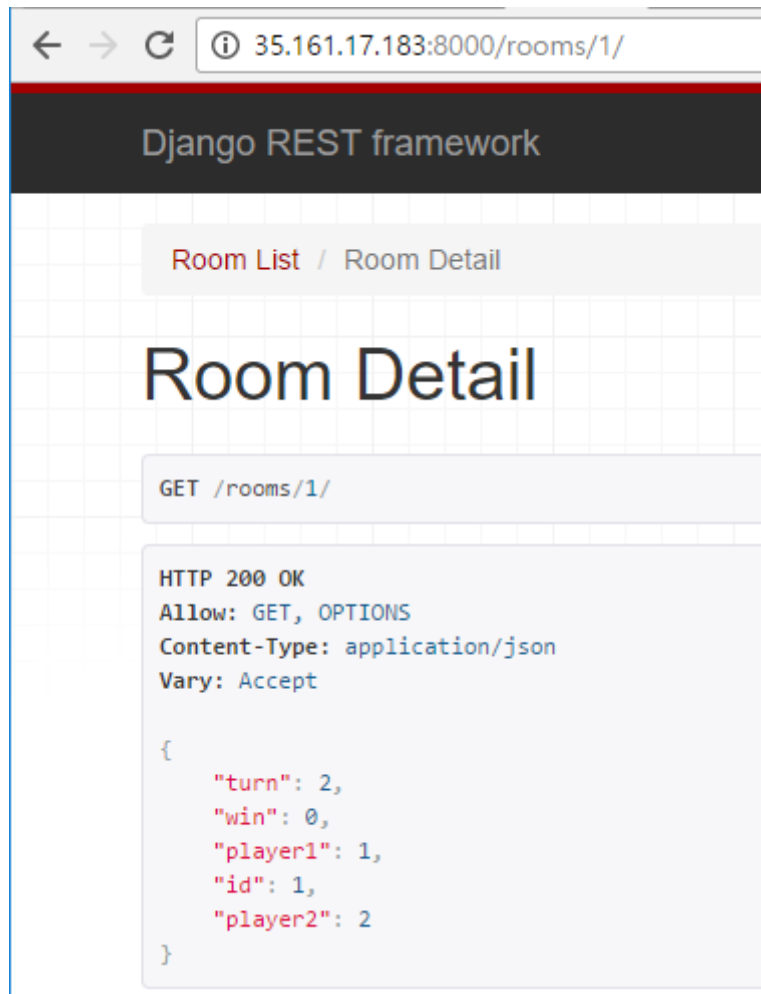
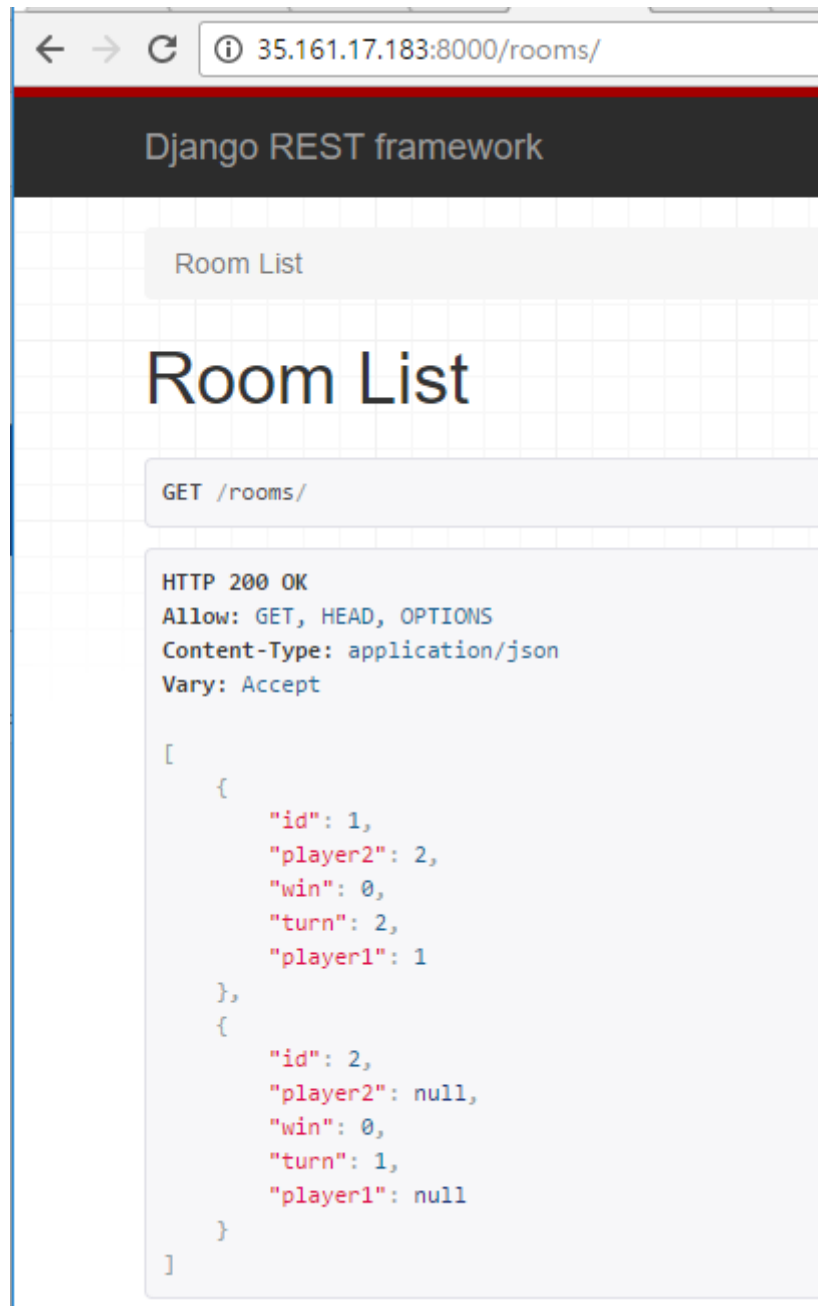


1. GET rooms/id



- "turn" : 값이 1이면 다음 턴에 첫번째 유저가, 2이면 두번째 유저가 돌을 놓습니다.
- "id" : 이 방의 id입니다.
- "player1" : 첫번째 유저의 id (아직 입장하지 않았으면 null (python에서는 None 으로 나타냄))
- "player2" : 두번째 유저의 id (아직 입장하지 않았으면 null (python에서는 None 으로 나타냄))
- "win" : 아직 승부가 나지 않았으면 0, 첫번째 유저가 이겼으면 1, 두번째 유저가 이겼으면 2
- 위에 명시한 속성 이외에, 구현의 편의를 위해 다른 속성을 추가해도 괜찮습니다. (예 : 현재의 바둑판 모양을 나타내는 속성 등)

2. GET rooms/

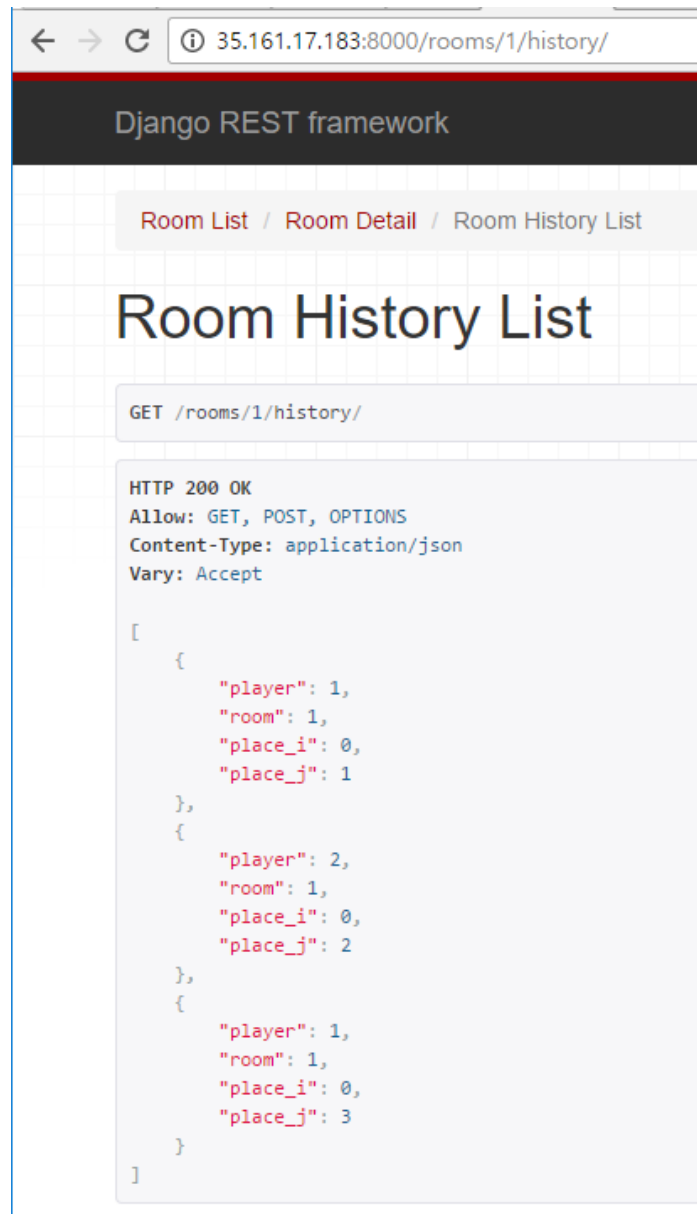


- 존재하는 room들에 대한 `GET rooms/id/` 결과를 리스트 형태로 return한다.

3. POST rooms/

- POST를 통해 새로운 room을 생성할 때엔, 방에 대한 아무런 정보를 주지 않을 것입니다. 새로 생성된 방의 각 속성은 아래와 같은 값을 갖도록 합니다.
 - "player1", "player2"의 기본값 : null (python에선 None 으로 나타냄)
 - "win"의 기본값 : 0 (아직 승리가 결정되지 않음)
 - "turn"의 기본값 : 1 (player1 이 시작)

4. GET rooms/id/history

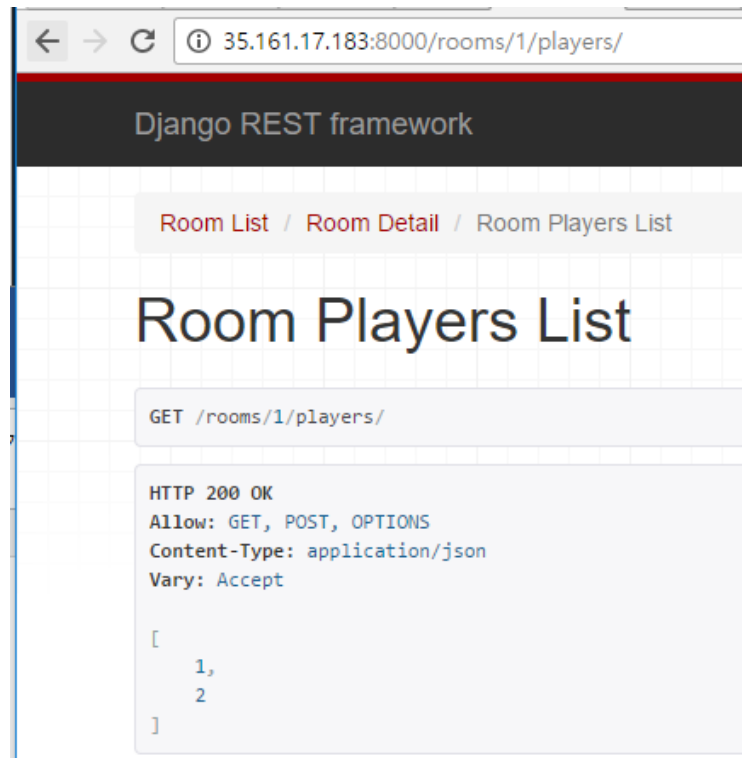


- "player" : 바둑돌을 놓은 유저의 id 값을 가집니다.
- "room" : 게임 방의 id 값을 가집니다.
- "place_i" 속성 : 유저가 놓은 바둑돌의 y값입니다.(0부터 시작)
- "place_j" 속성 : 유저가 놓은 바둑돌의 x값입니다.(0부터 시작)
- 위에 명시한 속성 이외에, 구현의 편의를 위해 다른 속성을 추가해도 괜찮습니다.

5. POST rooms/id/history

- POST 할 데이터로는 "place_i", "place_j" 속성의 값이 주어집니다. "player"는 현재 로그인한 사람, "room"은 url의 id를 자동으로 채워넣도록 합니다.

6. GET /rooms/id/players



- 현재 방에 입장한 유저들의 id의 목록을 반환합니다. 한 방에는 최대 2명만 입장할 수 있으므로, 목록의 최대 길이는 2가 됩니다.
- 목록의 첫번째 원소는 `GET rooms/id` 결과의 "player1" 와 동일합니다.
- 목록의 두번째 원소는 `GET rooms/id` 결과의 "player2" 와 동일합니다.

7. POST /rooms/id/players

- POST 할 데이터는 텅 빈 dictionary 값이 주어집니다.

| | |
|-------------|------------------|
| Media type: | application/json |
| Content: | { } |