

# ES6: NEXT GEN JS

*Peoria JS and Web Professionals*

July, 2018

## FOR OF LOOP

---

**Note:** This type of loop allows you to loop over anything which is iterable (arrays, strings, maps, sets, for example). You cannot use this type of loop over an object.

**Reference:** <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/for...of>

**Code snippet:** old approaches

```
14 const balls = ['basket', 'soccer', 'base', 'soft'];
15
16 ▼ for (let i = 0; i < balls.length; i++) {
17     console.log(balls[i]);
18 }
19
20 // yes, this approach tends to throw students for a loop
21
22 console.log('\n\nAlternate loop approach \n\n');
23 // alternate old approach
24
25 ▼ balls.forEach((ball) => {
26     console.log(ball);
27 }
28 )
29
30 // of course, you can't use break or continue within forEach loop
31
32 console.log('\n\nFor in loop approach \n\n');
33 // alternate approach - often less than useful
34 ▼ for (ball in balls){
35     console.log(ball);
36 }
```

**Code snippet:** for of loop fundamentals. Note that you can also apply a break or continue statement as needed.

```
38 console.log('\n\nFor of loop - new approach \n\n')
39
40 ▼ for (const ball of balls) {
41     console.log(ball);
42 }
43
44 // can also use break and continue
45
46 console.log('\n\nFor of loop - continue \n\n')
47
48 ▼ for (const ball of balls) {
49 ▼     if (ball === 'base') {
50         continue;
51     }
52     console.log(ball);
53 }
```

Result when viewed in the browser console is shown below.

For of loop - new approach

---

basket

---

soccer

---

base

---

soft

For of loop - continue

---

basket

---

soccer

---

soft

Reference file: **06ForOfLoop.html**