



P-1 P2

```
var teacher = "Sanket Singh";
function fun() {
  var teacher = "Sarathak";
  console.log("hello ", teacher);
}

function gun() {
  var student = "Aman";
  console.log(student);
}

fun();
gun();
```

Sanket Singh

teacher → global

Sarathak

teacher - fun

Aman
student
gun

active scope → ~~global~~ fun ~~global~~ gun

phase 1

→ Scope resolution ??

phase 2

→ It will go line by line

→ if it sees variable as LHS, i.e. if it sees a

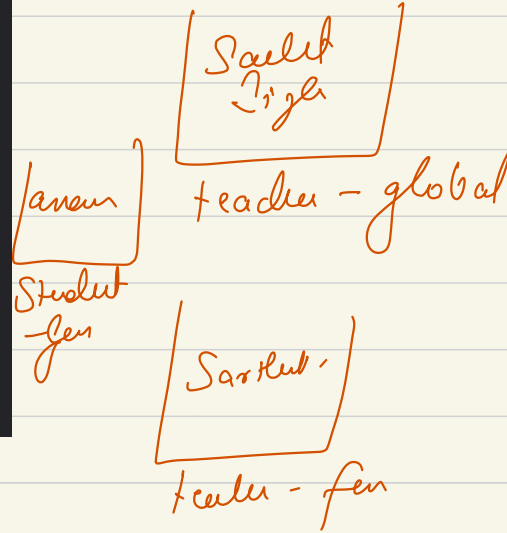
formal declaration of a variable, then it asks a
scope manager for the scope of the variable.

var
let
const

if the variable is found in the curr scope then
it is good, else we go one level up.

p-1 p-2

```
→ var teacher = "Sanket Singh"; global
→ function fun() { global
  → var teacher = "Sarthak";
  → console.log("hello ", teacher);
}
→ function gun() { global
  → var student = "Aman"; gun
  → console.log(student, teacher);
}
→ fun();
→ gun();
```



access → global fun

```

var teacher = "Sanket Singh";
function fun() {
  var teacher = "Sarthak";
  console.log("hello ", teacher);

  function gun() {
    var student = "Aman";
    console.log(student, teacher);
  }
  gun();
}
fun();

```

Handwritten annotations on the code:

- Arrows point to the first `var teacher` and the `fun()` call, both labeled *global*.
- An arrow points to the `var teacher` inside `fun()`, labeled *fun*.
- An arrow points to the `var student` inside `gun()`, labeled *fun*.
- Below the `console.log` in `gun()`, the values `aman, Sarthak` are written, with lines connecting them to the variables `student` and `teacher` respectively.
- The `fun()` call at the bottom is circled.

p-1

Aman
Student
fun

Sanket
Singh
teacher

Sarthak

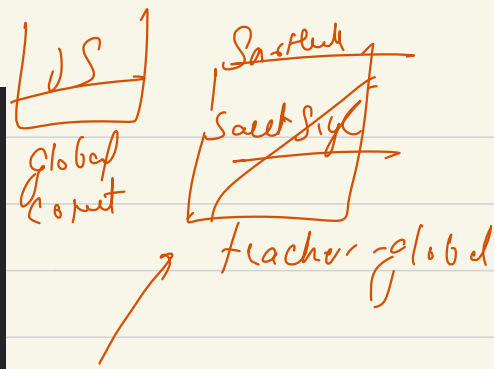
teacher fun

only → global fun gun

p-1 → p-2

```
var teacher = "Sanket Singh";  
function fun() {  
  teacher = "sarthak";  
  content = "JS";  
  console.log("wow", content, teacher);  
}  
fun();  
console.log(teacher);  
console.log(content);
```

global
global
JS
Sarthak

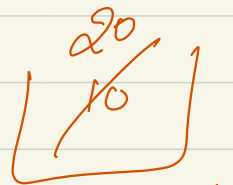


active global function

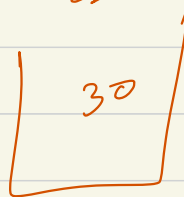
autoglobals

p-1 = p-2

```
var x = 10;  
if(true) {  
  var x = 20;  
  var y = 30;  
  console.log(x, y);  
}  
console.log(x, y);
```



global x



global y

active + global scope

f-1 - f9 -

```
var x = 10;  
console.log(x, y);  
if(true) {  
    var x = 20;  
    var y = 30;  
    console.log(x, y);  
}  
console.log(x, y);
```

Handwritten annotations on the code:

- Under `var x = 10;`: 10
- Under `console.log(x, y);` (first): 10 and undefined
- Under `var x = 20;`: 20 and global
- Under `var y = 30;`: 30 and global
- Under `console.log(x, y);` (second): 20 and 30
- Under `console.log(x, y);` (third): 20 and 30

active global

