

2024년 상반기 K-디지털 트레이닝

CLI 프레임워크 라이브러리 임포트

[KB] IT's Your Life

- ✓ 새로운 프로젝트 생성
 - Name: StudentStore

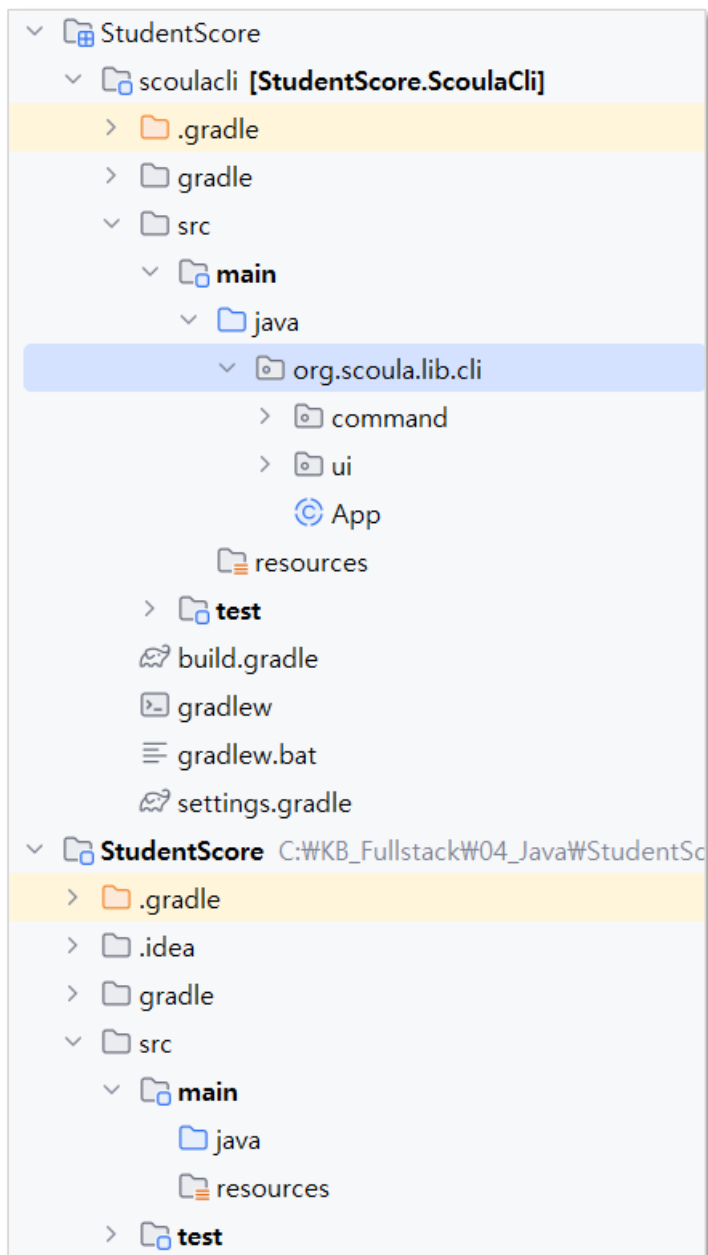
settings.gradle

```
rootProject.name = 'StudentScore'  
include ':ScoulaCli'  
project(':ScoulaCli').projectDir=new File('C:\\KB_Fullstack\\04_Java\\ScoulaLib\\scoulacli')
```

build.gradle

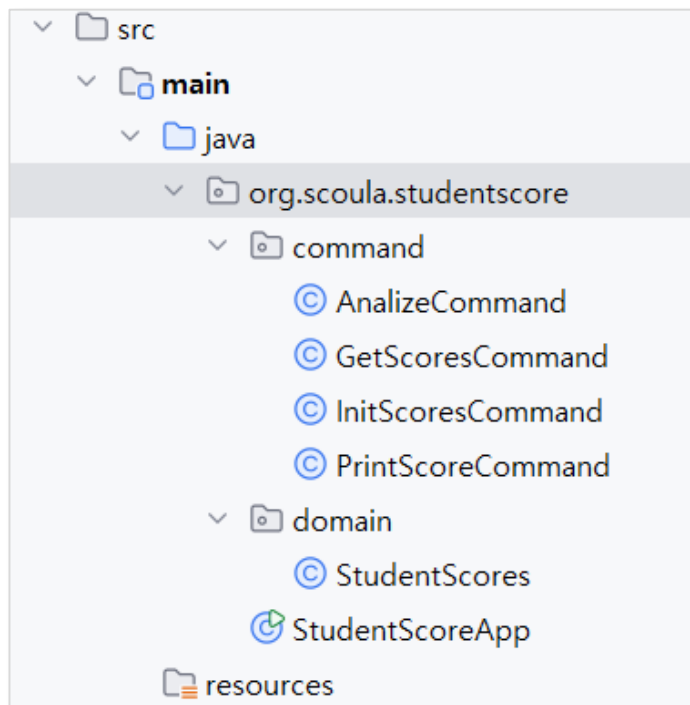
```
...  
dependencies {  
    implementation project(':ScoulaCli')  
  
    testImplementation platform('org.junit:junit-bom:5.9.1')  
    testImplementation 'org.junit.jupiter:junit-jupiter'  
}  
...
```

✓ 프로젝트 구조



✓ 어플리케이션 정의

- Command 구현체 정의
- App 클래스 상속
- createMenu() 오버라이드
- main() 메서드 작성
 - 어플리케이션 인스턴스 생성 후 run() 호출



StudentScoreApp.java

```
public class StudentScoreApp extends App {
    @Override
    public void createMenu(Menu menu) {
        menu.add(new MenuItem("학생수", new InitScoresCommand()));
        menu.add(new MenuItem("점수입력", new GetScoresCommand()));
        menu.add(new MenuItem("점수리스트", new PrintScoreCommand()));
        menu.add(new MenuItem("분석", new AnalyzeCommand()));
    }

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

☑ 문제점

- 너무 많은 Command 객체를 정의해야 함
→ 서비스 객체를 만들고, 메서드 참조를 이용해 메뉴에 설정

StudentScoreService.java

```
public class StudentScoreService {
    StudentScores studentScores = StudentScores.getInstance();

    public void initScores() {
        StudentScores studentScores = StudentScores.getInstance();

        int studentNum = Input.getInt("학생수> ");
        studentScores.setStudentNum(studentNum);
    }

    public void getScores() {
        StudentScores studentScores = StudentScores.getInstance();
        int [] scores = studentScores.getScores();

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


StudentScoreService.java

```
public void printScores() {  
    StudentScores studentScores = StudentScores.getInstance();  
    int [] scores = studentScores.getScores();  
  
    for(int i=0; i<scores.length; i++) {  
        System.out.println("scores[" + i + "]: " + scores[i]);  
    }  
}
```

StudentScoreService.java

```
public void analize() {
    StudentScores studentScores = StudentScores.getInstance();
    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);
}
}
```

StudentScoreApp.java

```
public class StudentScoreApp extends App {  
    StudentScoreService service = new StudentScoreService();  
  
    @Override  
    public void createMenu(Menu menu) {  
        menu.add(new MenuItem("학생수", service::initScores));  
        menu.add(new MenuItem("점수입력", service::getScores));  
        menu.add(new MenuItem("점수리스트", service::printScores));  
        menu.add(new MenuItem("분석", service::analyze));  
    }  
  
    public static void main(String[] args) {  
        App app = new StudentScoreApp();  
        app.run();  
    }  
}
```