

2024년 상반기 K-디지털 트레이닝

인터페이스

[KB] IT's Your Life

- ✓ **mywork 프로젝트 mylib 모듈에서 작업함.**
 - void execute() 메서드를 가지는 Command 인터페이스를 정의하세요.
 - 패키지: org.scoula.lib.cli.command

✓ 다음 조건을 만족하는 MenuItem 클래스를 정의하세요.

- 패키지: org.scoula.lib.cli.ui
- 멤버
 - String title
 - Command 인터페이스 참조
- 생성자
 - title과 Command 인터페이스를 매개변수로 가짐
- 메서드
 - String getTitle()
 - title을 리턴
 - void run()
 - 연관된 Command가 null이 아닌 경우 run() 메서드 호출

✓ 다음 조건을 만족하는 Menu 클래스를 정의하세요.

- 패키지: org.scoula.lib.cli.ui
- 멤버 변수
 - MenuItem 배열 list;
- 생성자
 - 매개변수 MenuItem 배열 크기
 - list 초기화
- 메서드
 - void add(int ix, MenuItem item)
 - 인덱스 ix에 메뉴(MenuItem)를 list에 추가
 - void print()
 - list에 담긴 메뉴를 출력,
 - 출력 완료 후 줄바꿈
1] 메뉴1 2] 메뉴2 3] 메뉴3 ...
 - MenuItem select()
 - Input을 이용해 사용자로부터 메뉴를 선택
 - 해당하는 MenuItem을 찾아 리턴
 - 잘못된 선택이면 null 리턴

✓ Application을 다음과 같이 수정하세요.

○ 멤버 변수

- Menu menu

○ 메서드

- createMenu()
 - Menu 인스턴스를 매개변수로 받음
- init()
 - Menu 인스턴스 생성 - 크기 6
 - createMenu() 호출 시 menu 전달
- run() 메서드의 무한 루프 수정
 - 메뉴 목록 출력
 - 메뉴 선택
 - 선택한 메뉴가 null이 아니면 해당 메뉴 실행
 - 메뉴 실행 후 줄바꿈

✓ 다음 클래스들을 정의하세요.

- 패키지: `package org.scoula.test.exm02.command;`
- 모두 Command 객체의 구현 클래스
- `TodoListCommand`
 - '[[Todo 목록 보기]]' 출력
- `TodoDetailCommand`
 - '[[Todo 상세보기]]' 출력
- `TodoCreateCommand`
 - '[[Todo 생성]]' 출력
- `TodoUpdateCommand`
 - '[[Todo 수정]]' 출력
- `TodoDeleteCommand`
 - 'Todo 삭제입니다' 출력
- `ExitCommand`
 - '종료합니다.' 출력 후 어플리케이션 종료(`System.exit(0);`)

✔ 다음과 같은 메뉴를 가지는 TestApplication 클래스를 정의하세요.

- 패키지: org.scoula.test.exm02
- Application 클래스 상속

순번	title	Command
0	목록	TodoListCommand
1	보기	TodoDetailCommand
2	생성	TodoCreateCommand
3	수정	TodoUpdateCommand
4	삭제	TodoDeleteCommand
5	종료	ExitCommand

✓ TestApplication을 운영하는 TestApplicationExample을 정의하세요.

○ 패키지: org.scoula.test.exm02

```
어플리케이션을 초기화합니다.  
1]목록 2]보기 3]생성 4]수정 5]삭제 6]종료  
선택> 1  
[[Todo 목록 보기]]
```

```
1]목록 2]보기 3]생성 4]수정 5]삭제 6]종료  
선택> 2  
[[Todo 상세 보기]]
```

```
1]목록 2]보기 3]생성 4]수정 5]삭제 6]종료  
선택> 6  
종료합니다.
```