|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **주차** | 7주차 | **기간** | 2024.11.19 ~2024.11.26 | **지도교수** | 이형구 교수님 |
| 이번주 한일 요약 | 연결된 클라이언트 그리기 | | | | |

서버에 연결된 클라이언트를 서버 화면에 띄우는 작업을 진행했습니다.

클라이언트 작업

텍스트, 스크린샷, 디스플레이이(가) 표시된 사진

자동 생성된 설명

1. 매 프레임마다 클라이언트 캐릭터의 위치값과 회전값을 저장하고 send하도록 한다.  
(이 과정도 비동기적 함수로 돌리거나 몇프레임마다 보낼지 정확하게 정해야 할 것 같다.)

텍스트, 스크린샷, 디스플레이, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

2. 받은 캐릭터 정보를 send하는 함수이다.

서버 작업

1. Client 접속 성공시 ReceiveData() 함수를 비동기적으로 실행시킵니다.

텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

2. 받은 데이터가 있다면, Buffer에 받은 값을 캐릭터 위치와 회전값을 가지는 FCharacterState에   
각각 변환하여 저장합니다.

3. 받은 값을 사용하여 SpawnOrUpdateClientCharacter() 함수를 수행합니다.  
텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

4. ClientCharacter가 존재하지 않는다면, ClientCharacter 캐릭터를 생성하고,   
ClientCharacter가 이미 있다면 ClientCharacter에게 받은 값으로 캐릭터 위치와 회전을 업데이트 해줍니다.  
5. 여기서 ClientCharacter는 캐릭터를 부모 클래스로 상속받습니다.  
GetMesh로 받은 SkeletalMesh를 이용하여 캐릭터를 그립니다.  
텍스트, 스크린샷, 디스플레이, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

구름, 하늘, 스크린샷, 야외이(가) 표시된 사진

자동 생성된 설명

위 과정으로 서버에 클라이언트가 접속하면 클라이언트 회전과 위치에 맞춰 캐릭터를 화면에 그립니다. 하지만 아직 캐릭터의 위치와 회전만 받아오기 때문에 제대로된 출력이 아닙니다.

|  |  |  |  |
| --- | --- | --- | --- |
| **문제점 정리** | 캐릭터 출력에 필요한 값이 부족하다. | | |
| **해결방안** | 무엇을 더 알아야 제대로된 애니메이션이 가능한지 찾아보자. | | |
| **다음주차** | 8주차 | **다음기간** | 2024.11.26  ~2024.12.03 |
| **다음주 할 일** | 화면에 보이는 캐릭터를 더 자세하게 그릴 수 있도록 한다. | | |
| **지도 교수**  **Comment** |  | | |