

```
//ES5
var multiplyES5 = function(x,y){
    return x*y;
};
console.log(multiplyES5(6,7));
```

```
//ES6
const multiplyES6 = (x,y) => x*y;
console.log(multiplyES6(5,7));
```

```
//ES5
var phraseSplitterES5 = function(phrase){
    return phrase.split(' ');
};
console.log(phraseSplitterES5("ES5 features"));
```

```
//ES6
const phraseSplitterES6 = phrase => phrase.split(' ');
console.log(phraseSplitterES6("ES6 Awesomeness"));
```

```
var squareEs5 = function(a) {
    return a*a;
};
console.log(squareEs5(2));
```

```
let squareEs6 = x => x*x;
console.log(squareEs6(10));
```

```
var addEs5 = function(a,b) {
    return a+b;
};
console.log(addEs5(2,4));
```

```
let addEs6 = (a,b) => a+b;  
console.log(addEs6(3,6));
```

```
setTimeout(function(){  
    console.log("Hello!")  
},1000);
```

```
setTimeout(() => console.log("Hello!"),1000);
```

```
let numbers = [1,2,3,4,5,6,7,8,9];
```

```
//ES5  
let es5OddNumbers = numbers.filter(function(number){  
    return number%2;  
});  
console.log(es5OddNumbers);
```

```
//ES6  
const es6OddNumbers = numbers.filter(number => number%2);  
console.log(es6OddNumbers);
```

```
let arr = [1,2,3,4,5,6,7,8,9];
```

```
const square = arr.map(a => a*a);  
console.log(square);
```

```
const cube = arr.map(a => a*a*a);  
console.log(cube);
```

```
const es6EvenNumbers = arr.filter(number => !(number%2));  
console.log(es6EvenNumbers);
```

```
const isMultipleof3 = arr.filter(number => !(number%3));
```

```
console.log(isMultipleof3);

// `return` is implied if using an expression after an arrow.
let developers = [{name : 'karthik'}, {name : 'Ajay'}];

let es5Output = developers.map(function(developer){
    return developer.name;
});
console.log(es5Output);

const es6Output = developers.map(developer => developer.name);
console.log(es6Output);

var setNameIdsEs5 = function setNameIds(id, name){
    return {
        id : id,
        name : name
    };
};
console.log(setNameIdsEs5(10, 'xyz'));

const setNameIdsEs6 = (id, name) => ({
    id : id,
    name : name
});
console.log(setNameIdsEs6(19, 'ABC'));

const smartPhones = [
    { name: 'iphone', price: 649 },
    { name: 'Galaxy S6', price: 576 },
    { name: 'Galaxy Note 5', price: 489 }
];
```

```
// ES5
```

```
var pricesEs5 =  
smartPhones.map(function(smartPhone) {  
    return smartPhone.price;  
});  
console.log(pricesEs5);
```

```
//ES6
```

```
const pricesEs6 = smartPhones.map(smartphone => smartphone.price);  
console.log(pricesEs6);
```

```
// this keyword
```

```
var x = function(){  
    this.val = 1;  
    setTimeout(function(){  
        this.val++;  
        console.log(this.val);  
    },1000);  
};  
var xx = new x();
```

```
// replacing that with this keyword
```

```
var x = function(){  
    var that = this;  
    that.val = 1;  
    setTimeout(function(){  
        that.val++;  
        console.log(that.val);  
    },1000);  
};  
var xx = new x();
```

```

// using fat arrow
var x = function(){
    this.val = 1;
    setTimeout(() => {
        this.val++;
        console.log(this.val);
    },1000);
};
var xx = new x();

// ES5
var x = function(){
    console.log(arguments[0]);
}
x(1,2,3);

// ES6 using fat arrow
var x = () => { console.log(arguments[0]);};
x(1,2,3); // error : arguments is not defined

// use spread operator
var x = (...n) => {
    console.log(n[0]);
};
x(1,2,3);

```