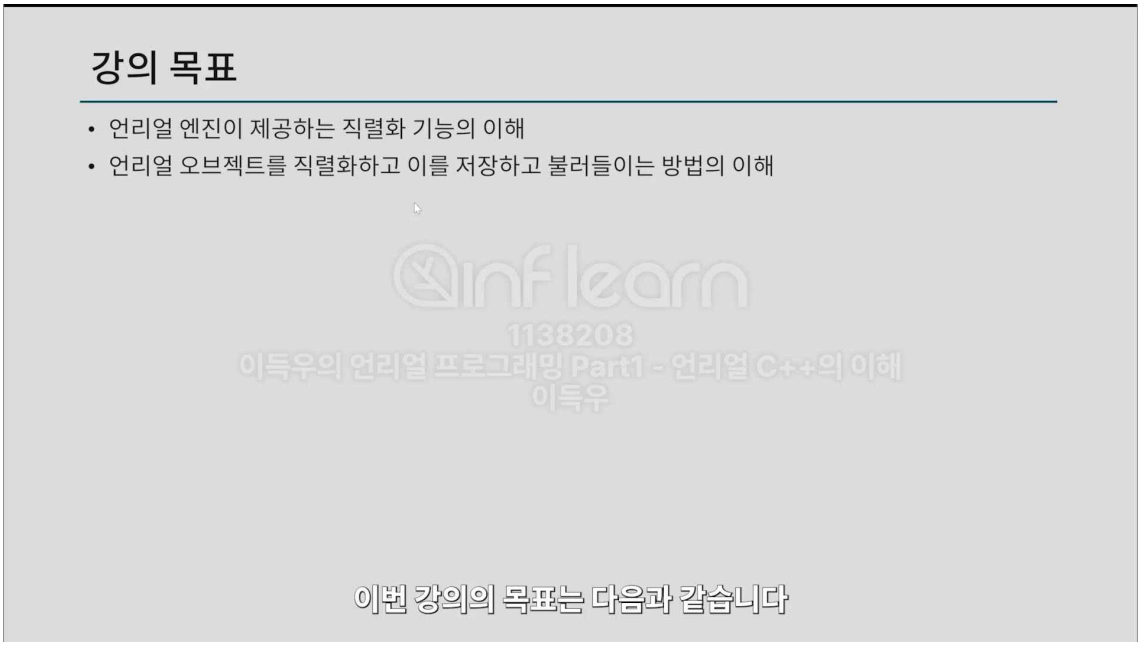
언리얼 프로그래밍 Part1-13

제목:언리얼 오브젝트 관리 I - 직렬화

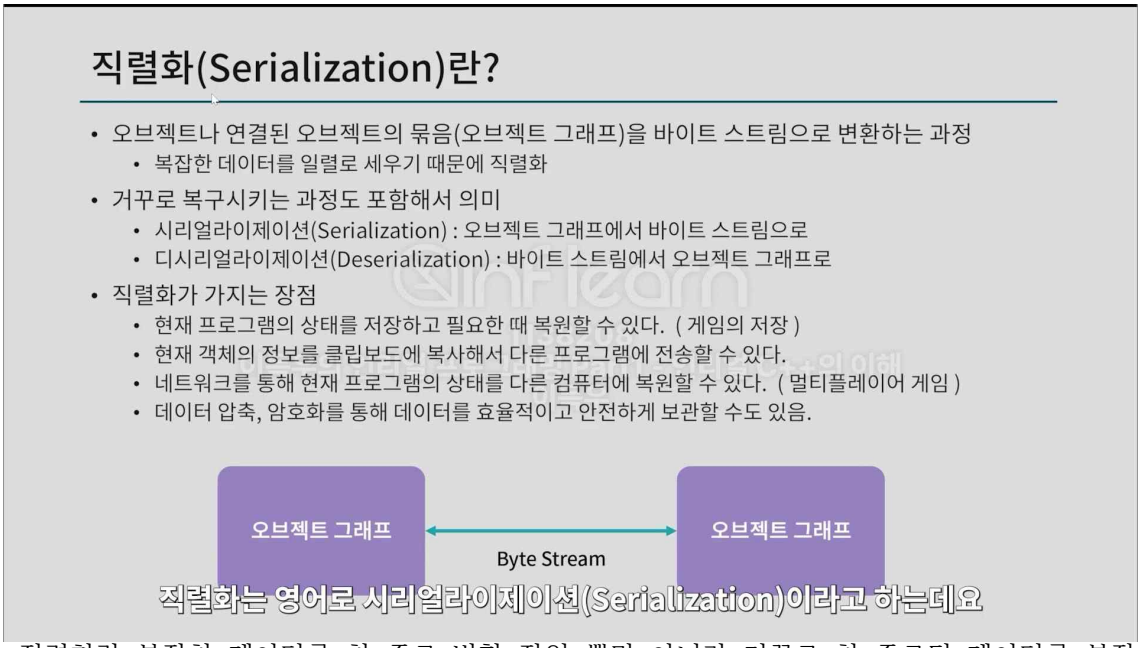
**강의 내용 : 언리얼 엔진이 제공하는 직렬화 기능을 이해하고 언리얼 오브젝트의 데이터를 저장하고 불러들이기

**강의 목표 :



***언리얼 엔진의 직렬화

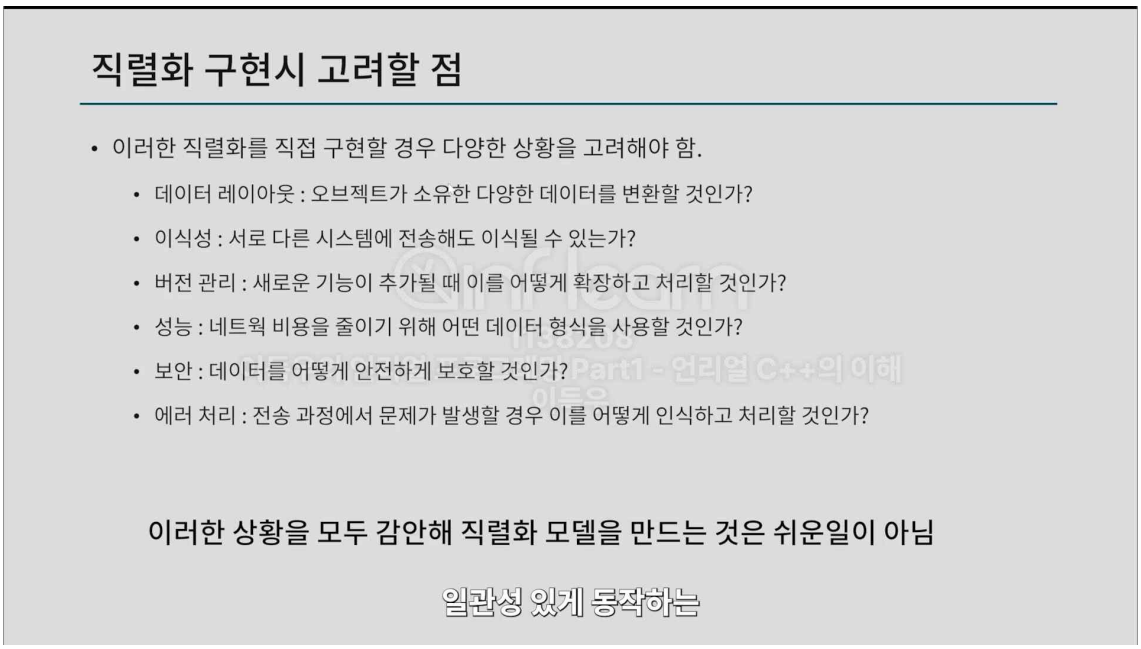
**직렬화(Serialization)란?



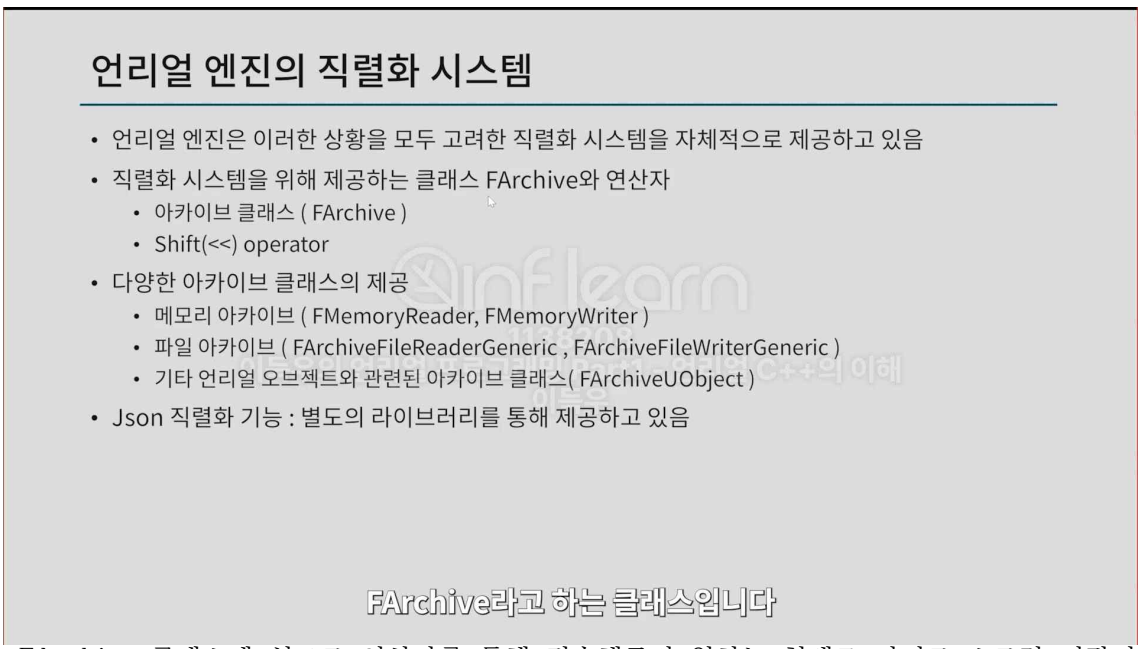
-직렬화란 복잡한 데이터를 한 줄로 변환 작업 뿐만 아니라 거꾸로 한 줄로된 데이터를 복잡한 오브젝트로 복구하는 작업까지 포함한다.

-복잡한 오브젝트들 간에 데이터를 연동할 때 한줄로된 바이트스트림으로 변환하거나 복구하는 작업을 직렬화라고 한다.

**직렬화 구현시 고려할 점



**언리얼 엔진의 직렬화 시스템



-FArchive 클래스에 쉬프트 연산자를 통해 전송해주면 원하는 형태로 바이트 스트림 저장이 가능하다.
-FArchive를 상속받는 다양한 아카이브 클래스를 제공한다.(사진에서 참고)

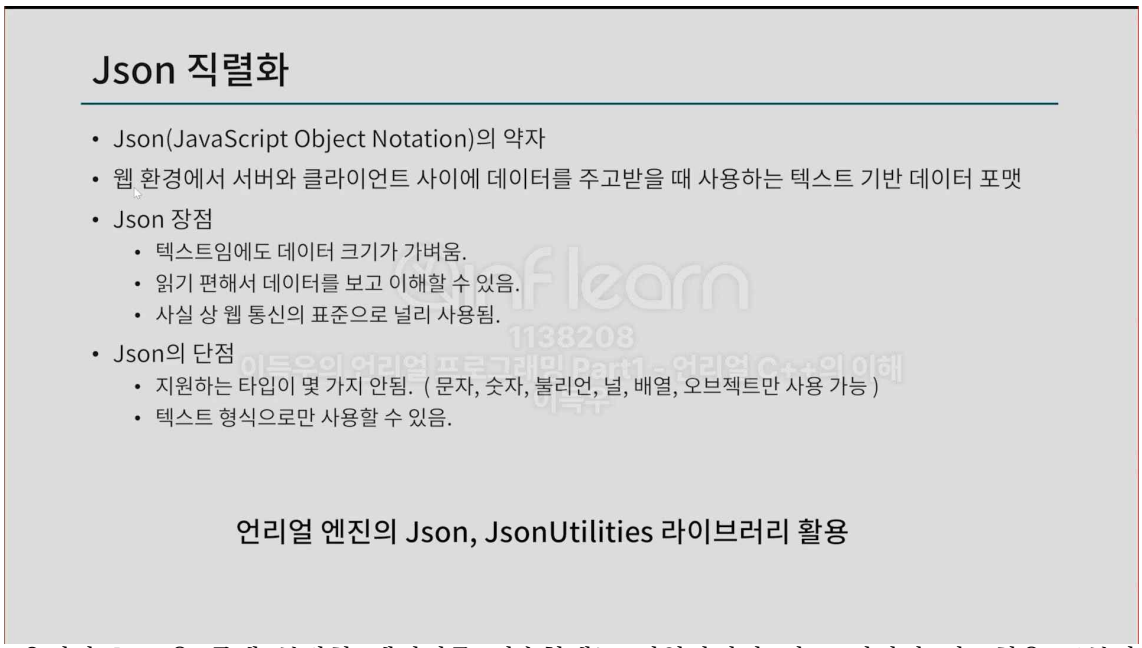
**실습
-Student클래스처럼 언리얼엔진 오브젝트의 기본 설정을 맞춰주면 저장하거나 불러들일 때 즉 직렬화를 진행할 때 시리얼 라이즈 함수를 오버라이드 해서 구현해주면 된다.

-언리얼 오브젝트에서 시리얼라이즈 함수가 구현이 되어 있다.

-파일을 읽고 저장할때 번거로운 delete와 null값을 사용하는데이를 한번에 처리 가능한 스마트포인터를 사용해보자(TUniquePtr)

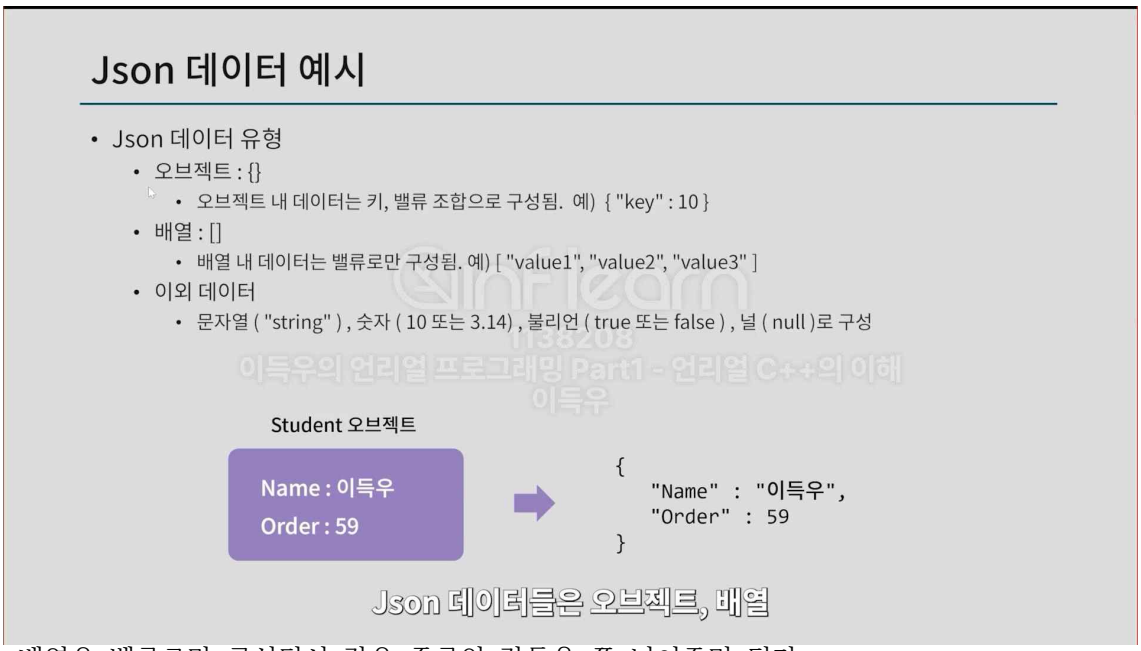
-직렬화 데이터들은 Saved에 ObjectData.bin, RawData.bin에 저장되어 있다.

**Json 직렬화



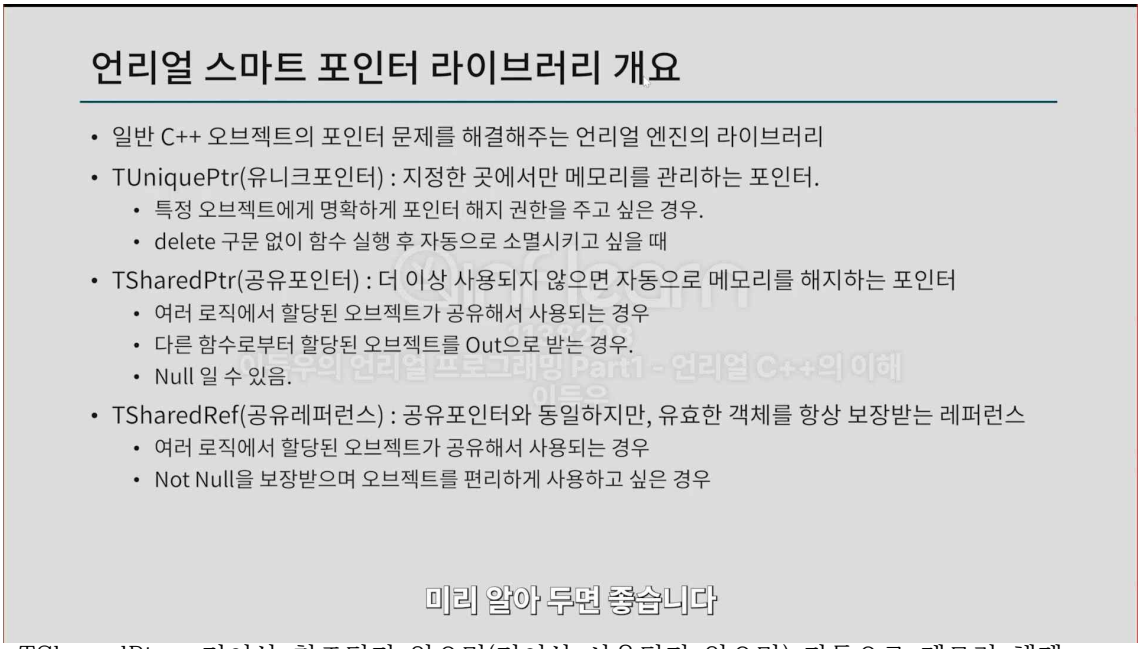
-우리가 Json을 통해 복잡한 데이터를 전송할때는 지원타입이 별로 없어서 자료형을 구분하기가 불가능하다 따라서 우리가 데이터를 받아 적절히 처리해야 한다.
-텍스트로만 처리가 가능하여 네트워크 상에서 극도의 효율을 추구할때는 단점으로 적용된다.

**Json데이터 예시



-배열은 벨류로만 구성되서 같은 종류의 값들을 쭉 넣어주면 된다.

**언리얼 스마트 포인터 라이브러리 개요



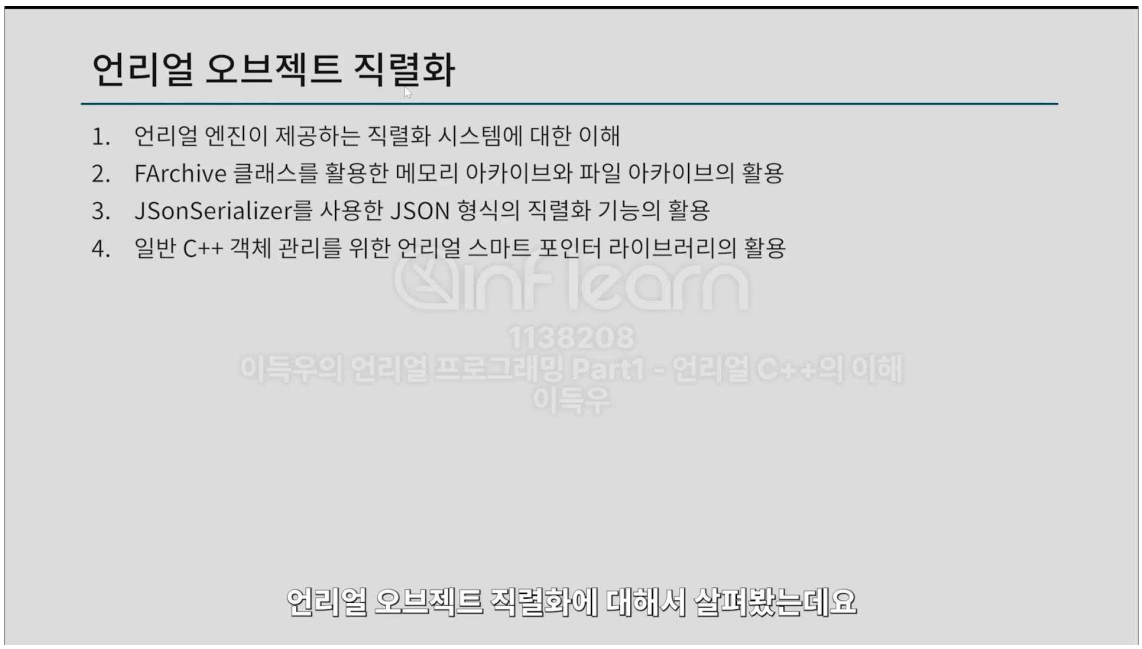
-TSharedPtr : 더이상 참조되지 않으면(더이상 사용되지 않으면) 자동으로 메모리 해제

-유니크포인터의 경우 어떤 특정 오브젝트에게 해당 오브젝트의 포인터 메모리 해지 권한을 주고싶을 때 사용하면 된다.
-유니크 포인터를 사용해서 해당 동적 할당된 오브젝트를 자동으로 소멸시키고 싶을 때 사용해주면 좋다.

-공유 포인터의 경우 여러 로직에서 이 공유포인터가 가리키는 동적 할당된 오브젝트를 사용할 때 그 오브젝트가 더 이상 사용되지 않으면 자동으로 소멸되도록 지정할 때 유용하다.
-공유 포인터는 기본 값이 널값으로 진행을 하는 데 다른 함수로부터 할당된 오브젝트를 Out으로 받고싶은 경우 공유 포인터를 사용해주면 된다.

-공유 레퍼런스는 공유포인터와 다르게 레퍼런스 방식으로 동작하기에 Not Null을 보장받을 수 있어서 널인지 아닌지 점검할 필요가 없다.
-공유 레퍼런스가 사용이 간편하고 편리하다.

**정리


언리얼 오브젝트 직렬화
1. 언리얼 엔진이 제공하는 직렬화 시스템에 대한 이해
2. FArchive 클래스를 활용한 메모리 아카이브와 파일 아카이브의 활용
3. JSonSerializer를 사용한 JSON 형식의 직렬화 기능의 활용
4. 일반 C++ 객체 관리를 위한 언리얼 스마트 포인터 라이브러리의 활용
inflearn
1138208
이득우의 언리얼 프로그래밍 Part1 - 언리얼 C++의 이해
이득우
언리얼 오브젝트 직렬화에 대해서 살펴봤는데요