

학습 목표

- Git, GitHub, IDE 등 실제 개발을 위한 환경에 익숙해진다.
 - 교육 분야에 맞는 프로그래밍 언어를 사용하여 간단한 문제를 해결한다.
-

프리코스 진행 방식

진행 방식

- 미션은 과제 진행 요구 사항, 기능 요구 사항, 프로그래밍 요구 사항 세 가지로 구성되어 있다.
- 세 개의 요구 사항을 만족하기 위해 노력한다. 특히 기능을 구현하기 전에 기능 목록을 만들고, 기능 단위로 커밋 하는 방식으로 진행한다.
- 기능 요구 사항에 기재되지 않은 내용은 스스로 판단하여 구현한다.
- 매주 진행할 미션은 화요일 오후 3시부터 확인할 수 있으며, 다음 주 월요일까지 구현을 완료하여 제출해야 한다. 제출은 일요일 오후 3시부터 가능하다.
 - 정해진 시간을 지키지 않을 경우 미션을 제출하지 않은 것으로 간주한다.
 - 종료 일시 이후에는 추가 푸시를 허용하지 않는다.

미션 제출 방법

- 미션 구현을 완료한 후 GitHub을 통해 제출해야 한다.
 - GitHub을 활용한 제출 방법은 [프리코스 과제 제출](#) 문서를 참고해 제출한다.
- GitHub에 미션을 제출한 후 [우아한테크코스 지원 플랫폼](#)에 PR 링크를 포함하여 최종 제출한다.
 - 자세한 안내는 [제출 가이드](#)를 참고한다.
 - 과제를 수행하면서 느낀 점, 배운 점, 많은 시간을 투자한 부분 등 자유롭게 작성한다.

과제 제출 전 체크 리스트

- 기능을 올바르게 구현했더라도 요구 사항에 명시된 출력 형식을 따르지 않으면 0점을 받게 된다.
- 기능 구현을 완료한 후 아래 가이드에 따라 모든 테스트가 성공적으로 실행되는지 확인한다.
- 테스트가 실패하면 점수가 0점이 되므로 제출하기 전에 반드시 확인한다.

테스트 실행 가이드

- 터미널에서 `java -version` 을 실행하여 Java 버전이 21인지 확인한다. Eclipse 또는 IntelliJ IDEA와 같은 IDE에서 Java 21로 실행되는지 확인한다.
- 터미널에서 Mac 또는 Linux 사용자의 경우 `./gradlew clean test` 명령을 실행하고, Windows 사용자의 경우 `gradlew.bat clean test` 또는 `./gradlew.bat clean test` 명령을 실행할 때 모든 테스트가 아래와 같이 통과하는지 확인한다.

문자열 덧셈 계산기

과제 진행 요구 사항

- 미션은 [문자열 덧셈 계산기](#) 저장소를 포크하고 클론하는 것으로 시작한다.
- 기능을 구현하기 전 `README.md` 에 구현할 기능 목록을 정리해 추가한다.
- Git의 커밋 단위는 앞 단계에서 `README.md` 에 정리한 기능 목록 단위로 추가한다.
 - [AngularJS Git Commit Message Conventions](#)을 참고해 커밋 메시지를 작성한다.
- 자세한 과제 진행 방법은 프리코스 진행 가이드 문서를 참고한다.

기능 요구 사항

입력한 문자열에서 숫자를 추출하여 더하는 계산기를 구현한다.

- 쉼표(,) 또는 콜론(:)을 구분자로 가지는 문자열을 전달하는 경우 구분자를 기준으로 분리한 각 숫자의 합을 반환한다.
 - 예: "" => 0, "1,2" => 3, "1,2,3" => 6, "1,2:3" => 6
- 앞의 기본 구분자(쉼표, 콜론) 외에 커스텀 구분자를 지정할 수 있다. 커스텀 구분자는 문자열 앞부분의 "/"와 "\n" 사이에 위치하는 문자를 커스텀 구분자로 사용한다.
 - 예를 들어 "/"\n1;2;3"과 같이 값을 입력할 경우 커스텀 구분자는 세미콜론(;)이며, 결과 값은 6이 반환되어야 한다.
- 사용자가 잘못된 값을 입력할 경우 `IllegalArgumentException` 을 발생시킨 후 애플리케이션은 종료되어야 한다.

입출력 요구 사항

입력

- 구분자와 양수로 구성된 문자열

출력

- 덧셈 결과

결과 : 6

실행 결과 예시

덧셈할 문자열을 입력해 주세요.

1,2:3

결과 : 6

프로그래밍 요구 사항

- JDK 21 버전에서 실행 가능해야 한다.
- 프로그램 실행의 시작점은 `Application` 의 `main()` 이다.
- `build.gradle` 파일은 변경할 수 없으며, 제공된 라이브러리 이외의 외부 라이브러리는 사용하지 않는다.
- 프로그램 종료 시 `System.exit()` 를 호출하지 않는다.
- 프로그래밍 요구 사항에서 달리 명시하지 않는 한 파일, 패키지 등의 이름을 바꾸거나 이동하지 않는다.
- 자바 코드 컨벤션을 지키면서 프로그래밍한다.
 - 기본적으로 [Java Style Guide](#)를 원칙으로 한다.

라이브러리

- `camp.nextstep.edu.missionutils` 에서 제공하는 `Console` API를 사용하여 구현해야 한다.
 - 사용자가 입력하는 값은 `camp.nextstep.edu.missionutils.Console` 의 `readLine()` 을 활용한다.