

2024년 상반기 K-디지털 트레이닝

CLI Framework V1

[KB] IT's Your Life

1 CLI Framework V1

- ✓ **단인 책임의 원칙**
 - 기능별로 메서드화

```
import java.util.Scanner;

public class App {
    boolean run = true;
    int studentNum = 0;
    int[] scores = null;

    Scanner scanner = new Scanner(System.in);

    public void printMenu() {
        System.out.println("-----");
        System.out.println("1.학생수 | 2.점수입력 | 3.점수리스트 | 4.분석 | 5.종료");
        System.out.println("-----");
    }

    public int getSelect() {
        System.out.print("선택> ");
        return Integer.parseInt(scanner.nextLine());
    }
}
```

```
private void analyze() {
    int max = 0;
    int sum = 0;
    double avg = 0;
    for(int i=0; i<scores.length; i++) {
        max = (max<scores[i])? scores[i] : max;
        sum += scores[i];
    }
    avg = (double) sum / studentNum;
    System.out.println("최고 점수: " + max);
    System.out.println("평균 점수: " + avg);
}

public void getStudentNum() {
    System.out.print("학생수> ");
    studentNum = Integer.parseInt(scanner.nextLine());
    scores = new int[studentNum];
}
```

```
public void getScores() {
    for(int i=0; i<scores.length; i++) {
        System.out.print("scores[" + i + "]> ");
        scores[i] = Integer.parseInt(scanner.nextLine());
    }
}

public void printScore() {
    for(int i=0; i<scores.length; i++) {
        System.out.println("scores[" + i + "]: " + scores[i]);
    }
}

public void exit() {
    run = false;
}
```

```
public void executeCommand(int selectNo) {  
    if(selectNo == 1) {  
        getStudentNum();  
    } else if(selectNo == 2) {  
        getScores();  
    } else if(selectNo == 3) {  
        printScore();  
    } else if(selectNo == 4) {  
        analyze();  
    } else if(selectNo == 5) {  
        exit();  
    }  
}
```

```
public void run() {
    while(run) {
        printMenu();
        int selectNo = getSelect();
        executeCommand(selectNo);
    }

    System.out.println("프로그램 종료");
}

public static void main(String[] args) {
    App app = new App();
    app.run();
}
}
```

☑ 문제점

○ 단일 책임의 원칙 위배

- 메서드 레벨에서는 단일 책임의 원칙 준수
- 클래스 레벨에서는 단일 책임의 원칙 위배

○ 기능 확장 시 추가 수정 사항 발생

- 메뉴에 추가
- 새로운 기능 정의
- executeCommand()에 if문 추가

→ OCP(Open Close Principal) 위배