



# Validate Day00

🕒 Created @October 29, 2021 11:01 AM

## Introduction

---

평가 진행 시 다음과 같은 사항들을 지켜주십시오.

- 상호 간 존중하는 마음으로 예의를 지키며 평가를 진행합니다.
- 평가 진행 중 발생한 기능 장애들은 평가자와 함께 문제점을 밝혀내도록 하십시오. 이 과정에서 토론과 논의가 오갈 수 있습니다.
- 동료의 이해한 것과 다르게 이해할 수도 있다는 점을 배려하십시오. 열린 마음으로 평가에 임하는 것을 권장합니다.

## Guidelines

---

반드시 제출 시 이용되었던 Git 레포지토리를 기준으로 평가를 진행해야 합니다.

평가 진행 시 악의가 있는 별칭 등으로 결과를 조작하지는 않았는지 꼼꼼히 확인하십시오. 그리고 평가 진행 시 이용될 수 있는 스크립트에 대해서도 크로스 체크를 통해 별 문제가 없다는 것을 확인하여 당황스러운 상황을 피할 수 있도록 하십시오.

빈 레포지토리, (서브젝트에서 의도되지 않은) 기능 상에 문제가 있는 프로그램, 설명할 수 없는 코드가 포함된 치팅 등의 상황에서는 **FAIL**을 부여합니다. 특히 치팅에 대해선 경위와 사유를 듣고, 다음에도 같은 상황이 벌어지지 않도록 장려해주시요.

## General

- 제출된 레포지토리가 **gofmt**와 **golanci-lint**를 만족하는지 하십시오.

```
gofmt -l [filename]
golanci-lint run [filename]
```

- 제출된 레포지토리에 go.sum이 없거나 변조되어 모듈이 정상 작동하지 않을 수 있습니다. **go mod tidy**를 이용하여 모듈 정리를 진행 했음에도 문제가 있다면, **FAIL**입니다.
- 모든 기능들은 서브젝트에 명시된 \*\_test.go 파일을 이용하여 제출 이전에 검증되어 있어야 합니다. 테스트 파일이 잘 작성되어 있는지 확인하십시오. 테스트 케이스가 빈약하다면 평가자가 테스트 케이스를 요구할 수 있으며, 제시된 테스트 케이스가 정상적으로 작동하지 않는다면 **FAIL**입니다.

☒ YES

☒ No

## Validation

### 1) ex00

- go mod 내에 모듈 이름이 ex00으로 생성되어 있는지 확인하십시오.
- 평가자는 go mod를 이용하는 이유, GOPATH, GOROOT 등 프로젝트 관리를 위한 Golang의 설정들을 요구할 수 있으며, 평가 기준은 전적으로 평가자에게 맡깁니다.

```
ex00
└─ go.mod
```

☒ YES

☒ No

### 2) ex01

- go.mod 내에 모듈 이름이 ex01로 생성되어 있는지 확인하십시오.

- "Welcome to GoLand!\n"가 정상적으로 출력되는지 확인하십시오.
- fmt 혹은 내장 함수를 이용해선 안 됩니다. "bufio"와 "os"를 이용하여 코드가 작성되었는지 확인하십시오.
- 디렉토리 구조를 잘 확인하십시오. 특히 모듈 이름을 잘 확인하십시오.

```
ex01
├─ go.mod
└─ main.go
```

☒ YES

☒ No

### 3) ex02

- go.mod 내에 모듈 이름이 ex02로 생성되어 있는지 확인하십시오.
- korean.go , english.go , mandarin.go가 greet 디렉토리 안에 있는지 확인하고, 각 패키지 이름이 greet으로 작성되어 있는지 확인하십시오.
- 각 파일 내에 정의된 함수의 원형이 서브젝트에 명시된대로 구현되었는지 확인하십시오.
- 출력 결과에 대한 검증이 이뤄져야 합니다.
- 평가자는 원한다면 Golang과 UTF-8의 관계를 물을 수 있습니다.
- 디렉토리 구조를 잘 확인하십시오. 특히 모듈 이름을 잘 확인하십시오.

```
ex02
├─ go.mod
├─ main.go
└─ greet
    ├── korean.go
    ├── english.go
    └─ mandarin.go
```

☒ YES

☒ No

### 4) ex03

- go.mod 내에 모듈 이름이 ex03로 생성되어 있는지 확인하십시오.
- **go mod tidy**를 이용하여 외장 패키지의 불러오기가 잘 되는지 확인하십시오.
- 테스트 파일에 한국어, 영어, 한자로 된 인사 말의 검증이 잘 작성되어 있는지 확인하십시오.
- 피평가자는 평가자에게 테스트라는 것에 대해 설명할 수 있어야 합니다.
- 테스트의 결과가 **FAIL**이어선 안 됩니다.
- 디렉토리 구조를 잘 확인하십시오. 특히 모듈 이름을 잘 확인하십시오.

```
ex03
├─ go.mod
├─ main.go
├─ greet
│  └─ greet_test.go
│     └─ korean.go
│        └─ english.go
│           └─ mandarin.go
```

☒ **YES**

☒ **No**

## 5) ex04

- Number, Answer, Match의 선언이 서브젝트에서 명시한대로 이뤄졌는지 확인하십시오.
- 테스트 코드는 Number와 Match에 대해서만 확인하십시오.
- Number에서 파일을 읽을 때, Working Directory는 go.mod가 있는 경로를 기준으로 평가해야 합니다. 다른 경로에서는 정상적인 실행이 안 될 수도 있다는 점을 평가자는 이해해야 합니다. 단, 피평가자는 그 이유에 대해서 설명할 수 있어야 합니다.
- 피평가자는 Number의 인자로 상대 경로를 받는 이유를 테스트 코드와 연관지어 설명할 수 있어야 합니다.
- Answer의 기능에서 랜덤하게 생성된 숫자가 0~99 사이의 값을 내는지 로직과 결과를 확인하십시오.
- Match의 기능에서 range 반복문을 이용했는지 확인하십시오.
- 로또 당첨 숫자와 자신의 로또 숫자, 그리고 출력 문구에 대해서 적절하게 동작하는지 확인하십시오.

- 디렉토리 구조를 잘 확인하십시오. 특히 모듈 이름을 잘 확인하십시오.

```
ex04
├─ numbers.txt
├─ go.mod
├─ main.go
├─ lottery
│   ├── lottery.go
│   └─ lottery_test.go
└─ greet
    ├── greet_test.go
    ├── korean.go
    ├── english.go
    └─ mandarin.go
```

☒ YES


☒ No

## Ratings

☒ OK

★ Outstanding project

 Empty work

 Invalid program

 Cheat

 Crash