



SVELTE • LEGACY APIS

Reactive `let`/`var` declarations

In runes mode, reactive state is explicitly declared with the `$state` runes.

In legacy mode, variables declared at the top level of a component are automatically considered *reactive*. Reassigning or mutating these variables (`count += 1` or `object.x = y`) will cause the UI to update.

```
<script>
  let count = 0;
</script>

<button on:click={() => count += 1}>
  clicks: {count}
</button>
```



Because Svelte's legacy mode reactivity is based on *assignments*, using array methods like `.push()` and `.splice()` won't automatically trigger updates. A subsequent assignment is required to 'tell' the compiler to update the UI:

```
<script>
  let numbers = [1, 2, 3, 4];

  function addNumber() {
    // this method call does not trigger an update
    numbers.push(numbers.length + 1);

    // this assignment will update anything
    // that depends on `numbers`
    numbers = numbers;
  }
</script>
```



