한다.

## | 구현 해야 할 내용 3. 웹페이지 구현하기 ( index.html )

2개의 알고리즘을 실행해 결과를 볼 수 있도록 입력과 출력을 제공하는 웹페이지를 완성하여 보자.

➤ 초기 화면은 다음과 같다.

The screenshot shows a web browser window with the address bar displaying 'localhost/Monthly9Test/'. The page content is divided into two main sections, each enclosed in a blue dashed border.

**단어세기**

입력 문자열 :

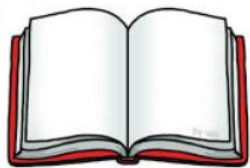
가장 많이 나온 단어 :

**Kickboard**

입력 파일명:

갈 수 있는 경로의 수 :

# Web + Ajax + Algorithm Project



- ✓ 알고리즘을 이용해서 문제를 해결하고 프로그램 결과로 표현할 수 있어야 한다.
- ✓ 웹 페이지의 요청 메커니즘을 이해해야 한다.
- ✓ 비동기 통신의 요청, 응답을 핸들링 할 수 있어야 한다.

## | 구현 해야 할 내용 3. 웹페이지 구현하기 ( index.html )

### 3.1 [구현] CSS 적용

index.html 페이지를 CSS를 적용하여 다음과 같이 만들어 보자. (2곳을 구현하세요)  
(결과 값 출력은 기본폰트보다 1.5배 크기로 빨간색으로 표시)

### 3.2 [구현] jQuery 이벤트

실행 버튼 클릭시 입력된 데이터를 추출하여, 비동기로 서버를 호출하고, 실행결과를 화면에 반영해 보자. ( Kickboard 실행버튼 코드 참조)

실행 결과 화면 예시)

### 단어세기

입력 문자열 : I like cat. I like cat. I like cat.

실행

가장 많이 나온 단어 : cat.

### Kickboard

입력 파일명: input.txt

실행

갈수 있는 경로의 수 : 3