|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **주차** | 3 주차 | **기간** | 23.07.10~ 23.07.16 | **지도교수** | (서명) |
| 이번주 한일 요약 | 1. 전문가를 위한 C++책(C++ 공부1)  2. 언리얼 소스코드 분석(Actor) (x)  3. 게임 프로그래밍 패턴 공부(책) | | | | |

작성자(학번 이름): 2018182037 조상준

팀명:

<알아두면 좋을 내용>

재사용 철학

1 작성은 한번, 사용은 여러 번

2 무슨 수를 쓰더라도 코드 중복은 피한다.

3 같은 일을 반복하지 않는다.

코드를 재사용할 수 있도록 설계하는 방법

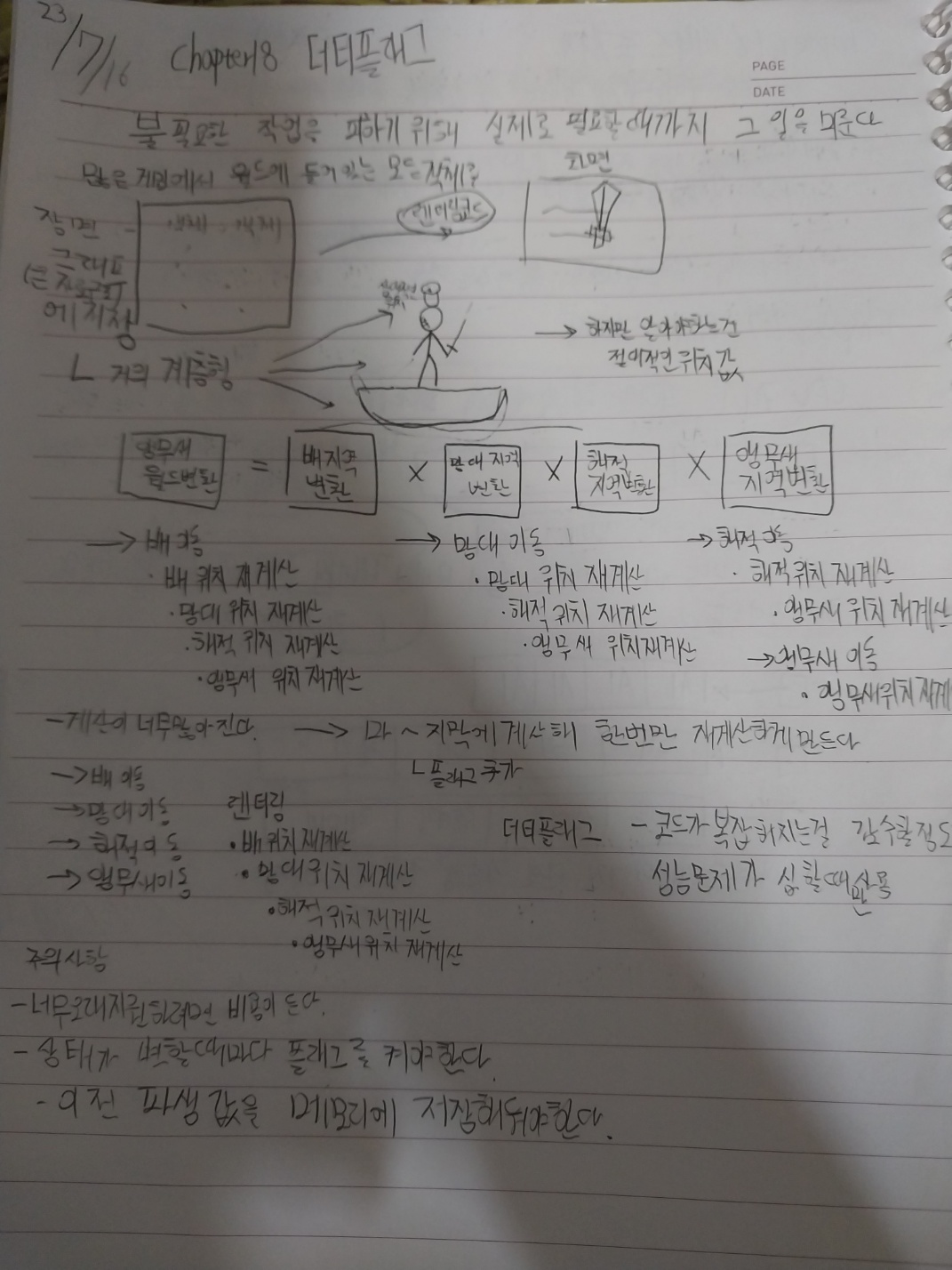
1 용도나 분야가 약간 달라도 충분히 사용할 수 있도록 범용성을 갖춰야 한다.

2 재사용 가능한 코드는 사용하기 쉽게 만들어야 한다.

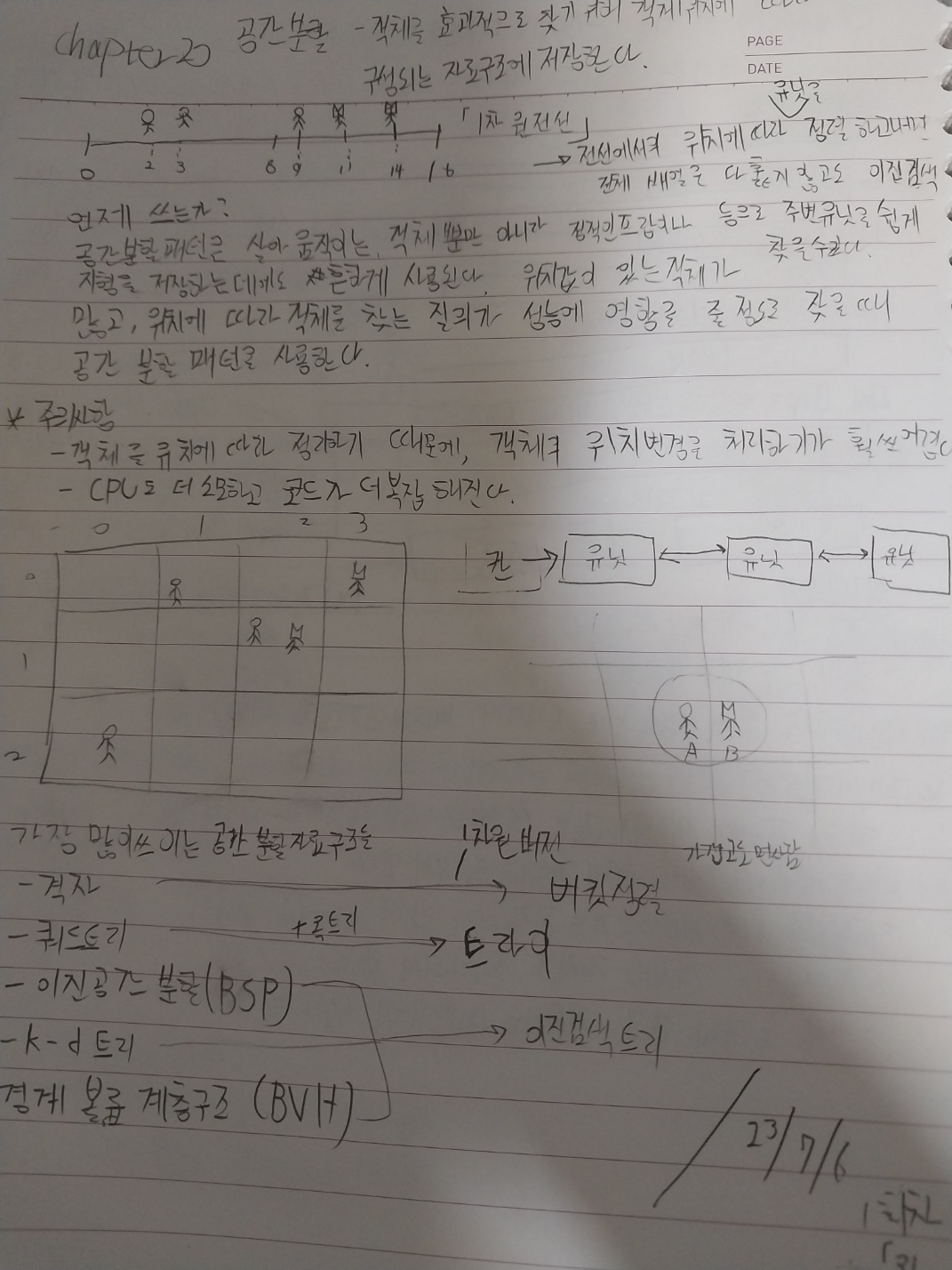
재사용할 수 있는 코드를 설계하는 데 가장 중요한 부분은 추상화다.

추상화의 핵심 - 인터페이스와 구현을 실질적으로 분리하는데 있다.

코드작성자가 인터페이스를 변경하지 않고도 내부 구현 코드를 쉽게 수정할 수 있게 해준다.

더티플래그 – 불필요한 작업을 피하기 위해 실제로 필요할 때까지 그 일을 미룬다.

공간분할 – 객체를 효과적으로 찾기 위해 객체위치에 따라 구성되는 자료구조에 저장한다.



가장 많이 쓰이는 공간 분할 자료구조들

- 격자

- 쿼드트리

- 이진공간분할(BSP)

- k-d트리

- 경계 볼륨 게층구조(BVH)

|  |  |  |  |
| --- | --- | --- | --- |
| **문제점 정리** | . | | |
| **해결방안** |  | | |
| **다음주차** | 4주차 | **다음기간** | 2023.7.17 ~ 2023.7.23 |
| **다음주 할일** | 1. 전문가를 위한 C++책(C++ 공부1)  2. 언리얼 개발문서 정독,이해,분석  3. 언리얼 소스코드 분석(gameplay framework) | | |
| **지도 교수**  **Comment** |  | | |