

in genral we can't write  
array like this `const arr = [`  
    `'Ahmed',`  
    `index: 14,`  
    `input: 'Hello, Hello,`  
    `Ahmed',`  
    `groups: undefined`  
]; so how this exist [ [  
    `'Ahmed', index: 14, input:`  
    `'Hello, Hello, Ahmed',`  
    `groups: undefined ] ]`

Right — you’ve spotted something interesting here.

In **literal syntax**, you’re correct:

Js > Copy

```
// ❌ SyntaxError
const arr = [
  'Ahmed',
  index: 14, // illegal here
  input: 'Hello, Hello, Ahmed',
  groups: undefined
];
```

That fails because inside an array literal (`[ ... ]`), every comma-separated slot must be:

- a value (like `'Ahmed'` or `42`)
  - or an expression
- You can’t write `key: value` — that’s **object literal** syntax.

## So how can something like

Js > Copy

```
[ 'Ahmed', index: 14, input: 'Hello...', groups: undefined ]
```

show up in the console?

It's **not actual JavaScript syntax** — it's **DevTools' display format**.

When you log a `RegExpMatchArray` (from `.matchAll()` or `.exec()`), the engine hands DevTools an object that is:

- **An array** ( `['Ahmed']` at index `0` )
- **With extra properties** ( `index`, `input`, `groups` )