EXAMPLE 7: LOCAL VARIABLES "HIDE" OR "SHADOW" GLOBALS WITH THE SAME NAME

EXAMPLE 7: LOCAL VARIABLES "HIDE" OR "SHADOW" GLOBALS WITH THE SAME NAME

```
<script>
window.onload = printX;
var x = 5;
function printX() {
  var x = 10;
  var pi = 3.1415;
  e = 2.71828;
  console.log("Inside printX: x = " + x);
  printAnotherX();
}

function printAnotherX() {
  var x = 20;
  console.log("Inside printAnotherX: x = " + x);
}

</script>
```

```
Inside printX: x = 10
Inside printAnotherX: x = 20
```

EXAMPLE 7: LOCAL VARIABLES "HIDE" OR "SHADOW" GLOBALS WITH THE SAME NAME

```
<script>
window.onload = function();
var x = 5;
function printX() {
  var x = 10;
  var pi = 3.1415;
  e = 2.71828;
  console.log("Inside printX: x = " + x);
  printAnotherX();
}

function printAnotherX() {
  var x = 20;
  console.log("Inside printAnotherX: x = " + x);
}

</script>
```

HERE IS A VARIABLE FLOATING ABOUT OUTSIDE ANY FUNCTION - ITS A GLOBAL VARIABLE

IT WILL EXIST FOR AS LONG AS THE PAGE IS LOADED IN THE BROWSER

EXAMPLE 7: LOCAL VARIABLES "HIDE" OR "SHADOW" GLOBALS WITH THE SAME NAME

```
<script>
window.onload = printX;
var x = 5;
function printX() {
  var x = 10;
  var pi = 3.1415;
  e = 2.71828;
  console.log("Inside printX: x = " + x);
  printAnotherX();
}

function printAnotherX() {
  var x = 20;
  console.log("Inside printAnotherX: x = " + x);
}

</script>
```

HERE INSIDE THE FUNCTION `printX` IS ANOTHER VARIABLE OF THE SAME NAME - THIS

EXAMPLE 7: LOCAL VARIABLES "HIDE" OR "SHADOW" GLOBALS WITH THE SAME NAME

THE FUNCTION `printX` CALLS ANOTHER FUNCTION CALLED `printAnotherX`

```
<script>
window.onload = printX;
var x = 5;
function printX() {
  var x = 10;
  var pi = 3.1415;
  e = 2.71828;
  console.log("Inside printX: x = " + x);
  printAnotherX();
}

function printAnotherX() {
  var x = 20;
  console.log("Inside printAnotherX: x = " + x);
}

</script>
```

EXAMPLE 7: LOCAL VARIABLES "HIDE" OR "SHADOW" GLOBALS WITH THE SAME NAME

```
<script>
window.onload = printX;
var x = 5;
function printX() {
  var x = 10;
  var pi = 3.1415;
  e = 2.71828;
  console.log("Inside printX: x = " + x);
  printAnotherX();
}

function printAnotherX() {
  var x = 20;
  console.log("Inside printAnotherX: x = " + x);
}

</script>
```

THE FUNCTION `printX` CALLS ANOTHER FUNCTION CALLED `printAnotherX`

HERE INSIDE THE FUNCTION `printAnotherX` IS YET ANOTHER VARIABLE OF THE SAME NAME

EXAMPLE 7: LOCAL VARIABLES "HIDE" OR "SHADOW" GLOBALS WITH THE SAME NAME

```
<script>
window.onload = printX;
var x = 5;
function printX() {
  var x = 10;
  var pi = 3.1415;
  e = 2.71828;
  console.log("Inside printX: x = " + x);
  printAnotherX();
}

function printAnotherX() {
  var x = 20;
  console.log("Inside printAnotherX: x = " + x);
}

</script>
```

```
Inside printX: x = 10
Inside printAnotherX: x = 20
```

WHEN A LOCAL AND A GLOBAL VARIABLE BOTH EXIST IN A SCOPE, THE LOCAL VERSION TAKES

EXAMPLE 7: LOCAL VARIABLES "HIDE" OR "SHADOW" GLOBALS WITH THE SAME NAME

```
Inside printX: x = 10
Inside printAnotherX: x = 20
```

WHEN A LOCAL AND A GLOBAL VARIABLE BOTH EXIST IN A SCOPE, THE LOCAL VERSION TAKES

```
<script>
window.onload = printX;
var x = 5;
function printX() {
  var x = 10;
  var pi = 3.1415;
  e = 2.71828;
  console.log("Inside printX: x = " + x);
  printAnotherX();
}

function printAnotherX() {
  var x = 20;
  console.log("Inside printAnotherX: x = " + x);
}

</script>
```