

# 1. Template Literals

ES6 introduces very simple string templates along with placeholders for the variables. The syntax for using the string template is `${PARAMETER}` and is used inside of the back-ticked string.

Template-Literals.js

```
let name = `My name is ${firstName} ${lastName}`
```



## 2. Destructuring Assignment

Destructuring is one of the most popular features of ES6. The destructuring assignment is an expression that makes it easy to extract values from arrays, or properties from objects, into distinct variables.

```
Destructing-Assignment.js

//Array Destructuring
let fruits = ["Apple", "Banana"];
let [a, b] = fruits; // Array destructuring assignment
console.log(a, b);

//Object Destructuring
let person = {name: "Peter", age: 28};
let {name, age} = person; // Object destructuring assignment
console.log(name, age);
```



# 3. Enhanced Object Literals

ES6 provides enhanced object literals which make it easy to quickly create objects with properties inside the curly braces.

```
Enhanced-Object-Literals.js

function getMobile(manufacturer, model, year) {
  return {
    manufacturer,
    model,
    year
  }
}

getMobile("Samsung", "Galaxy", "2020");
```





# 4. Promises

In ES6, Promises are used for asynchronous execution. We can use promise with the arrow function as demonstrated below.



```
var asyncCall = new Promise((resolve, reject) => {  
  // do something  
  resolve();  
}).then(()=> {  
  console.log('DON!');  
})
```



# 5. Classes

Classes are introduced in ES6 which looks similar to classes in other object-oriented languages, such as C++, Java, PHP, etc. But, they do not work exactly the same way. ES6 classes make it simpler to create objects, implement inheritance by using the "extends" keyword and also reuse the code efficiently.

```
Classes.js

class UserProfile {
  constructor(firstName, lastName) {
    this.firstName = firstName;
    this.lastName = lastName;
  }

  getName() {
    console.log(`The Full-Name is ${this.firstName} ${this.lastName}`);
  }
}

let obj = new UserProfile('John', 'Smith');
obj.getName(); // output: The Full-Name is John Smith
```

