구스 시연시나리오

홈페이지

공개 스터디와 비공개 스터디 확인

상단의 스터디생성 메뉴를 통해 스터디 생성(비공개)

모집 게시판

게시판으로 이동 후 생성된 스터디에 관한 모집 게시글 작성

다른 이용자들이 해당 스터디에 가입신청을 넣음

스터디 페이지

스터디원의 신청을 수락한 후 대기하면 스터디원이 입장

공개 스터디에서의 강퇴 기능 시연

이후 비공개 스터디 생성 후 신청인원 수락 후 입장

스터디 룸

상단의 타이머를 30분으로 설정 후 시작

화면공유를 시작한 스터디원의 화면 확대(?)

IDE 화면에서 작성된 코드 수정

완료 후 컴파일버튼으로 실행결과 확인

퇴장 후 일정등록

공지사항 작성 후 확인

**import** **java.io.BufferedReader**;

**import** **java.io.InputStreamReader**;

**import** **java.util.StringTokenizer**;

**public** **class** **Main** {

**private** **static** int N, M;

**private** **static** int[][] map;

**private** **static** int[][] dp;

**private** **static** int[] dx = {-1, 0, 1, 0};

**private** **static** int[] dy = {0, 1, 0, -1};

**public** **static** void main(String[] args) **throws** Exception {

BufferedReader br = **new** BufferedReader(**new** InputStreamReader(System.in));

StringTokenizer st = **new** StringTokenizer(br.readLine());

N = Integer.parseInt(st.nextToken());

M = Integer.parseInt(st.nextToken());

map = **new** int[N + 1][M + 1];

dp = **new** int[N + 1][M + 1];

**for** (int i = 1; i <= N; i++) {

st = **new** StringTokenizer(br.readLine());

**for** (int j = 1; j <= M; j++) {

map[i][j] = Integer.parseInt(st.nextToken());

dp[i][j] = -1;

}

}

System.out.println(dfs(1, 1));

}

**private** **static** int dfs(int x, int y) {

**if** (x == N && y == M) **return** 1;

**if** (dp[x][y] != -1) **return** dp[x][y];

**else** {

dp[x][y] = 0;

**for** (int i = 0; i < dx.length; i++) {

int nx = x + dx[i];

int ny = y + dy[i];

**if** (nx < 1 ny < 1 nx > N || ny > M) **continue**;

**if** (map[x][y] > map[nx][ny]) {

dp[x][y] += dfs(nx, ny);

}

}

}

**return** dp[x][y];

}

}

마지막에 노션링크 여부 확인