

BE 도메인 설계

필요한 메서드

- 인원수, 말 개수, 판 모양 받아와서 윷놀이 판 세팅
- 턴 사용자 명시
- 랜덤 윷 결과값 리턴 및 저장
- 지정 윷 결과값 저장
- 나온 윷 결과에 따라 말 이동할 지점 계산해서 리턴
- 말 이동하면 말 위치 정보와 판의 지점 정보 업데이트
- 말 이동하고 나면 해당 윷을 윷 결과 리스트에서 제거
- 윷 결과 리스트에 있는 거 다 사용했는지, 즉 턴이 끝났는지 확인
- 말 하나 골인하면 사용자의 말 정보 업데이트
- 특정 사용자의 말이 모두 골인했는가?(=승리했는가) 확인
- 전체 정보 초기화(한 판 더 대비)
- 이동한 지점에 같은 편 말이 있을 경우 업기
- 이동한 지점에 다른 편 말이 있을 경우 잡기(턴 추가 부여, 잡힌 말 위치 정보 초기화)
- 한 번 더 던지기 조건 체크(윷, 모 개수+추가 턴 기회)

GameManager나 Game이 필요할 듯

도메인

- GameManager
 - Board
 - Player
-
- 인원수, 말 개수, 판 모양 받아와서 윷놀이 판 세팅
 - 턴 사용자 명시
 - 윷 던지기
 - 나온 윷 결과에 따라 말 이동할 지점 계산해서 리턴

- 말 이동
 - 턴 관리
 - 말 하나 골인하면 사용자의 말 정보 업데이트
 - 특정 사용자의 말이 모두 골인했는가?(=승리했는가) 확인 ; 승리 조건 체크
 - 전체 정보 초기화(한 판 더 대비) → 각 클래스 리셋 함수 사용
 - 한 번 더 던지기 조건 체크(윷, 모 개수+추가 턴 기회)
 - 전반적인 상태 리턴(이걸 뷰에 넘기면서 업데이트)
- Player
 - id ; 순서용
 - Piece ; 보유 말 정보

 - 보유 말 정보 업데이트
 - 보유 말 정보 리턴
 - 리셋
- Piece
 - position ; 위치 정보
 - Player owner ; 소속 정보

 - 말 이동하면 위치 정보 업데이트
 - 리셋
- Board
 - <Cell> cells ; 지점 정보

 - 말 이동 하면 판 지점 정보 업데이트
 - 말 이동 가능 지점 계산
 - 도착 처리
 - 이동한 지점에 같은 편 말이 있을 경우 업기

- 이동한 지점에 다른 편 말이 있을 경우 잡기(턴 추가 부여, 잡힌 말 위치 정보 초기화)
 - 리셋
- Cell
 - id ; 고유 아이디, 인덱스
 - <Cell> nextCell ; 경로 탐색을 위해서 이동 가능한 지점 저장 → Cell? Int?
⇒ Linked-list 방식으로 최종 결정
 - type ; 유형 - enum(시작, 종료, 분기점)
 - <Piece> withPieces ; 엮은 말들

 - 다음 지점 리턴(시작 위치도 넣게 해서 분기점에서 출발할 때랑 분기점 지나쳐 갈 때 구분)
 - 말 들어오시고
 - 말 나가시고
 - 내 말 정보 리턴
 - 상대 말 정보 리턴
 - 리셋
- Yut
 - resultList ; 결과 저장 - enum(뺨도, 도, 개, 걸, 윷, 모)

 - 랜덤 윷 결과값 리턴 및 저장
 - 지정 윷 결과값 저장
 - 말 이동하고 나면 해당 윷을 윷 결과 리스트에서 제거
 - 윷 결과 리스트에 있는 거 다 사용했는지, 즉 턴이 끝났는지 확인○
 - 리셋

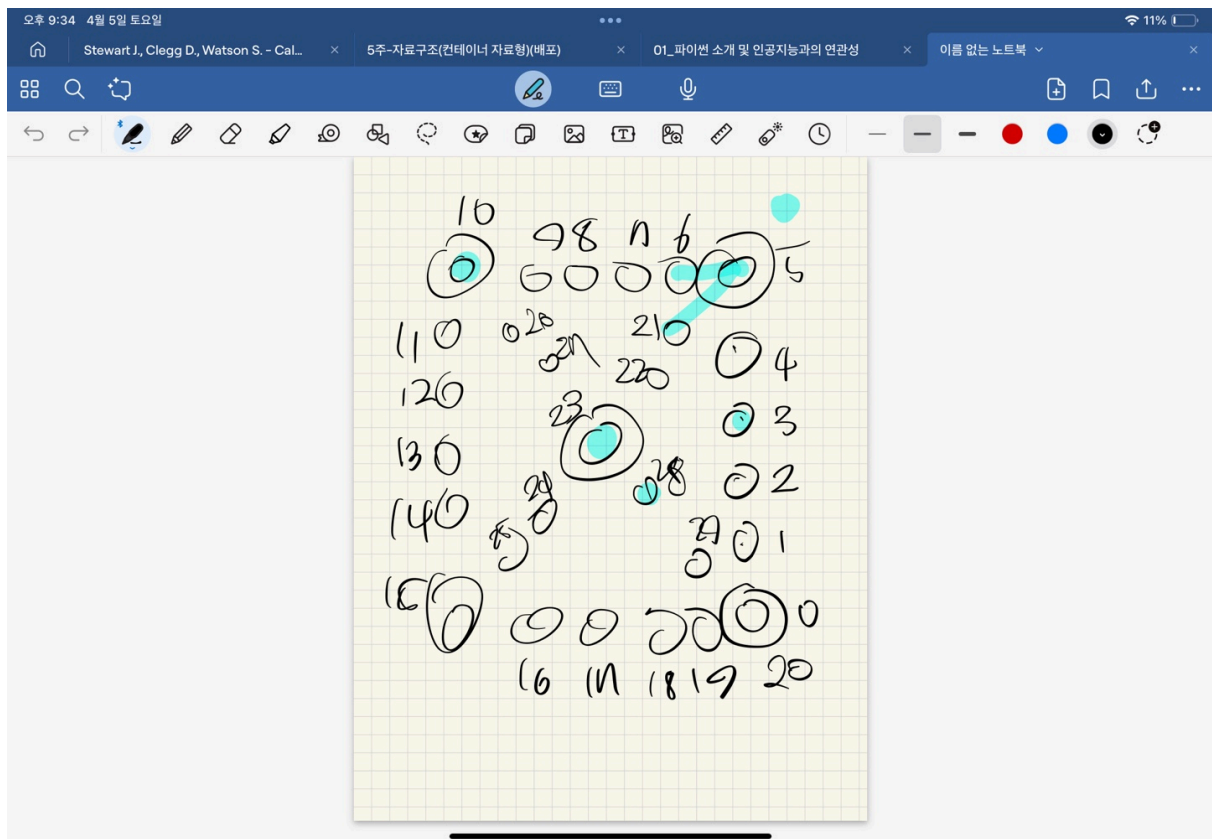
GameManager vs Controller

근데 다시 컨트롤러와 GameManager 차이로 돌아가서, 내가 이해한 건 Controller가 View로부터 입력 데이터를 받아서 이거 해줘! 하면 GameManager가 내부 진행을 해서 결과를 알려주면 그걸 다시 Controller가 View로 전달한다는 건데, 그럼 Controller는 GameManager에 있는 함수만 쓰는거야?

맞아요!! 당신이 지금 이해한 그 구조가 **MVC의 정석적 흐름**이에요 ¹⁰⁰

→ "Controller는 View로부터 입력을 받아 → GameManager에게 넘기고 → 결과를 다시 View에게 전달한다."

문제점



- nextCell을 파악하기 위해서는 이전 경로, 지름길 사용 여부를 알아야 함
 - 27→23 의 경우에는 기본길이 28인데, 22→23 의 경우는 기본길이 24, 지름길이 28 임
 - = 즉, 경우에 따라 기본길과 지름길이 달라지는 경우 대비 부족
- ⇒ 일단 중앙 지점(23)을 별도의 type으로 두고, 이에 맞춰 로직을 구성하기로 함