

Optional Chaining (?.)

Optional chaining allows you to safely access properties or call methods on an object that might be null or undefined without causing a runtime error. If any part of the chain before the ?. is null or undefined, the entire expression short-circuits and evaluates to undefined. This prevents errors that would otherwise occur when attempting to access a property of a non-existent object.

```
const user = {  
  profile: {  
    address: {  
      street: "123 Main St"  
    }  
  }  
};  
  
const streetName = user?.profile?.address?.street; // "123 Main St"  
  
const bioMarks=user?.marks?.bio;           // returns undefined before it proceed to bio as marks are  
unavailable
```

Nullish Coalescing (??)

The nullish coalescing operator provides a default value for an expression if the left-hand operand evaluates to null or undefined. Unlike the logical OR operator (||), which treats other falsy values like 0, "" (empty string), or false as a reason to use the default, ?? only considers null and undefined as "nullish" values.

```
const userName = null;  
  
const defaultName = "Guest";  
  
const displayName = userName ?? defaultName; // "Guest"
```

Combining These Both Operators:

These operators are frequently used together to provide a default value when a deeply nested property might be missing or null/undefined.

```
const user = {
```

```
profile: {  
  address: {  
    street: "123 Main St"  
  }  
}  
};  
  
const bioMarks=user?.marks?.bio??50;           //now it will return 50 as the leftside is undefined
```