보통 자바에서 입력은 Scanner 클래스를 배워 사용한다

두 클래스의 차이점은,

Scanner 클래스는 Space, Enter 모두 경계로 입력값을 인식하고,

BufferedReader 클래스는 Enter만을 경계로 입력값을 인식한다

그래서 가공하는 작업이 추가로 필요하다 (구분하여 나누기 작업)

Scanner가 한글자씩 받고 처리하는 반면에 Buffer는 들어오는 값을 모아두었다가 한번에 쏟아낸다(flush). 그래서 Scanner보다 BufferedReader와 BufferedWriter가 빠르다.

데이터의 양이 적을 때는 차이가 미미하겠지만 양이 많을 경우에

하나하나씩 전달하지 않고 버퍼에 한 번에 모아서 전달하는 BufferedReader클래스가

속도면에서 빠르고 효율적이다

BufferedReader bf = new BufferedReader(new InputStreamReader(System.in)); //선언

String s = bf.readLine(); //String

int i = Integer.parseInt(bf.readLine()); //Int

StringTokenizer st = new StringTokenizer(s); //StringTokenizer인자값에 입력 문자열 넣음

int a = Integer.parseInt(st.nextToken()); //첫번째 호출

int b = Integer.parseInt(st.nextToken()); //두번째 호출

String array[] = s.split(" "); //공백마다 데이터 끊어서 배열에 넣음

BufferedWriter bw = new BufferedWriter(newOutputStreamWriter(System.out));//선언

String s = "abcdefg";//출력할 문자열

bw.write(s);//출력

bw.newLine(); //줄바꿈

bw.flush();//남아있는 데이터를 모두 출력시킴 bw.close();//스트림을 닫음

Stringbuilder

String은 불변(immutable)하기 때문에 값을 변경할 수 없다

.concat 혹은 +를 이용한 값 변경은 기존 String에 들어있던 값을 버리고

새로 값을 할당하는 것이다, 따라서 concat이나 +를 많이 사용할 경우 속도가 현저히 느려진다.

sb.append(값)

- StringBuffer, StringBuilder 뒤에 값을 붙인다

sb.insert(인덱스, 값)

- 특정 인덱스부터 값을 삽입한다

sb.delete(인덱스, 인덱스)

- 특정 인덱스부터 인덱스까지 값을 삭제한다

sb.indexOf(값)

- 값이 어느 인덱스에 들어있는지 확인한다

sb.substring(인덱스, 인덱스)

- 인덱스부터 인덱스까지 값을 잘라온다

sb.length()

- 길이 확인

sb.replace(인덱스, 인덱스, 값)

- 인덱스부터 인덱스까지 값으로 변경

sb.reverse()

- 글자 순서를 뒤집는다