**this keyword in JavaScript:**

* this refers to the **object that is executing the current function**. (this అనేది ప్రస్తుత function ను execute చేస్తున్న **object** ను సూచిస్తుంది.)
* But the meaning of this depends on **how** and **where** the function is called.

**Different Contexts of this:**

1. Global Context (outside any function or object):

****

* In the **browser**, this refers to the window object.

1. **Inside a regular function:**
   * Again, in **non-strict mode**, this will be the window.
   * In **strict mode**, this will be undefined.

****

1. Inside an object method:



* + Here, this refers to the **object** (person), so this.name is "Indraja".

1. **Arrow functions**



* In arrow functions, this is **not bound** to the object.
* It takes this from the **surrounding (parent) scope**.
* So, in this case, this.name is probably undefined!

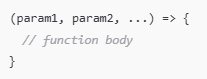
1. **In event handlers (DOM):**



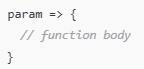
* + this refers to the element that received the event (<button>).

****

**Arrow function syntax:**

****

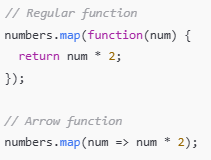
1. If it's just **one parameter**, you can skip the parentheses:

****

1. If the function has **only one return statement**, you can make it even shorter:

** or **

1. Regular function vs Arrow functions:

****

Arrow functions are shorter and cleaner, but they **don’t have their own *this*** — so be careful when using them in object methods or class components.