8. [Java] ??? ???? - Character.isDigit(c)

> https://school.programmers.co.kr/learn/courses/30/lessons/120902

> 틀린 코드 (1차 시도)

```
class Solution {
    public int solution(String my_string) {
        char[] arr = my_string.toCharArray();
        int sum = 0;
        for (char c : arr ) {
            if (c != '+' && c != '-' && c !=' ') {
                sum+= Integer.parseInt(c);
            }
        }
        return sum;
    }
}
```

실행 결과

```
/Solution.java:7: error: incompatible types: char cannot be converted to String
                sum+= Integer.parseInt(c);
                                       ^
Note: Some messages have been simplified; recompile with -Xdiags:verbose to get full output
1 error

테스트 결과 (~˘▾˘)~
1개 중 0개 성공
```

? Integer.parseInt()

- String을 int형으로 변환하는 코드를 써서 에러가 났다 .
- char타입을 정수로 변환할 때는 Integer.parseInt() 대신 char 타입 변수에 - '0'을 한다 . (ASCII 코드)

> “ String → int VS char → int

1. String을 int로 변환할 때는 Integer.parseInt()

```
String str = "123";

// 문자열 → int
int num = Integer.parseInt(str);       // 123
```

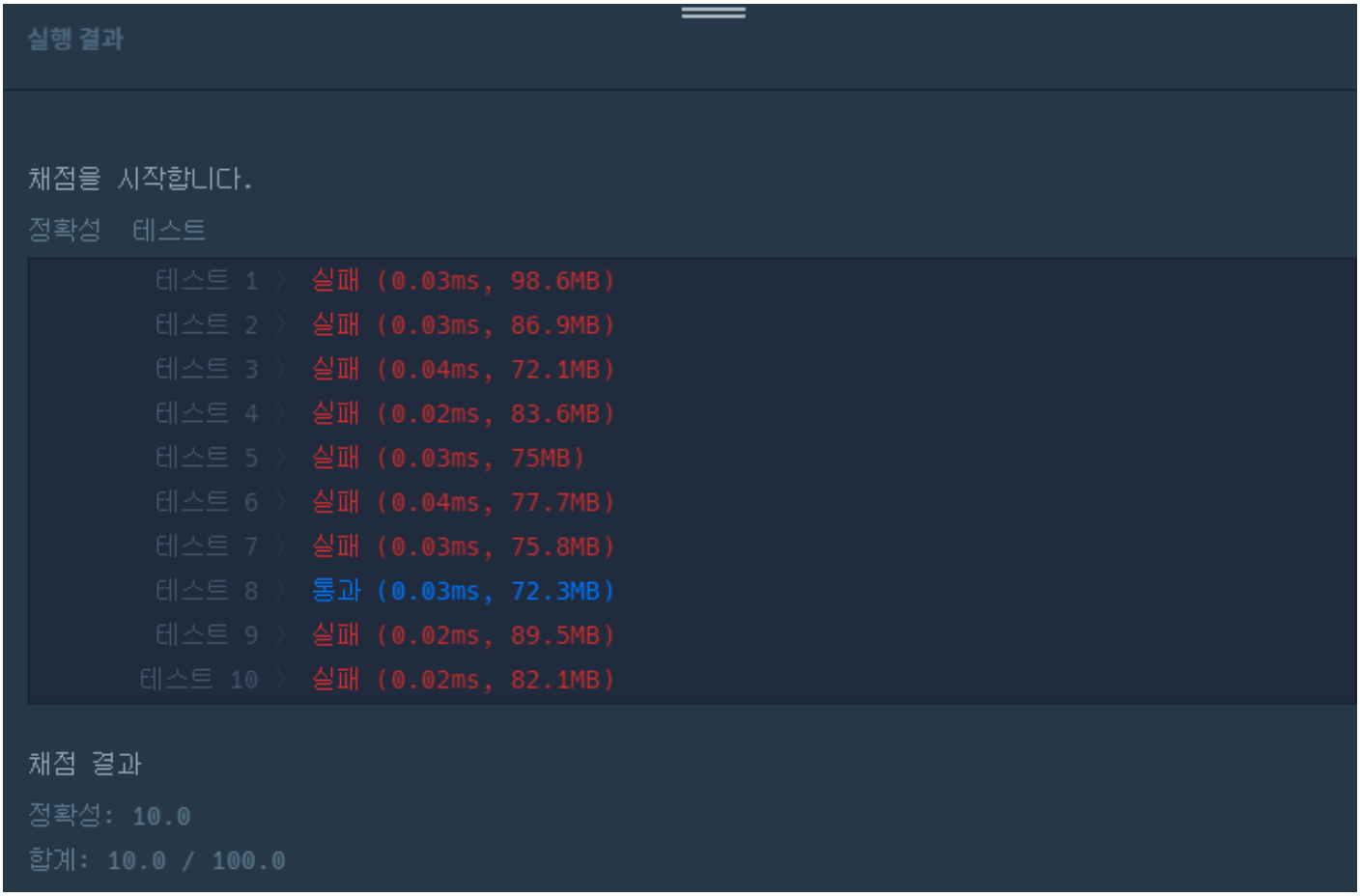
2. char을 숫자로 변환할 때는 char 타입 변수 - '0'

```
char c = '7';

// 문자 → 숫자 (정수값)
int digit = c - '0';   // '7' - '0' = 7
```

틀린 코드 (2차 시도)

```
class Solution {
    public int solution(String my_string) {
        char[] arr = my_string.toCharArray();
        int sum = 0;
        for (char c : arr ) {
            if (c != '+' && c != '-' && c !=' ') {
                sum+= c - '0';
            }
        }
        return sum;
    }
}
```



```
채점을 시작합니다.
정확성  테스트
      테스트 1 〉 실패 (0.03ms, 98.6MB)
      테스트 2 〉 실패 (0.03ms, 86.9MB)
      테스트 3 〉 실패 (0.04ms, 72.1MB)
      테스트 4 〉 실패 (0.02ms, 83.6MB)
      테스트 5 〉 실패 (0.03ms, 75MB)
      테스트 6 〉 실패 (0.04ms, 77.7MB)
      테스트 7 〉 실패 (0.03ms, 75.8MB)
      테스트 8 〉 통과 (0.03ms, 72.3MB)
      테스트 9 〉 실패 (0.02ms, 89.5MB)
     테스트 10 〉 실패 (0.02ms, 82.1MB)

채점 결과
정확성: 10.0
합계: 10.0 / 100.0
```

내가 쓴 코드는 모든 숫자를 "한 자리 숫자"로 처리한다. 즉, "12"가 있으면 1 + 2 = 3으로 계산한다.

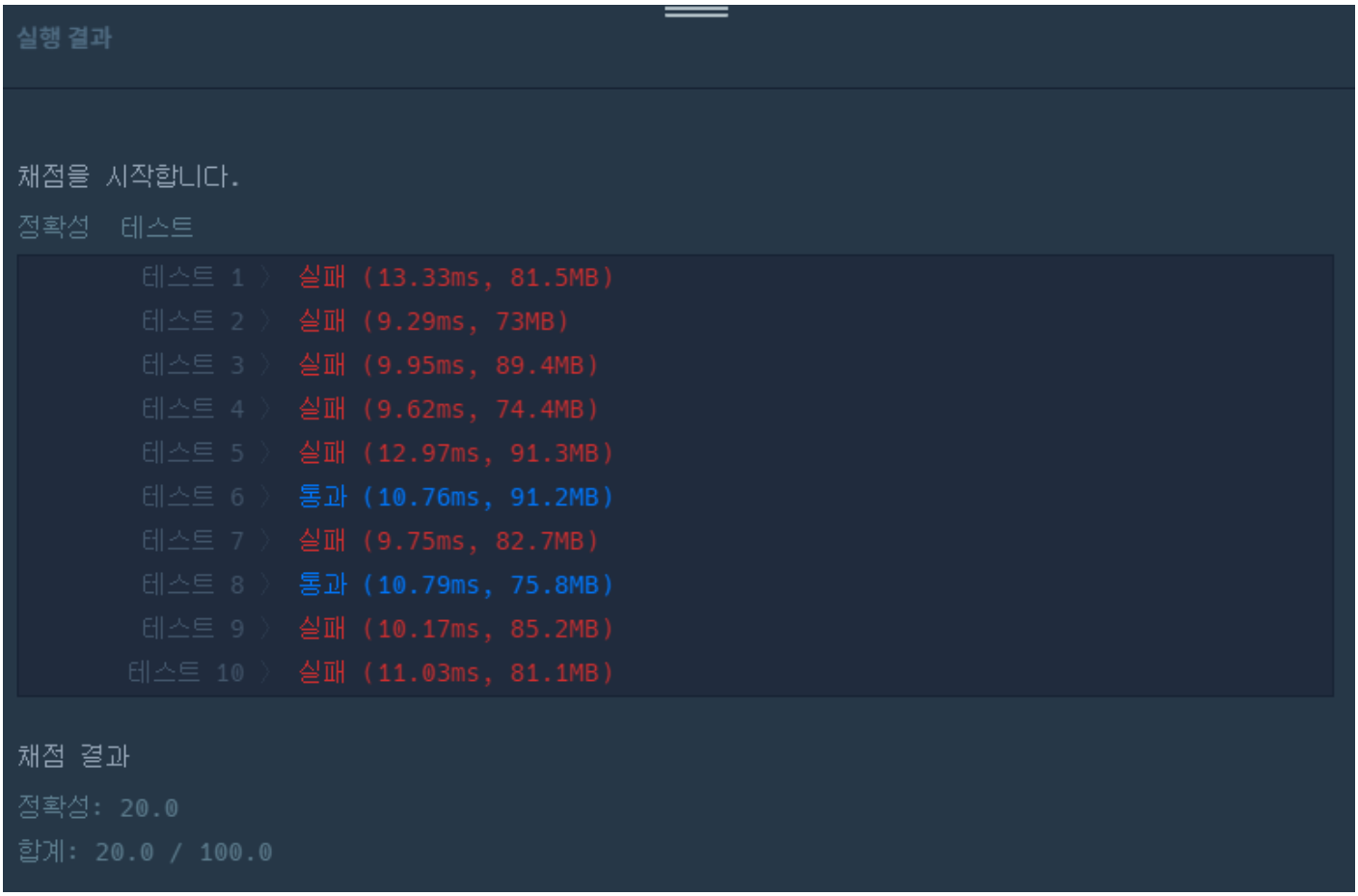
```
String my_string = "a12b3";
→ arr = ['a', '1', '2', 'b', '3']
→ 결과: 1 + 2 + 3 = 6 (□ □□ 12 + 3이 아님)
```

문제에서 원하는 것은 만약 "12 + 3" → 15로 계산하는 것이다.

String num 변수를 밖으로 빼고, Character.isDigit()으로 c가 0~9사이 숫자형 문자열이면 num에 저장하는 형식으로 코드를 짜 주었다.

틀린 코드 (3차 시도)

```
class Solution {
    public int solution(String my_string) {
        char[] arr = my_string.toCharArray();
        int sum = 0;
        String num = "";
        for (char c : arr ) {
```



```
            if (Character.isDigit(c)) {
                num += c;
            } else {
                if (!num.isEmpty()) {
                    sum += Integer.parseInt(num);
                    num = "";
                }
            }
        }
        if (!num.isEmpty()) {
            sum += Integer.parseInt(num);
        }
        return sum;
    }
}
```

실행 결과

```
채점을 시작합니다.
정확성  테스트
        테스트 1 〉  실패 (13.33ms, 81.5MB)
        테스트 2 〉  실패 (9.29ms, 73MB)
        테스트 3 〉  실패 (9.95ms, 89.4MB)
        테스트 4 〉  실패 (9.62ms, 74.4MB)
        테스트 5 〉  실패 (12.97ms, 91.3MB)
        테스트 6 〉  통과 (10.76ms, 91.2MB)
        테스트 7 〉  실패 (9.75ms, 82.7MB)
        테스트 8 〉  통과 (10.79ms, 75.8MB)
        테스트 9 〉  실패 (10.17ms, 85.2MB)
       테스트 10 〉  실패 (11.03ms, 81.1MB)

채점 결과
정확성: 20.0
합계: 20.0 / 100.0
```

덧셈만 하는게 아니라 뺄셈도 있었다 . 연산자 (+, -)에 따라 연산을 모두 처리하는 코드를 짜야 한다 .

1. 문자열을 " "(공백) 기준으로 나눠서
2. 처음 값은 result에 저장
3. 이후에는 연산자 (+ 또는 -)와 숫자가 번갈아 등장

4. 연산자에 따라 +면 더하고, -면 빼야 함

틀린 코드 (4차 시도)

```
class Solution {
    public int solution(String my_string) {
        String[] splited = my_string.split(" ");
        int answer = Integer.parseInt(splited[0]);

        for(int i = 1; i < splited.length; i += 2) {
            String op = splited[i];
            int num = Integer.parseInt(splited[i + 1]);

            if (op == '+')) {
                answer += num;
            } else if (op == '-') {
                answer -= num;
            }
        }
        return answer;
    }
}
```

? if(op == '+')

- 이 코드는 컴파일 에러가 난다.
- op는 String 타입으로 선언되었는데 '+'는 char로 타입이 달라서 비교 불가능하기 때문이다.
- Java에서는 문자열 내용을 비교할 때 ==를 쓰지 않는다. ==는 두 객체가 같은 주소(참조)를 가리킬 때만 true이다.
- 내용 비교는 반드시 .equals()를 써야 한다.

개선한 코드

```
class Solution {
    public int solution(String my_string) {
        String[] splited = my_string.split(" ");
        int answer = Integer.parseInt(splited[0]);
```

```
        for(int i = 1; i < splited.length; i += 2) {
            String op = splited[i];
            int num = Integer.parseInt(splited[i + 1]);

            if (op.equals("+")) {
                answer += num;
            } else if (op.equals("-")) {
                answer -= num;
            }
        }
        return answer;
    }
}
```

Revision #13
Created 19 May 2025 05:34:15 by Dain
Updated 23 May 2025 04:25:10 by Dain