

**1. TennisScoreCounter (테니스 경기 계수기)**

| **필드명** | **타입** | **설명** |
| --- | --- | --- |
| setCount | int | 경기의 세트 수 (3세트 또는 5세트) |
| player1 | Player | 첫 번째 선수 객체 |
| player2 | Player | 두 번째 선수 객체 |
| currentSet | int | 현재 진행 중인 세트 번호 |
| gameScore | Map<Integer, int[]> | 각 세트별 두 선수의 게임 승리 수 저장 (Key: 세트 번호, Value: [선수1 게임 수, 선수2 게임 수]) |

🔹 **설명:**

* 경기의 세트 수와 두 선수의 정보를 저장하는 핵심 클래스입니다.
* gameScore는 세트별 승리한 게임 수를 저장하며, currentSet은 현재 진행 중인 세트를 추적합니다.

**2. Player (선수)**

| **필드명** | **타입** | **설명** |
| --- | --- | --- |
| name | String | 선수 이름 |
| gamePoint | int | 현재 게임에서 획득한 포인트 (0, 15, 30, 40) |
| setWins | int | 해당 선수가 획득한 세트 승리 개수 |

🔹 **설명:**

* 개별 선수 정보를 저장하는 클래스입니다.
* gamePoint는 현재 진행 중인 게임의 포인트(0, 15, 30, 40)를 나타냅니다.
* setWins는 해당 선수가 획득한 세트 수를 추적합니다.

3. Match (경기 진행)

| **필드명** | **타입** | **설명** |
| --- | --- | --- |
| matchType | int | 경기 유형 (3세트 경기: 3, 5세트 경기: 5) |

🔹 **설명:**

* matchType을 통해 3세트 경기인지 5세트 경기인지 구분합니다.
* startMatch()를 호출하면 경기 진행을 시작합니다.
* isGameOver()는 경기 종료 여부를 확인하는 기능을 합니다.

**4. ScoreBoard (스코어보드)**

| **필드명** | **타입** | **설명** |
| --- | --- | --- |
| 없음 | 없음 | (필드 없이 점수 출력 기능만 포함) |

🔹 **설명:**

* dispScore() 메서드를 통해 현재 스코어보드를 출력하는 역할을 합니다.
* 테니스 경기의 점수 형식을 고려하여 점수를 표시합니다.

**5. FileManager (파일 저장 및 불러오기)**

| **필드명** | **타입** | **설명** |
| --- | --- | --- |
| 없음 | 없음 | (필드 없이 파일 입출력 기능만 포함) |

🔹 **설명:**

* saveToFile()는 경기 결과를 파일에 저장합니다.
* loadFromFile()는 저장된 경기 데이터를 불러오는 기능을 합니다.

| **클래스명** | **역할** |
| --- | --- |
| **TennisScoreCounter** | 경기 점수 관리 (핵심 로직) |
| **Player** | 선수 정보 및 점수 저장 |
| **Match** | 경기 진행 및 종료 판별 |
| **ScoreBoard** | 경기 점수 출력 |
| **FileManager** | 경기 결과 파일 저장/불러오기 |