

TypeScript break

If this site saves you hours of work, please
whitelist it in your ad blocker 🙏 to
support us ❤️
in creating more helpful and free content
in the future.

Summary: in this tutorial, you will learn about the TypeScript break statement to terminate a loop or a `switch`.

Using TypeScript break to terminate a loop

The `break` statement allows you to terminate a loop and pass the program control over the next statement after the loop.

You can use the `break` statement inside the `for`, `while`, and `do...while` statement.

The following example shows how to use the `break` statement inside a `for` loop:

```
let products = [
  { name: 'phone', price: 700 },
  { name: 'tablet', price: 900 },
  { name: 'laptop', price: 1200 }
];

for (var i = 0; i < products.length; i++) {
  if (products[i].price == 900)
    break;
}

// show the products
console.log(products[i]);
```

Output:

```
{ name: 'tablet', price: 900 }
```

How it works:

- First, initialize a list of products with name and price properties.
- Then, search for the product whose price is `900` and terminate the loop once the product is found by using the `break` statement.
- Finally, show the matching product to the console.

Using the break statement to break a switch

The following example returns the discount of a specified product. It uses the `break` statement to break out of a `switch` :

```
let products = [
  { name: 'phone', price: 700 },
  { name: 'tablet', price: 900 },
  { name: 'laptop', price: 1200 }
];

let discount = 0;
let product = products[1];

switch (product.name) {
  case 'phone':
    discount = 5;
    break;
  case 'tablet':
    discount = 10;
    break;
  case 'laptop':
    discount = 15;
    break;
}

console.log(`There is a ${discount}% on ${product.name}.`);
```

Note that besides a loop or a switch, the break statement can be used to break out of a labeled statement. However, it is rarely used in practice so that we don't cover in this tutorial.

Summary

- Use `break` statement to terminate a loop or switch.