

**아이템34 부정확한 타입보다 미완성 타입 사용하기**

## 아이템34 부정확한 타입보다 미완성 타입 사용하기

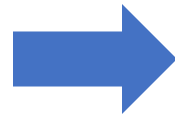
\* GeoJSON : json으로 위치 데이터와 속성 데이터를 저장하는 형식.

```
interface Point{
  type: 'Point';
  coordinates: number[];
}

interface LineString{
  type: 'LineString';
  coordinates: number[][];
}

interface Polygon{
  type: 'Polygon';
  coordinates: number[][][];
}

type Geometry = Point | LineString | Polygon;
```



```
type GeoPosition = [number, number];
interface Point {
  type: 'Point';
  coordinates: GeoPosition;
}
```

'튜플타입'

## 아이템34 부정확한 타입보다 미완성 타입 사용하기

\* Mapbox 맵박스 라이브러리 : 벡터 지도용 javascript 라이브러리

```
type Expression = any;
type Expression2 = number | string | any[];

const tests: Expression2[] = [
  10,
  "red",
  true,
  ["+", 10, 5],
  ["case", [ ">", 20, 10 ], "red", "blue", "green"], // 값이 너무 많습니다.
  ["**", 2, 31], // "**" 는 함수가 아니므로 오류가 발생해야 합니다.
  ["rgb", 255, 128, 64],
  ["rgb", 255, 0, 127, 0] // 값이 너무 많습니다.
];
```

## 아이템34 부정확한 타입보다 미완성 타입 사용하기

```
type FnName = '+' | '-' | '*' | '/' | '>' | '<' | 'case' | 'rgb';
type CallExpression = [FnName, ...any[]];
type Expression3 = number | string | CallExpression;

const tests: Expression3[] = [
  10,
  "red",
  true,
  ["+", 10, 5],
  ["case", [ ">", 20, 10 ], "red", "blue", "green"],
  ["**", 2, 31], // "**" 는 FnName 형식에 할당할 수 없습니다.
  ["rgb", 255, 128, 64],
  ["rgb", 255, 0, 127, 0]
];
```

## 아이템34 부정확한 타입보다 미완성 타입 사용하기

\* 호출 표현식= 자바스크립트에서 함수나 메서드를 호출하는 문법

```
type Expression4 = number | string | CallExpression;  
type CallExpression = MathCall | CaseCall | RGBCall;
```

```
interface MathCall{  
  0: '+' | '-' | '/' | '>' | '<';  
  1: Expression4;  
  2: Expression4;  
  length: 3;  
}
```

```
interface CaseCall {  
  0: 'case';  
  1: Expression4;  
  2: Expression4;  
  3: Expression4;  
  length: 4 | 6 | 8 | 10 | 12 | 14 | 16  
}
```

```
interface RGBCall{  
  0: 'rgb';  
  1: Expression4;  
  2: Expression4;  
  3: Expression4;  
  length: 4;  
}
```

```
const tests: Expression4[] = [  
  10,  
  "red",  
  true,  
  ["+", 10, 5],  
  ["case", [">", 20, 10], "red", "blue", "green"],  
  ["**", 2, 31],  
  ["rgb", 255, 128, 64],  
  ["rgb", 255, 128, 64, 73]  
];
```

// ~ ~ ~ 'number' 형식은 'string' 형식에 할당할 수 없습니다.

**아이템35 데이터가 아닌, API와 명세를 보고 타입 만들기**

## 타입 명세

**GeoJSON 명세 사용시 GeometryCollection 명시적으로 차단하기**

**GeometryCollection 차단 < 헬퍼함수 호출로 모든 타입 지원하기**

\*GeoJSON :json형태로 지형 데이터를 정의하는 포맷

\*GeometryCollection : geosion의 타입, 다양한 도형 모음

아이템35 데이터가 아닌, API와 명세를 보고 타입 만들기

## API

**GraphQL**  
String (null 가능)

**TypeScript**  
string (null 불가능)

\***GraphQL** : API를 위한 쿼리 언어이며 이미 존재하는 데이터로 쿼리를 수행하기 위한 런타임.



## 아이템35 데이터가 아닌, API와 명세를 보고 타입 만들기

**Apollo** :GraphQL 쿼리를 타입스크립트 타입으로 변환해주는 도구.

[api.github.com/graphql](https://api.github.com/graphql)

- 쿼리 매개변수, 응답 -> 인터페이스 생성
- Null 여부 -> 응답 인터페이스로 변환
- 주석 -> JSDoc로 변환

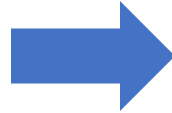
\***JSDoc**: Javascript 소스코드 파일에 주석을 달기위해 사용되는 마크업언어

**아이템36 해당 분야의 용어로 타입 이름 짓기**

## 아이템36 해당 분야의 용어로 타입 이름 짓기

**name**

**endangered**  
**habitat**



**commonName**

**genus**  
**species**  
**status**  
**climates**

## 아이템36 해당 분야의 용어로 타입 이름 짓기

- 동일한 의미 -> 같은 용어
- 모호하고 의미없는 이름 피하기
- 데이터 자체 고려하기