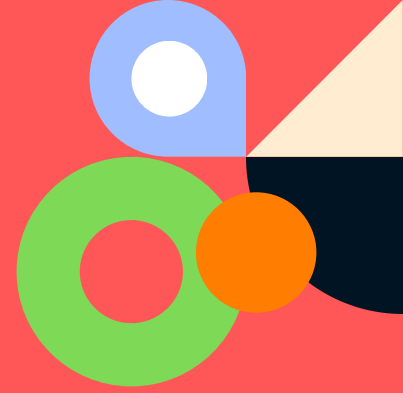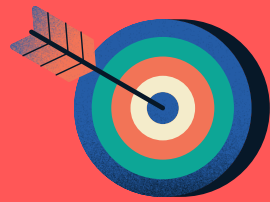# Currying
in
javascript

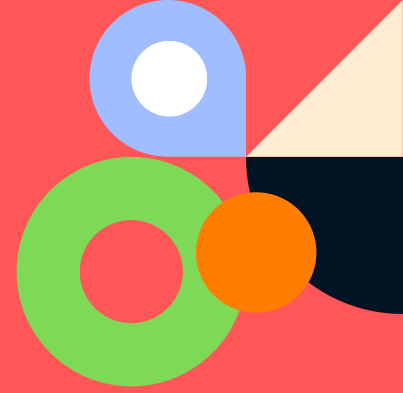{JavaScript}

Currying In Javascript

A functional programming technique where a multi-argument function is transformed into a series of single-argument functions, facilitating partial application and composability.

{ JavaScript }

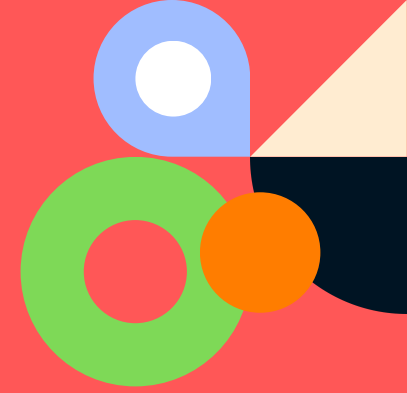# Key Points

➡️ **Enables partial application of arguments.**

➡️ **Enhances composability through function chaining.**

➡️ **Utilizes function closures and higher-order functions for implementation.**
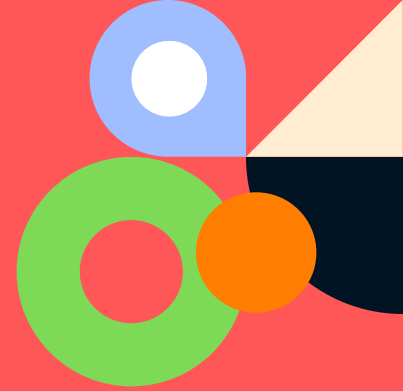
{ JavaScript }

Original function with multiple arguments

```javascript
function filterByProperty(array, property,
value){
    return array.filter(item =>item[property]
=== value);
}
```
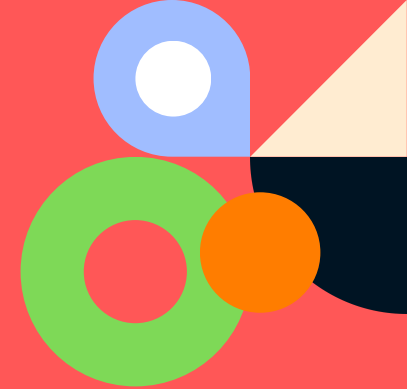
{ JavaScript }

Curried version

```javascript
function curriedFilterByProperty(property) {
  return function(value) {
    return function(array) {
      return
array.filter(item => item[property] === value);
    };
  };
}
```

JavaScript

Let's see how it works .

JavaScript

👉 Take a sample array of user objects

```
const users = [
{ name: 'Alice',
  age: 30,
  gender: 'female',
  nationality: 'US'
},
{ name: 'Bob',
  age: 25,
  gender: 'male',
  nationality: 'UK'
},
{ name: 'Charlie',
  age: 35,
  gender: 'male',
  nationality: 'US'
},
{ name: 'Diana',
  age: 40,
  gender: 'female',
  nationality: 'UK'
}];
```

```javascript
const filterByGender =
 curriedFilterByProperty('gender');
const filterByNationality =
 curriedFilterByProperty('nationality');

const femaleUsers = filterByGender('female')(users);
console.log('Female Users:', femaleUsers);

const UKUsers = filterByNationality('UK')(users);
console.log('UK Users:', UKUsers);
```

{ JavaScript }

Did you find it
Useful?

Leave a comment!

Alamin CodePapa

@CodePapa360

FOLLOW FOR MORE

Like

Comment

Repost