

Types vs Interface

Aspect	Type	Interface
Can describe functions	✓	✓
Can describe constructors	✓	✓
Can describe tuples	✓	✓
Interfaces can extend it	⚠	✓
Classes can extend it	⊘	✓
Classes can implement it (<code>implements</code>)	⚠	✓
Can intersect another one of its kind	✓	⚠
Can create a union with another one of its kind	✓	⊘
Can be used to create mapped types	✓	⊘
Can be mapped over with mapped types	✓	✓
Expands in error messages and logs	✓	⊘
Can be augmented	⊘	✓
Can be recursive	⚠	✓

⚠ In some cases

На відміну від типів, інтерфейс може описувати тільки об'єктний тип, тобто такі типи інтерфейсом не опишеш

1. Для type aliases недоступне декларативне злиття, так не можна робити

```
type TAccount = {  
  name: string;  
};  
  
type TAccount = {  
  age: number;  
};  
  
/* А з інтерфейсами все ОК */  
interface IAccount {  
  name: string;  
}  
  
interface IAccount {  
  age: number;  
}
```

2. Інтерфейси можна розширювати

```
interface Person {  
  name: string;  
}  
interface Student extends Person {  
  age: number;  
}  
  
/* А як з type ? */  
type Person1 = {  
  name: string;  
};  
  
type Student1 = Person1 & { age: number };
```