

Web/Python Programming

웹/파이썬 프로그래밍

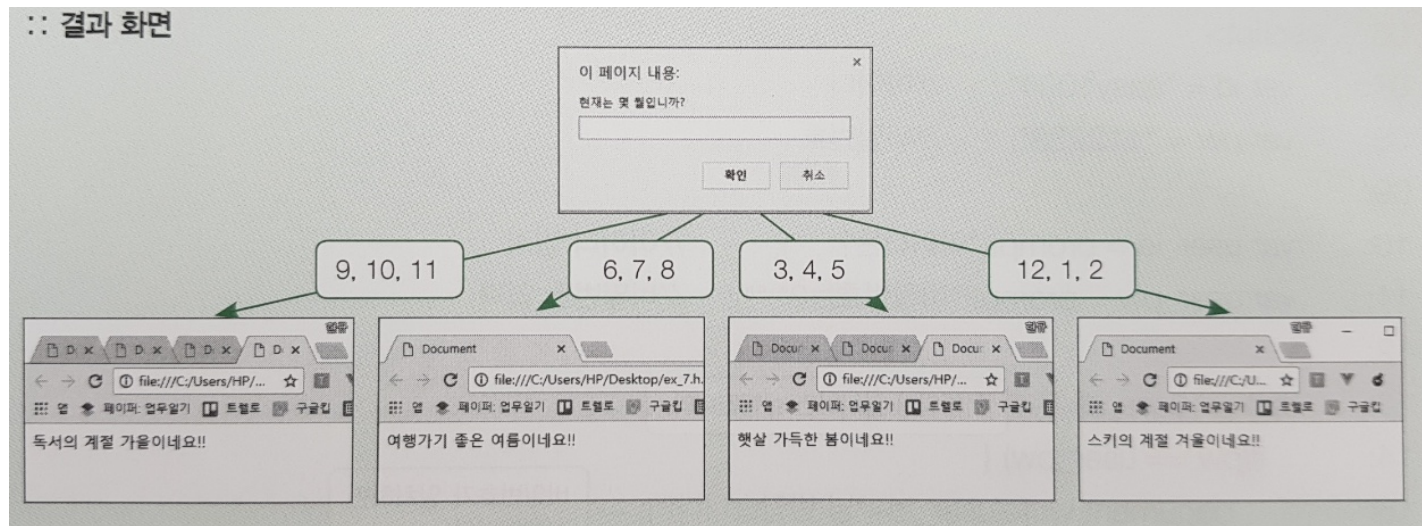
```
21 <?php language_attributes(); ?>
22 <?php bloginfo( 'charset' ); ?>
23 <?php bloginfo( 'width=device-width' ); ?>
24 <?php wp_title( '|', true, 'right' ); ?>
25 <?php rel="profile" href="http://gmpg.org/xfn/11" ?>
26 <?php rel="pingback" href="http://gmpg.org/xfn/11" ?>
27 <?php fruitful_get_favicon(); ?>
28 <?php echo get_template_directory_uri(); ?>
29 <?php wp_head(); ?>
30 </head>
31 <?php body_class(); ?>
32 <div id="page-header" class="hfeed site">
33     $theme_options = fruitful_get_theme_options();
34     $logo_pos = $menu_pos = '';
35     if (isset($theme_options['logo_position']))
36         $logo_pos = esc_attr($theme_options['logo_position']);
37     if (isset($theme_options['menu_position']))
38         $menu_pos = esc_attr($theme_options['menu_position']);
39     $logo_pos_class = fruitful_get_class($logo_pos);
40     $menu_pos_class = fruitful_get_class($menu_pos);
41     $responsive_menu_type = fruitful_get_class($menu_pos);
42     $responsive_menu_type = fruitful_get_class($menu_pos);
43     $responsive_menu_type = fruitful_get_class($menu_pos);
44     $responsive_menu_type = fruitful_get_class($menu_pos);
45     $responsive_menu_type = fruitful_get_class($menu_pos);
46     $responsive_menu_type = fruitful_get_class($menu_pos);
47     $responsive_menu_type = fruitful_get_class($menu_pos);
48     $responsive_menu_type = fruitful_get_class($menu_pos);
49     $responsive_menu_type = fruitful_get_class($menu_pos);
50     $responsive_menu_type = fruitful_get_class($menu_pos);
51     $responsive_menu_type = fruitful_get_class($menu_pos);
52     $responsive_menu_type = fruitful_get_class($menu_pos);
53     $responsive_menu_type = fruitful_get_class($menu_pos);
54     $responsive_menu_type = fruitful_get_class($menu_pos);
55     $responsive_menu_type = fruitful_get_class($menu_pos);
56     $responsive_menu_type = fruitful_get_class($menu_pos);
57     $responsive_menu_type = fruitful_get_class($menu_pos);
58     $responsive_menu_type = fruitful_get_class($menu_pos);
59     $responsive_menu_type = fruitful_get_class($menu_pos);
60     $responsive_menu_type = fruitful_get_class($menu_pos);
61     $responsive_menu_type = fruitful_get_class($menu_pos);
62     $responsive_menu_type = fruitful_get_class($menu_pos);
63     $responsive_menu_type = fruitful_get_class($menu_pos);
64     $responsive_menu_type = fruitful_get_class($menu_pos);
65     $responsive_menu_type = fruitful_get_class($menu_pos);
66     $responsive_menu_type = fruitful_get_class($menu_pos);
67     $responsive_menu_type = fruitful_get_class($menu_pos);
68     $responsive_menu_type = fruitful_get_class($menu_pos);
69     $responsive_menu_type = fruitful_get_class($menu_pos);
70     $responsive_menu_type = fruitful_get_class($menu_pos);
71     $responsive_menu_type = fruitful_get_class($menu_pos);
72     $responsive_menu_type = fruitful_get_class($menu_pos);
73     $responsive_menu_type = fruitful_get_class($menu_pos);
74     $responsive_menu_type = fruitful_get_class($menu_pos);
75     $responsive_menu_type = fruitful_get_class($menu_pos);
76     $responsive_menu_type = fruitful_get_class($menu_pos);
77     $responsive_menu_type = fruitful_get_class($menu_pos);
78     $responsive_menu_type = fruitful_get_class($menu_pos);
79     $responsive_menu_type = fruitful_get_class($menu_pos);
80     $responsive_menu_type = fruitful_get_class($menu_pos);
81     $responsive_menu_type = fruitful_get_class($menu_pos);
82     $responsive_menu_type = fruitful_get_class($menu_pos);
83     $responsive_menu_type = fruitful_get_class($menu_pos);
84     $responsive_menu_type = fruitful_get_class($menu_pos);
85     $responsive_menu_type = fruitful_get_class($menu_pos);
86     $responsive_menu_type = fruitful_get_class($menu_pos);
87     $responsive_menu_type = fruitful_get_class($menu_pos);
88     $responsive_menu_type = fruitful_get_class($menu_pos);
89     $responsive_menu_type = fruitful_get_class($menu_pos);
90     $responsive_menu_type = fruitful_get_class($menu_pos);
91     $responsive_menu_type = fruitful_get_class($menu_pos);
92     $responsive_menu_type = fruitful_get_class($menu_pos);
93     $responsive_menu_type = fruitful_get_class($menu_pos);
94     $responsive_menu_type = fruitful_get_class($menu_pos);
95     $responsive_menu_type = fruitful_get_class($menu_pos);
96     $responsive_menu_type = fruitful_get_class($menu_pos);
97     $responsive_menu_type = fruitful_get_class($menu_pos);
98     $responsive_menu_type = fruitful_get_class($menu_pos);
99     $responsive_menu_type = fruitful_get_class($menu_pos);
100    $responsive_menu_type = fruitful_get_class($menu_pos);
```

Today

- JavaScript
 - Conditional statement (if/ switch)
 - Repetition (for/while/do while)
 - Built-in objects and methods (date, math, array, string)

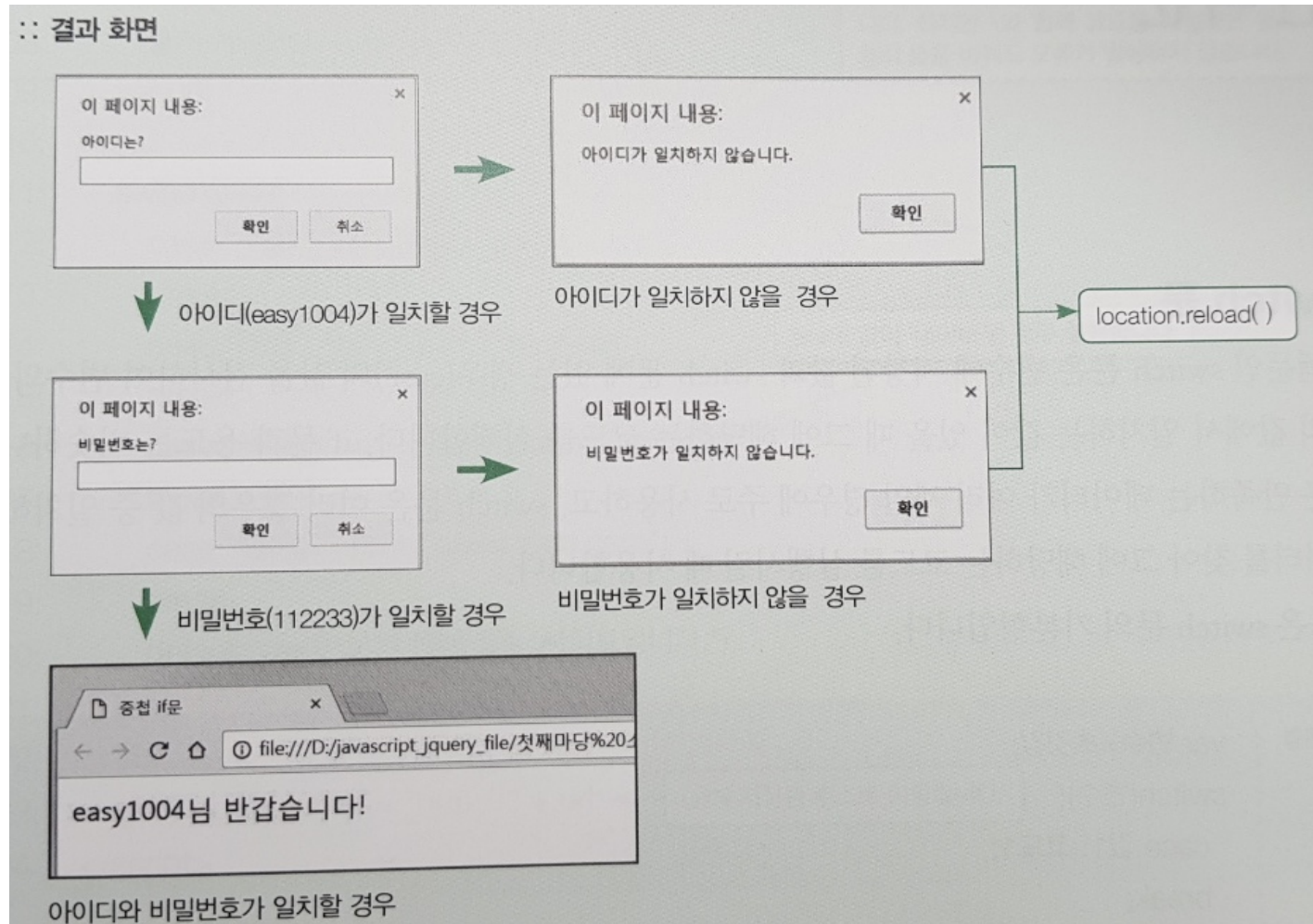
If - else if - else

```
if (condition1){  
    Code1;  
} else if (condition2) {  
    Code2;  
} else if (condition3) {  
    Code3;  
} else {  
    Code4;  
}
```



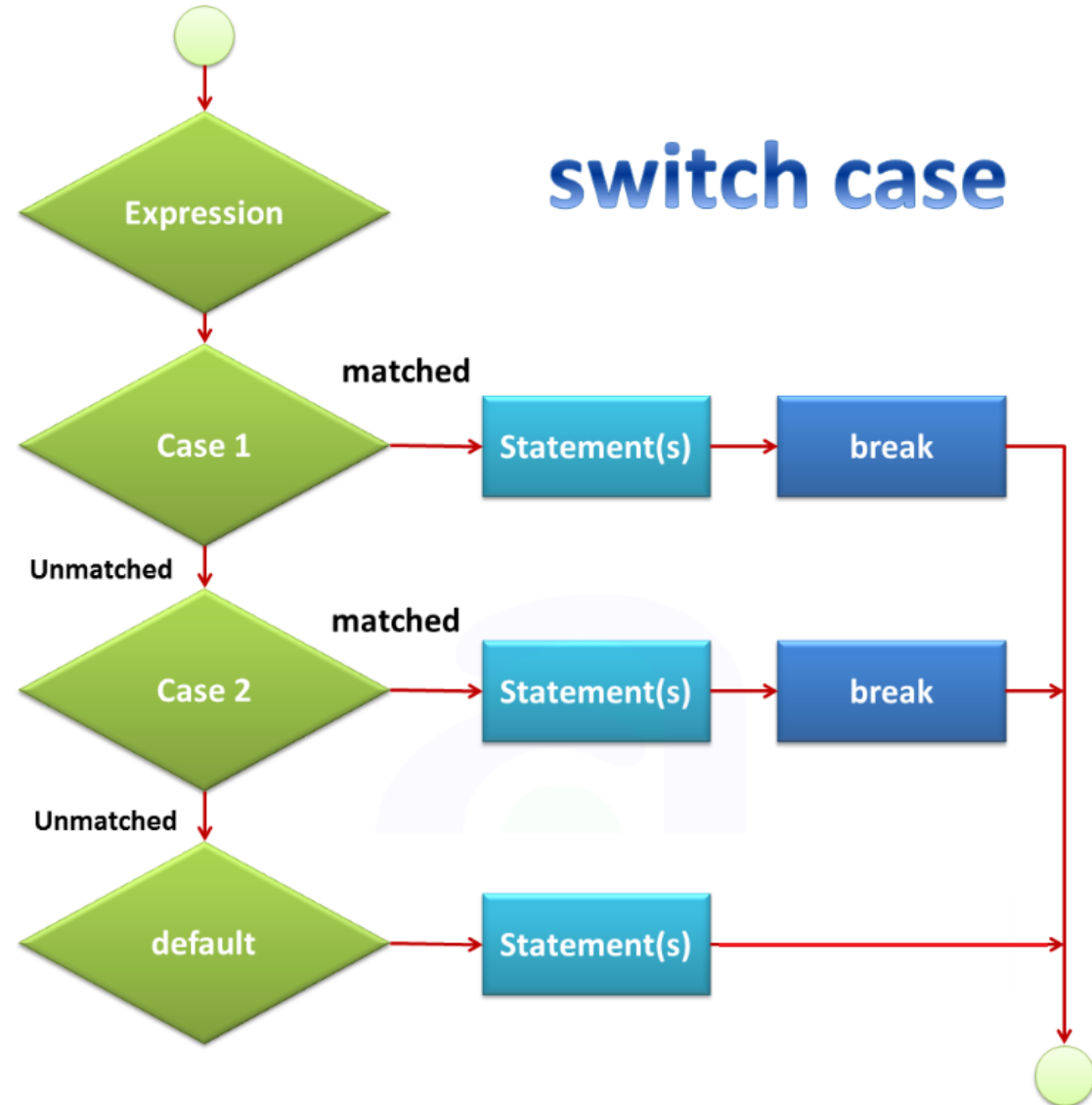
Nested if

```
if (condition1){  
    if (condition2){  
        Code;  
    }  
}
```



switch

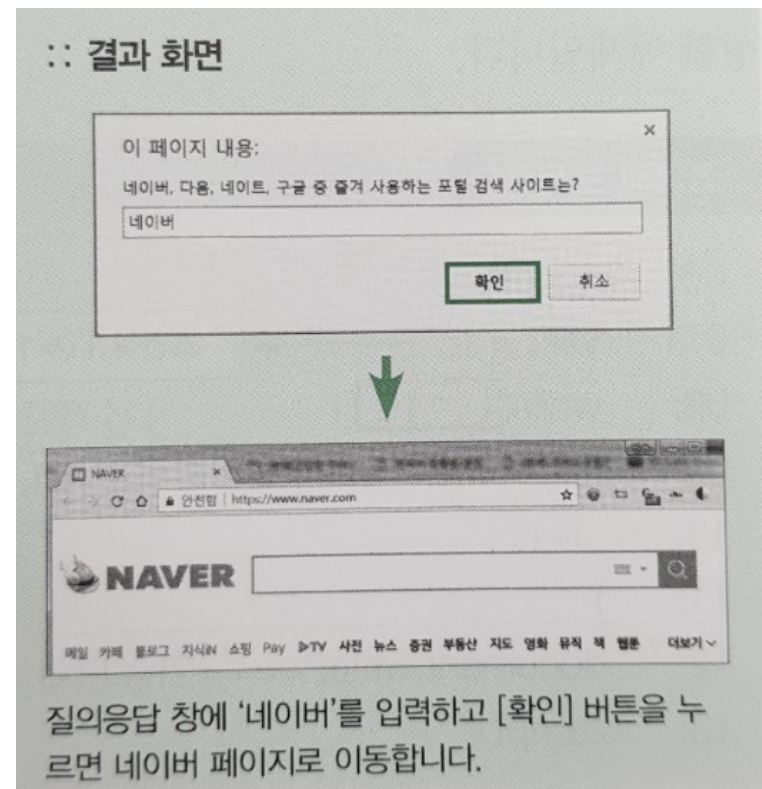
```
var sth = value;  
switch (sth){  
  case value1:  
    Code1;  
    break;  
  case value2:  
    Code2;  
    break;  
  default:  
    Code3;  
}
```



switch example

- Complete your switch_1_test.html

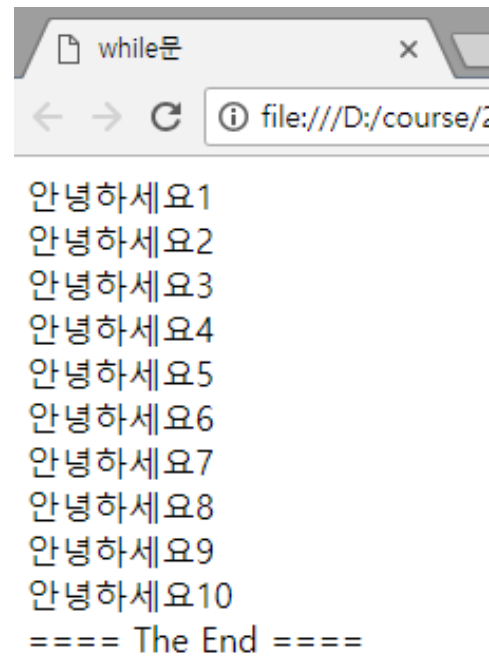
```
6 <script>
7   var site = prompt("네이버, 다음, 네이트, 구글 중 \
8   즐겨 사용하는 포털 검색 사이트는?", "");
9   var url;
10
11   switch(site){
12     case "구글": url = "www.google.com";
13     break;
14     case "다음": url = "www.daum.net";
15     break;
16     case "네이버": url = "www.naver.com";
17     break;
18     case "네이트": url = "www.nate.com";
19     break;
20     default: alert("보기 중에 없는 사이트입니다.");
21   }
22
23   if(url) location.href = "http://" + url;
24 </script>
```



Repetition - while

```
var a = initialized;  
while (condition1){  
    Code1;  
}
```

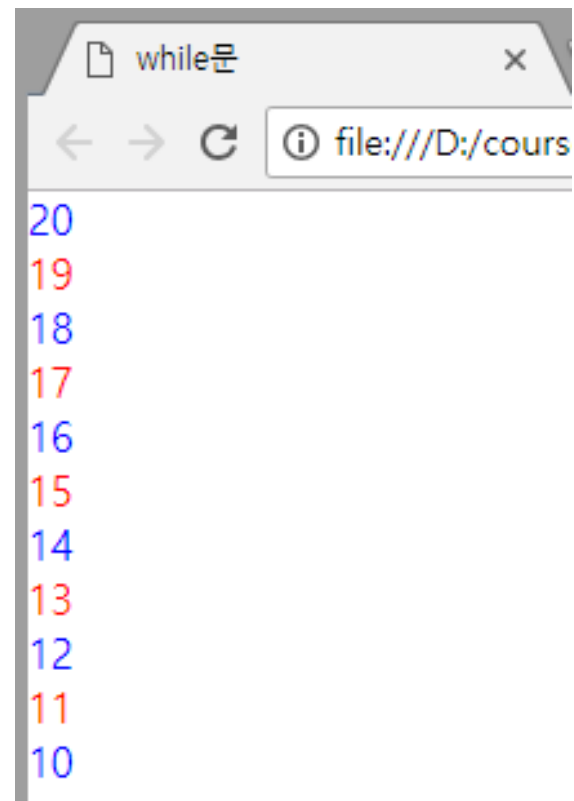
- Complete your while_1_test.html



Repetition – while & if

```
var a = initialized;
while (condition1){
    if (condition2){
        Code1;
    } else {
        Code 2;
    }
    Code3;
}
```

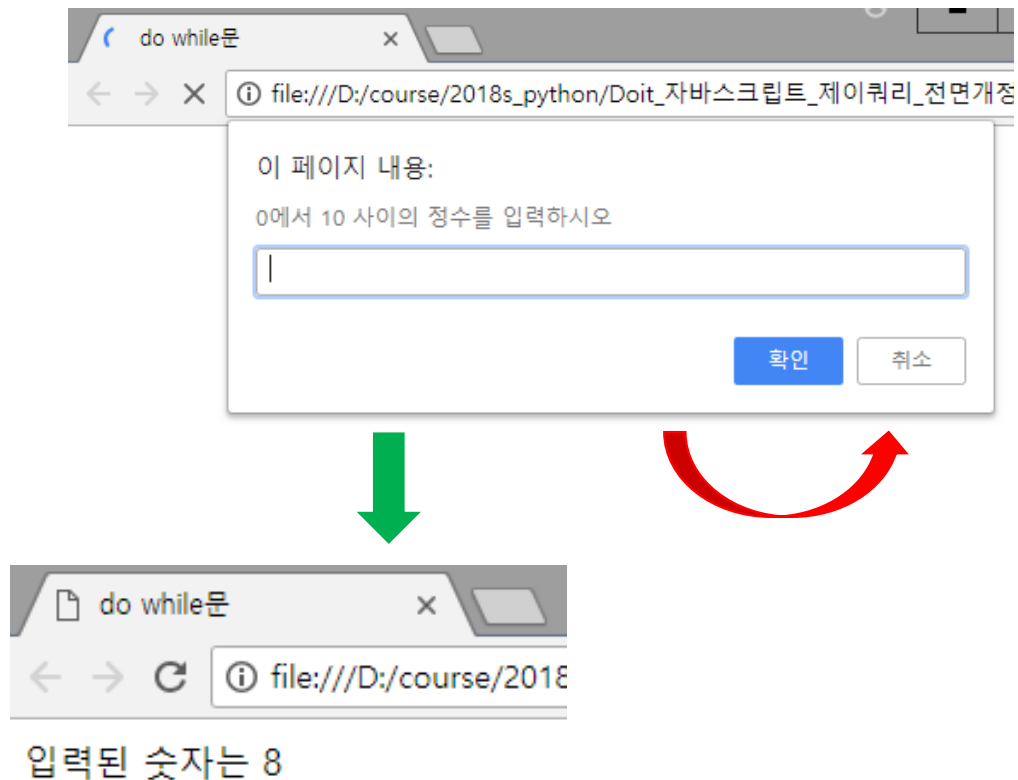
- Complete your while_3_test.html



Repetition – do while

```
do {  
    Code1;  
    Code2;  
} while (condition)
```

- Complete your do_while_2_test.html

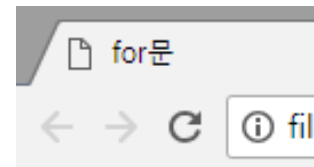


Repetition – for

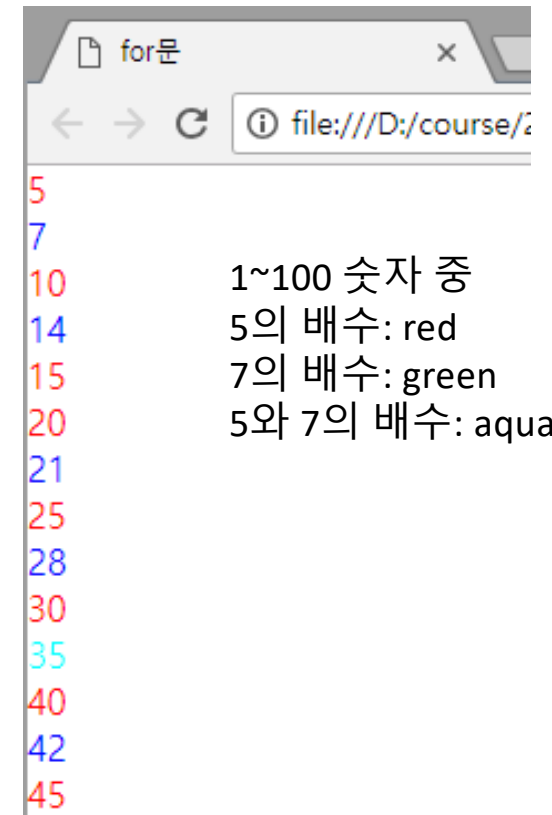
- Complete your for_1_test.html
- Complete your for_2_test.html

```
for (var a = initialized; condition; update) {  
    Code1;  
}
```

```
var a = initialized;  
while (condition){  
    Code1;  
    update;  
}
```



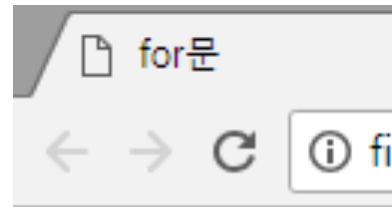
반복1
반복2
반복3
반복4
반복5
반복6
반복7
반복8
반복9
반복10



Repetition control - break

```
for (var a = initialized; condition; update) {  
    if (condition2){  
        break;  
    }  
    Code1;  
}  
var a = initialized;  
while (condition){  
    Code1;  
    update;  
    if (condition2) break;  
}
```

- Modify your for_1_test.html

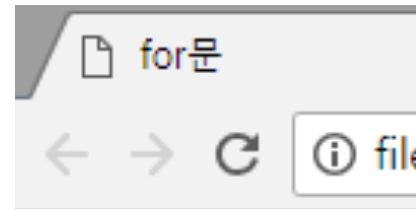


반복1
반복2
반복3
반복4
반복5

Repetition control - continue

```
for (var a = initialized; condition; update) {  
    if (condition2){  
        continue;  
    }  
    Code1;  
}  
var a = initialized;  
while (condition){  
    update;  
    if (condition2) continue;  
    Code1;  
}
```

- Modify your for_1_test.html

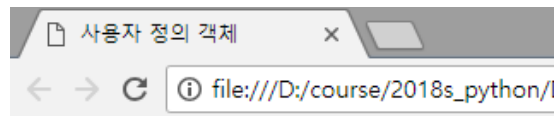


반복1
반복3
반복5
반복7
반복9

Basic Javascript object

```
var s = new Object();
```

Complete your obj1_test.html



tv 객체 메서드 호출

tv 색상: white
tv 가격: 300000

car 객체 메서드 호출

car 색상: black
car 가격: 5000000

```
6 <script>
7   var tv = new Object( );
8   tv.color = "white";
9   tv.price = 300000;
10  tv.info = function( ) {
11      document.write("tv 색상: " + this.color, "<br>");
12      document.write("tv 가격: " + this.price, "<br>");
13  }
14
15  var car = {
16      color: "black",
17      price: 5000000,
18      info: function( ) {
19          document.write("car 색상: " + this.color, "<br>");
20          document.write("car 가격: " + this.price, "<br>");
21      }
22  };
23
24  document.write("<h1>tv 객체 메서드 호출</h1>");
25  tv.info();
26  document.write("<h1>car 객체 메서드 호출</h1>");
27  car.info();
28 </script>
```


Date object

```
var t1 = new Date(); //today  
var t2 = new Date("2002/5/31");  
var t3 = new Date(2018,5,15);
```

날짜 관련 메서드

날짜 정보를 가져올 때(GET)		날짜 정보를 수정할 때(SET)	
getFullYear()	연도 정보를 가져옴	setFullYear()	연도 정보만 수정함
getMonth()	월 정보를 가져옴(현재 월 - 1)	setMonth()	월 정보만 수정함(월 - 1)
getDate()	일 정보를 가져옴	setDate()	일 정보만 수정함
getDay()	요일 정보를 가져옴(일: 0 ~ 토: 6)	'요일'은 날짜를 바꾸면 자동으로 바뀌므로 setDay()는 없음	
getHours()	시 정보를 가져옴	setHours()	시 정보만 수정함
getMinutes()	분 정보를 가져옴	setMinutes()	분 정보만 수정함
getSeconds()	초 정보를 가져옴	setSeconds()	초 정보만 수정함
getMilliseconds()	밀리초 정보를 가져옴(1/1,000초 단위)	setMilliseconds()	밀리초 정보만 수정함
getTime()	1970년 1월 1일부터 경과된 시간을 밀리초로 표기함	setTime()	1970년 1월 1일부터 경과된 시간을 밀리초로 수정함
toGMTString()	GMT 표준 표기 방식으로 문자형 데이터로 반환함	toLocaleString()	운영 시스템 표기 방식으로 문자형 데이터로 반환함

Date object example 1

■ Complete your date_ob1_test.html

```
<script>
var today = new Date( );
var nowMonth = today.getMonth( ),
    nowDate = today.getDate( ),
    nowDay = today.getDay( );

document.write("<h1>오늘 날짜 정보</h1>");
document.write("현재 월: " + nowMonth, "<br>");
document.write("현재 일: " + nowDate, "<br>");
document.write("현재 요일: " + nowDay, "<br>");

var worldcup = new Date(2002, 4, 31);
var theMonth = worldcup.getMonth( ),
    theDate = worldcup.getDate( ),
    theDay = worldcup.getDay( );

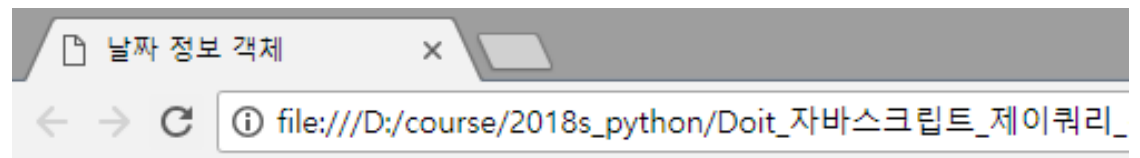
document.write("<h1>월드컵 날짜 정보</h1>");
document.write("2002월드컵 몇 월: " + theMonth, "<br>");
document.write("2002월드컵 몇 일: " + theDate, "<br>");
document.write("2002월드컵 무슨 요일: " + theDay, "<br>");
</script>
```



Date object example 2

```
// D-day counter  
var t1 = new Date(); // today  
var t2 = new Date(2018,11,31);  
var diff = t2 - t1; // msec  
// 1 day = 1000*60*60*24 msec
```

- Complete your date_ob2_test.html



오늘부터 올해 연말까지 남은 날짜

D-day: 230일 남았습니다.

Math object

- Random number generation between [a,b]
`Math.floor(Math.random()*(b-a+1))+a;`

수학 객체의 메서드 및 상수	
종류	설명
Math.abs(숫자)	숫자의 절댓값을 반환합니다.
Math.max(숫자 1, 숫자 2, 숫자 3, 숫자 4)	숫자 중 가장 큰 값을 반환합니다.
Math.min(숫자 1, 숫자 2, 숫자 3, 숫자 4)	숫자 중 가장 작은 값을 반환합니다.
Math.pow(숫자, 제곱값)	숫자의 거듭제곱값을 반환합니다.
Math.random()	0~1 사이의 난수를 반환합니다. [0,1)
Math.round(숫자)	소수점 첫째 자리에서 반올림하여 정수를 반환합니다.
Math.ceil(숫자)	소수점 첫째 자리에서 무조건 올림하여 정수를 반환합니다.
Math.floor(숫자)	소수점 첫째 자리에서 무조건 내림하여 정수를 반환합니다.
Math.sqrt(숫자)	숫자의 제곱근값을 반환합니다.
Math.PI	원주율 상수를 반환합니다.

Array object

```
var a = new Array();  
a[0] = 30;  
a[1] = "hello";  
a[2] = true;  
var b = new Array(30, "hello", true);  
var c = [30, "hello", true];
```

배열 객체의 메서드 및 속성

종류	설명
join(연결 문자)	배열 객체의 데이터를 연결 문자 기준으로 1개의 문자형 데이터로 반환합니다.
reverse()	배열 객체의 데이터 순서를 거꾸로 바꾼 후 반환합니다.
sort()	배열 객체의 데이터를 오름차순으로 정렬합니다.
slice(index1, index2)	배열 객체의 데이터 중 원하는 인덱스 구간만큼 잘라서 배열 객체로 가져옵니다.
splice()	배열 객체의 지정 데이터를 삭제하고 그 구간에 새 데이터를 삽입할 수 있습니다.
concat()	2개의 배열 객체를 하나로 결합합니다. concat()
pop()	배열에 저장된 데이터 중 마지막 인덱스에 저장된 데이터를 삭제합니다. pop()
push(new data)	배열 객체의 마지막 인덱스에 새 데이터를 삽입합니다.
shift()	배열 객체에 저장된 데이터 중 첫 번째 인덱스에 저장된 데이터를 삭제합니다.
unshift(new data)	배열 객체의 가장 앞의 인덱스에 새 데이터를 삽입합니다.
length	배열에 저장된 총 데이터의 개수를 반환합니다.

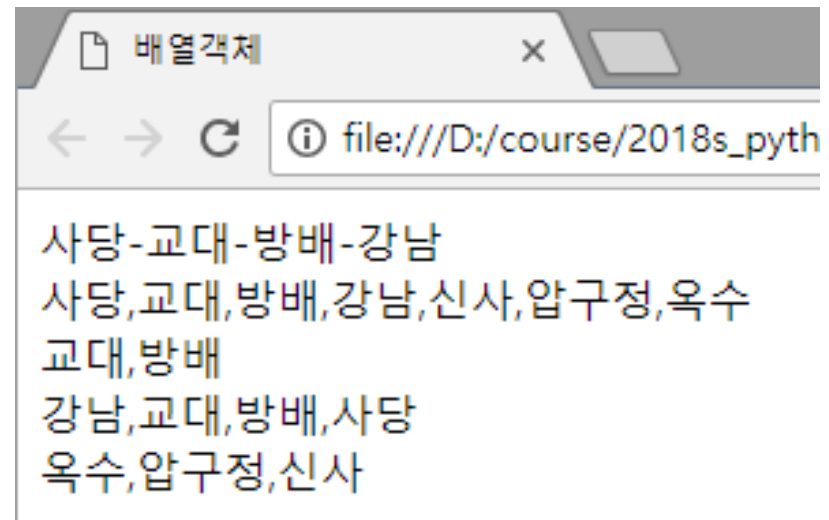
Array object example

- Complete your array_ob2_test.html

```
<script>
```

```
var arr_1 = ["사당", "교대", "방배", "강남"];  
var arr_2 = ["신사", "압구정", "옥수"];  
  
var result = arr_1.join("-");  
document.write(result, "<br>");  
  
result = arr_1.concat(arr_2);  
document.write(result, "<br>");  
  
result = arr_1.slice(1, 3);  
document.write(result, "<br>");  
  
arr_1.sort( );  
document.write(arr_1, "<br>");  
  
arr_2.reverse( );  
document.write(arr_2, "<br>");
```

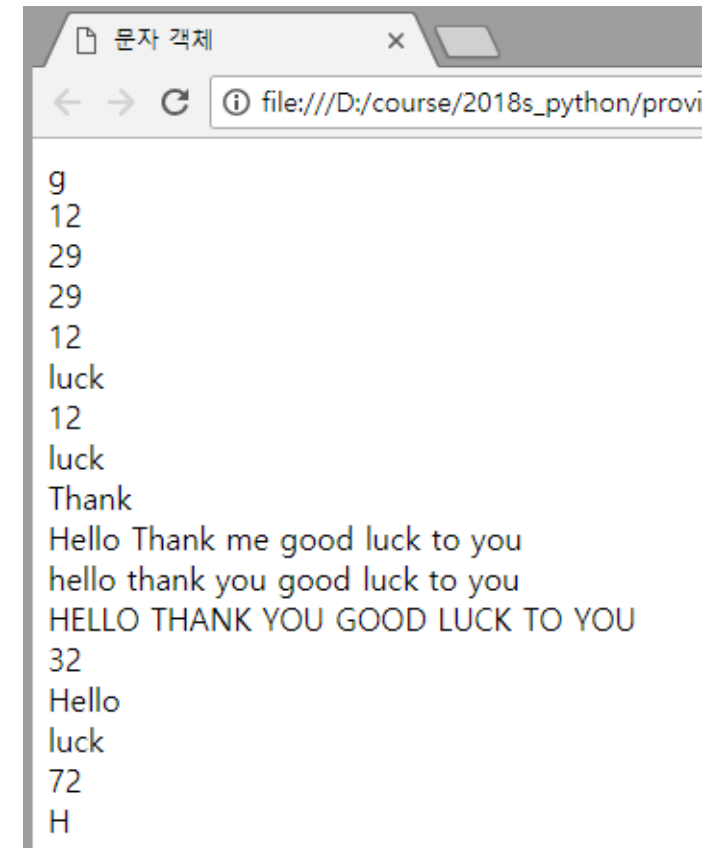
```
</script>
```



String object

```
var s1 = new String("hello javascript");  
var s2 = "hello python";
```

Refer to your string_ob1.html



String methods

문자열 객체의 메서드 및 속성

종류	설명
charAt(index)	문자열에서 인덱스 번호에 해당하는 문자를 반환합니다. 예 var str="web he she"; str.charAt(2); → "b"를 반환함
indexOf("찾을 문자")	문자열에서 왼쪽부터 찾을 문자와 일치하는 문자를 찾아 제일 먼저 일치하는 문자의 인덱스 번호를 반환합니다. 만일 찾는 문자가 없으면 -1을 반환합니다. 예 var str="web he she he"; str.indexOf("he"); → 4를 반환함
lastIndexOf("찾을 문자")	문자열에서 오른쪽부터 찾을 문자와 일치하는 문자를 찾아 제일 먼저 일치하는 문자의 인덱스 번호를 반환합니다. 만일 찾는 문자가 없으면 -1을 반환합니다. 예 var str="web he she he"; str.lastIndexOf("he"); → 11을 반환함
match("찾을 문자")	문자열에서 왼쪽부터 찾을 문자와 일치하는 문자를 찾아 제일 먼저 찾은 문자를 반환합니다. 만일 찾는 문자가 없으면 null을 반환합니다. 예 var str="web he she he"; str.match("boy"); → null을 반환함
replace("바꿀 문자", "새 문자")	문자열에서 왼쪽부터 바꿀 문자와 일치하는 문자를 찾아 제일 먼저 찾은 문자를 새 문자로 치환합니다. 예 var str="web he she"; str.replace("web","html"); → "html he she"를 반환함
search("찾을 문자")	문자열에서 왼쪽부터 찾을 문자와 일치하는 문자를 찾아 제일 먼저 일치하는 인덱스 번호를 반환합니다. 예 var str="web he she"; str.search("he"); → 4를 반환함
slice(a, b)	a개의 문자를 자르고 b번째 이후에 문자를 자른 후 남은 문자를 반환합니다. 예 var str="hello javascript" str.slice(3, 7); → "lo j"를 반환함 str.slice(3, 7);은 문자열에서 3글자 "hel"까지 자르고 7번째 이후 글자인 "j" 이후부터 "avascript"를 자른 후 나머지 "lo j"를 반환합니다. 예 var str="hello javascript" str.slice(3, -3); → "lo javascr"를 반환함 여기에서 -1은 뒤에서부터 첫 번째 글자를 가리킵니다.
substring(a, b)	a 인덱스부터 b 인덱스 이전 구간의 문자를 반환합니다. 예 var str="hello javascript" str.substring(3, 7); → "lo j"를 반환함 예 var str="hello javascript" str.substring(3, -3); → "hel"을 반환함. 여기에서 -3은 인덱스 0을 가리킵니다. 그러므로 인덱스 0부터 인덱스 3 이전 구간의 문자를 반환합니다.

substr(a, 문자 개수)	문자열에 a 인덱스부터 지정한 문자 개수만큼 문자열을 반환합니다. 예 var str="hello javascript" str.substr(3, 2); → "lo"를 반환함 str.substr(3, 2);은 3 문자열에서 인덱스 3인 "l"부터 2글자를 가져와서 "lo"를 반환합니다.
split("문자")	지정한 문자를 기준으로 문자 데이터를 나누어 배열에 저장하여 반환합니다. 예 var str="webkmobilek2002"; <div style="display: flex; justify-content: space-around; align-items: center;"><div>(Index)</div><div>(0)</div><div>(1)</div><div>(2)</div></div> <div style="display: flex; justify-content: space-around; align-items: center;"><div>var arr=str.split("k"); → arr=</div><div style="border: 1px solid black; padding: 5px; display: flex; gap: 10px;"><div style="border: 1px solid black; padding: 5px;">"web"</div><div style="border: 1px solid black; padding: 5px;">"mobile"</div><div style="border: 1px solid black; padding: 5px;">"2002"</div></div></div>
toLowerCase()	문자열에서 영문 대문자를 모두 소문자로 바꿉니다. 예 var str="ABC"; str.toLowerCase(); → "abc"를 반환함
toUpperCase()	문자열에서 영문 소문자를 모두 대문자로 바꿉니다. 예 var str="abc"; str.toUpperCase(); → "ABC"를 반환함
length	문자열에서 문자의 개수를 반환합니다. 예 var str="hello welcome"; str.length → 13을 반환함
concat("새로운 문자")	문자열에 새로운 문자열을 결합합니다. 예 var str="hello"; str.concat("web"); → "hello web"을 반환함
charCodeAt(index)	문자열 index에 해당 문자의 아스키 코드값을 반환합니다. 예 var str = "ABC", str.charCodeAt(0) → "A"의 아스키 코드값 65를 반환함 좀 더 자세한 아스키 코드값을 확인하고 싶다면 다음 주소를 참고하세요. http://office.microsoft.com/ko-kr/word-help/HA010167539.aspx
fromCharCode(아스키 코드 값)	아스키 코드값에 해당하는 문자를 반환합니다. 예 String.fromCharCode(65); → "A"를 반환함
trim()	문자의 앞 또는 뒤에 공백 문자열을 삭제합니다(" hello " → "hello"). 예 str="hello"; str.trim(); → 공백이 제거된 "hello"를 반환함