

```

1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <title>for of Loop</title>
9  </head>
10
11 <body>
12
13     <script>
14         /*
15         ,         ***** for of loop *****
16         ,
17         ,         루프의 상태를 알아보자.
18         ,         루프는 기본적으로 같은일을 반복하는것이다.
19         ,
20         ,         예를들어
21         ,         for (let i = 0; i < 20; i ++){
22         ,             console.log("i love kimchi");
23         ,         }
24         ,         기본값을 0으로 할당하고 i가 20보다 작을때 콘솔 구문이 시작되고
25         ,         그 후에 i++ 으로 i가 1이 되었다가 2가 되었다가 점점 수가 올라가고 20보다작을때 멈춘다.
26         ,
27         ,         별로 좋지않다. 만약 엘리먼트에 무언가를 해주고 싶다면
28         ,
29         ,         const friends = ["bibi", "kkamzzu", "chocho", "block"]
30         ,
31         ,         for(let i = 0; i < 20; i++){
32         ,             console.log(`${i} i love kimchi`)
33         ,         }
34         ,         배열에 있는 갯수까지만 잘 출력이 되고 나머지는 undefined가 뜬다.
35         ,         이런 경우를 피하면서 단지 friends 각각을 위해 숫자를 세거나 무언가를 하고싶다면 어떻게 해
36         ,         야할까.
37         ,
38         ,         ***** forEach *****
39         ,
40         ,         forEach는 배열에 있는 각각의 엘리먼트에 대해 특정한 액션을 실행한다.
41         ,         그러니 나는 특정한 액션을 선언하고 각각의 엘리먼트 들에게 실행할 수 있는것이다.
42         ,
43         ,         const friends = ["bibi", "kkamzzu", "chocho", "block"]
44         ,         const addHeart = c => console.log(c);
45         ,         friends.forEach(addHeart)
46         ,
47         ,         // friends.forEach((c,i,a) => console.log(c,i,a)) 로 arrow function으로
48         ,         도 가능.
49         ,
50         ,         forEach는 addHeart를 호출하는데 이것은 currentitem과 함께 호출될것이다.
51         ,         current item이 함수의 첫번째 인자가 될것이고
52         ,
53         ,         * 또한 forEach는 index와도 함께 호출이가능
54         ,         indedx는 두번째 인자가 될것이다
55         ,
56         ,         세번째 인자는 current array 현재의 배열을 준다.
57         ,
58         ,         * forEach는 array에서만 가능.
59         ,
60         ,         ***** for of *****

```

```

60 ,      for (const friend of friends) {
61 ,          console.log(friend)
62 ,      }
63 ,      이런식으로 for of를 써서 출력해줘도 이름이 잘 출력된다.
64 ,      let i = 0 ; ... 어찌구 보다는 나아진것같다.
65 ,      이것은 forEach보다 좋은 특징이 있다.
66 ,
67 ,      * const 혹은 let으로 선언할지를 선택할 수 있다.
68 ,
69 ,      for of는 array에서 뿐만 아니라 *iterable* 한것에서 모든 가능하다.
70 ,      iterable은 루프가 가능하다는 뜻.
71 ,
72 ,      for (const letter of "this is looooooong") {
73 ,          console.log(letter);
74 ,      }
75 ,      string도 for of로 풀어줄 수 있다.
76 ,      NodeList(유사배열), Typedarray, map, set 도 가능
77 ,
78 ,      ** for of의 가장 좋은 점은 루프를 멈출 수 있는것이다
79 ,
80 ,      const manyFriends = [
81 ,          "bibi",
82 ,          "kkamzzu",
83 ,          "kany",
84 ,          "chocho",
85 ,          "block",
86 ,          "tori",
87 ,          "wooram",
88 ,          "jkyung",
89 ,          "ahyeon",
90 ,          "jaewoong"
91 ,      ]
92 ,      이런 배열에서 예를들어 데이터베이스에 이름들을 보내주다가
93 ,      wooram을 찾는순간 멈추고싶다
94 ,      forEach를 사용하면 멈출 수 없다.
95 ,
96 ,      for (manyFriend of manyFriends) {
97 ,          if (manyFriend === "jkyung") {
98 ,              break;
99 ,          } else {
100 ,              console.log(manyFriend);
101 ,          }
102 ,      }
103 ,
104 ,      for of안에서 조건문으로 조건을 걸어서 break로 빼주면 가능하다.
105 ,
106 ,      * 정리
107 ,      for of는 루프를 사요할 수 있게 해주고
108 ,      멈추고 싶을때 멈추게 해준다.
109 ,      데이터베이스에 뭔가를 보내고 싶을때 데이터베이스가 꽉찼다면 보내지않고
110 ,      공간이있다면 보낼수있다. 공간이 없다면 에러를 뱉을테니 멈추게 해줄수있고
111 ,      마지막으로 기억해야할것은 for of는 iterable한 모든것에 사용이 가능.
112 ,      */
113
114 const friends = ["bibi", "kkamzzu", "chocho", "block"]
115
116 // forEach
117
118 const addHeart = (c, i, a) => console.log(c, i, a);
119 friends.forEach(addHeart)
120

```

```
121 // for of
122 for (const friend of friends) {
123     console.log(friend)
124 }
125
126 // for of iterable
127 for (const letter of "this is looooooong") {
128     console.log(letter);
129 }
130
131 // stop loop / for of
132 const manyFriends = [
133     "bibi",
134     "kkamzzu",
135     "kany",
136     "chocho",
137     "block",
138     "tori",
139     "wooram",
140     "jkyung",
141     "ahyeon",
142     "jaewoong"
143 ]
144 for (manyFriend of manyFriends) {
145     if (manyFriend === "jkyung") {
146         break;
147     } else {
148         console.log(manyFriend);
149     }
150 }
151
152
153 </script>
154 </body>
155
156 </html>
```