KB금융그룹 | 국민의 평생 금융파트너

2024년 상반기 K-디지털 트레이닝

CLI Framework V3 – Singleton, Command 패턴

[KB] IT's Your Life

KB 국민은행

1 CLI Framework V3

- 데이터 관리를 App로 부터 분리
  - 싱글톤 패턴 적용
- 각 명령을 클래스로 추상화
  - Command 패턴 적용

domain.StudentScores.java

```
public class StudentScores {
    int studentNum = 0;
    int[] scores = null;

    // Singleton 패턴
    private StudentScores() { }

    private static StudentScores instance = new StudentScores();

    public static StudentScores getInstance() {
        return instance;
    }
```

domain.StudentScores.java

```
    public int getStudentNum() {
        return studentNum;
    }

    public void setStudentNum(int studentNum) {
        this.studentNum = studentNum;
        this.scores = new int[studentNum];
    }

    public int[] getScores() {
        return scores;
    }
}
```

command.Command.java

```
public interface Command {
    void execute();
}
```

command.InitScoresCommand.java

```
public class InitScoresCommand implements Command{
    StudentScores studentScores = StudentScores.getInstance();

    @Override
    public void execute() {
        int studentNum = Input.getInt("학생수> ");
        studentScores.setStudentNum(studentNum);
    }
}
```

command.GetScoresCommand.java

```
public class GetScoresCommand implements Command{
    StudentScores studentScores = StudentScores.getInstance();

    @Override
    public void execute() {
        int [] scores = studentScores.getScores();

        for(int i = 0; i< scores.length; i++) {
            scores[i] = Input.getInt("scores[" + i + "]> ");
        }
    }
}
```

command.PrintScoreCommand.java

```
public class PrintScoreCommand implements Command{
    StudentScores studentScores = StudentScores.getInstance();

    @Override
    public void execute() {
        int [] scores = studentScores.getScores();

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

command.AnalizeCommand.java

```
public class AnalizeCommand implements Command{
    StudentScores studentScores = StudentScores.getInstance();

    @Override
    public void execute() {
        int [] scores = studentScores.getScores();
        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 / studentScores.getStudentNum();

        System.out.println("최고 점수: " + max);
        System.out.println("평균 점수: " + avg);
    }
}
```

command.ExitCommand.java

```
public class ExitCommand implements Command{
    @Override
    public void execute() {
        System.out.println("프로그램 종료");
        System.exit(0);
    }
}
```

App.java

```
public class App {
    Menu menu;
    Command[] commands;

    public App() {
        menu = new Menu();
        commands = new Command[] {
                new InitScoresCommand(),
                new GetScoresCommand(),
                new PrintScoreCommand(),
                new AnalizeCommand(),
                new ExitCommand()
        };
    }

    public void executeCommand(int selectNo) {
        Command command = commands[selectNo-1];
        command.execute();
    }

    public void run() {
        while(true) {
            menu.printMenu();
            int selectNo = menu.getSelect();
            executeCommand(selectNo);
        }
    }

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

문제점

기능이 추가된다면

- 추가 기능의 Command 구현체 작성

- 메뉴에 항목 추가
  - 현재는 고정되어 있음

- 메뉴와 Command가 분리되어 운영
  - 불일치 발생 가능