Array Magic

Typescript

Array In Typescript

If you've worked with JavaScript arrays before, TypeScript arrays are like their friendly cousins.

They still do the same job, but they've got an added type safety feature.

Imagine you're crafting a to-do app.

You can create an array for your tasks, just like you always do:

```
let tasks: string[] = ["Buy groceries", "Go for a run"];
```

Here's the deal with string[] – it means, "Hey, this array is all about strings!"

It's like having a guard at the door, making sure only the right folks get into your party.

1. Can one array handle both numbers and strings?

Absolutely! You can create an array that holds both numbers and strings using a trick called a "union type":

```
let mixedValues: (string | number)[] = ["apple", 42, 123, "orange"];
```

2. Can I create an array with custom objects of a specific type?

You bet! You can create arrays of user-defined types. For example:

```typescript
type Person = {
    name: string;
    age: number;
}

let people: Person[] = [
    { name: "Alice", age: 28 },
    { name: "Bob", age: 35 }
];
```

3. **What if I want an array with a fixed number of elements and different types for each element?**

That's where tuples come in! Tuples allow you to define precise sequences of types for a fixed number of elements.

For example:

```
let userDetails: [string, number, boolean] = ["Alice", 28, true];
```

Did You Find it
Useful?

Alamin CodePapa
@CodePapa360

Follow for more

Like | Comment | Repost