

[사용자] <T> 제네릭의 표현과 Partial (with zod)

≡ 구현 어려움 정도	개념
☑ 구현 완료	☑
📅 기간	@December 9, 2024
☑ 정리 완료	☑

1. <T> 란?

<T> 는 제네릭(Generic)의 표현으로, 함수나 클래스, 인터페이스가 특정 타입에 의존하지만 그 타입을 고정하지 않고 다양한 타입을 지원할 수 있도록 만듭니다.

제네릭의 의미

- T 는 임의의 타입을 나타냅니다. (관습적으로 T 를 사용하지만 T 대신 다른 이름도 가능)
- 함수나 컴포넌트가 호출될 때 T 가 구체적인 타입으로 대체됩니다.

```
function identity<T>(value: T): T {  
  return value;  
}  
  
// 호출 시 T가 number로 대체됨  
identity<number>(10); // 반환값: 10  
  
// 호출 시 T가 string으로 대체됨  
identity<string>("hello"); // 반환값: "hello"
```

2. Partial<T> 란?

Partial<T> 는 유틸리티 타입으로, T 의 모든 속성을 선택적(optional)으로 만듭니다. 즉, 객체의 모든 프로퍼티가 필수적(required)인 경우에도 선택적(optional)로 변경할 수 있습니다.

Partial<T> 의 특징

- T 객체의 모든 속성을 ? 로 바꿉니다.
- 주로 객체 업데이트나 선택적 값 설정이 필요한 경우에 사용됩니다.

```
type User = {  
  name: string;  
  age: number;  
};  
  
const partialUser: Partial<User> = {  
  name: "Alice", // age는 없어도 오류가 발생하지 않음  
};
```

zod를 사용에 위의 제네릭과 유틸리티 타입 Partial을 사용한 이유

유효성 검사 함수는 다양한 데이터 구조(스키마)를 처리해야 하기 때문에 타입을 동적으로 받아와야 함.

검증 중 일부 필드만 오류가 있는 경우가 있으므로 모든 속성을 선택적으로 받을 필요가 있음.

참고

- <https://www.typescriptlang.org/ko/docs/handbook/2/generics.html>
- [https://inpa.tistory.com/entry/Ts-!\[\]\(694fcb4611893e9db5249daba48abfc1_img.jpg\)-타입스크립트-유틸리티-타입-~~100~~-총정리](https://inpa.tistory.com/entry/Ts-%EC9D%BC-타입스크립트-유틸리티-타입-%ED8C%B0-총정리)