

프로그래밍 역량 강화 전문기관, 민코딩

Tennis KATA



Tennis Game

- ✓두 명의 Player가 게임을 진행한다.
- ✓테니스 경기 후, Score 결과를 알려주는 프로그램

✓TennisGame Class

- wonPoint(Player) : Player가 1 Point를 얻는다.
- getScore() : 현재 점수 결과를 String으로 return 한다.



소스코드 준비하기

✓사내링크

- <https://github.samsungds.net/cra1-sec/Tennis>

✓사외링크

- <https://github.com/mincoding1/Tennis>

테스트 게임 규칙 1

- ✓ 두 명의 플레이어가 공을 주고 받으면서,
공격 성공시 1 point를 획득한다.
- ✓ 점수가 4점 이상이면서,
상대방보다 2점 이상 더 높으면 게임에 승리한다.
- ✓ 각 점수를 부르는 명칭이 존재한다.
 - 0 point : "Love" (0점)
 - 1 point : "Fifteen" (15점)
 - 2 point : "Thirty" (30점)
 - 3 point : "Forty" (40점)

테스트 게임 규칙 2

✓ Deuce

- 두 플레이어 모두 3점 이상이면서, 점수가 같은 경우 Deuce
- 즉, 0-0 / 1-1 / 2-2 는 Deuce가 아니며 3-3 부터 동점은 Deuce 이다.

✓ Advantage Player

- Deuce 상태에서 1점을 더 낸 사람은, Advantage Player 이다.
(두 플레이어 모두 3점 이상이면서, 점수가 1점 더 높은 Player를 뜻함)
- Advantage Player가 1점 더 취득하면, 게임에 승리하게 된다.

테스트 게임 규칙 3

✓All

- 같은 점수일 때는 점수 + 'All' 으로 표현한다.
- 0:0 은 Love-All
- 15:15는 Fifteen-All
- 30:30은 Thirty-All
- 단, 40:40은 Forty-All이 아니라 Deuce
 - Forty-All 은 존재하지 않는다.

규칙 예시 1

✓ player1, player2가 경기중이고, player 1은 항상 서브권을 갖는다.

1. p1 0 / p2 15
→ Love-Fifteen
2. p1 15 / p2 15
→ Fifteen-All
3. p1 30 / p2 15
→ Thirty-Fifteen
4. p1 30 / p2 30
→ Thirty-All
5. p1 40 / p2 30
→ Forty-Thirty

이 프로그램은 **Player1은 항상 Server**이고,
Player2가 항상 Receiver 임을 가정한다.

실제 테니스 경기는
Server-Receiver 의 순서대로 점수를 표현한다.

규칙 예시 2

✓ player1, player2가 경기중이고, player 1은 항상 서브권을 갖는다.

1. p1 40 / p2 40
→ Deuce

2. p1 40 / p2 40 Ad
→ Advantage player2

3. p1 40 / p2 40
→ Deuce

4. p1 40 Ad / p2 40
→ Advantage player1

5. p1 40 Win / p2 30
→ Win for player1



[도전] 클린 코드 변경 시작!

- ✓[미션 1] 코드를 변경했을 때 입출력이 달라지지 않도록 기존 코드의 입출력을 확인하는 코드를 준비한다.
- ✓[미션 2] Method 추상화로 Clean Code를 만든다.