

```

1 // ===== Arow # () => expression =====
2
3 /*
4   () => expression
5
6   function fun() {
7
8       return expressino
9   }
10 */
11
12 // create function
13 let myfun = (x) => x * x;
14 //call function
15 console.log(myfun(5)); // 25
16
17 // if we function body then we have to use retrun keyword
18 let body = (x) => {
19     return x * x;
20 };
21 // call function
22 console.log(body(5));
23
24 // ===== this inside the arrow function =====
25 let arrow = () => this;
26
27 function normal(){
28     return this;
29 }
30 // call both of them give
31 console.log(arrow()); // window object
32 console.log(normal()); // window object
33
34 // ***** create object of both function *****
35 let functions = {
36     arrow: arrow,
37     normal: normal,
38     arrowTest: () => this,
39 }
40 // call after assing inside the object
41 console.log(functions.arrow()); // window object
42 console.log(functions.normal()); // object itself {arrow: Ć, normal: Ć}
43 console.log(functions.arrowTest()); // window object

```