

Functions

Explanation:

Functions are reusable blocks of code. In modern JavaScript, arrow functions are shorter and avoid issues with `this`. Anonymous functions are often used as callbacks, and default parameters prevent errors when no value is provided.

Examples:

```
const add = (a, b) => a + b;
const greet = function(name) { return `Hello ${name}`; };
function greetUser(name = 'Guest') { return `Welcome, ${name}`; }
```

Quick Summary:

Functions help React components stay clean. Arrow functions shorten syntax and default parameters give safety.

	:	
Function	code	
	Arrow function	<code>this</code>
	Anonymous function	callback
	Default parameter	value

Practice Exercises:

- Write an arrow function that squares a number.
- Create a function with a default parameter that greets a user.
- Write an anonymous function assigned to a variable that returns the cube of a number.

Conditionals

Explanation:

Conditionals allow you to run code only if a condition is true. In JavaScript, ternary operators and logical operators (&&, ||) are often used for concise expressions.

Examples:

```
let age = 20; let status = age >= 18 ? 'Adult' : 'Minor';
isLoggedIn && console.log('Welcome back!');
const username = user || 'Guest';
```

Quick Summary:

Conditionals make decisions in code using if, ternary, and logical operators.

:			
Conditionals		code	run
Ternary operator ('?:')	logical operator (&&,)		code

Practice Exercises:

- Use a ternary operator to check if a number is even or odd.
- Use `&&` to log a message only if a variable is true.
- Use `||` to set a default value for a variable.