**함 수**

|  |  |
| --- | --- |
| 1 | 함수에 대한 설명 중 틀린 것을 고르세요. 4, 6  (1) 여러 번 사용할 가능성이 높은 코드를 블록으로 감싼(지역화 시킨)후 이름을 붙인 단위가 함수다.O  (2) 다소 복잡하여 다시 작성하기에 부담스러운 경우에도 함수로 정의해 놓으면 좋다 O  (3) 관련성있는 코드간에 분리시킬 목적시에도 함수를 이용할 수 있다. O  (4) 함수 내에 선언된 변수는 전역변수이며 그 함수 내에서만 사용 가능하다. X  (5) 함수는 호출할 때만 동작한다 O  (6) 변수는 메모리에 생성되지만, 함수는 메모리에 생성되지 않는다 X |
| 2 | 함수에 대한 설명 중 틀린 것을 고르면? 5  (1) 함수는 이름 뒤에 언제나 괄호( )가 붙어있어야 한다. O  (2) 자바스크립트에서 개발자가 함수를 정의하려면 function 키워드를 사용해야 하면, 이러한 함수를  사용자 정의 함수라 한다 O  (3) 함수를 작동시키려면 반드시 호출해야 한다. O  (4) 전역변수는 어떤 함수에서건 접근이 가능하다O  (5) 실행부가 변수를 만나면 메모리에 올리지만, 함수의 경우는 메모리에 올리지 않는다 |
| 3 | 아래 코드의 실행 결과는? 3  **var a=3;**  **function test(){**  **a=5;**  **}**  **document.write(a);** |
| 4 | 아래 코드의 수행 결과는? 아무것도 안뜸.  **function test(){**  **return;**  **document.write("함수를 호출했습니다.");**  **}**  **test();** |
| 5 | 함수에 대한 설명 중 틀린 것은?1  (1) 함수의 실행 중 return을 만나면 실행부는 return이 기재된 line에 멈추어 있으며 실행을 멈춘다. X  (2) return 뒤에 붙는 값을 리턴값 이라고 한다. O  (3) return 뒤에 값을 붙일 경우, 실행부는 이 함수를 호출한 곳으로 되돌아가되, 값을 가지고 간다. O  (4) 함수는 호출하지 않으면 절대로 작동되지 않는다. O |
| 6 | 결과를 예상하세요. myFunc 함수를 호출했습니다 alert 뜨고 3 alert 뜬다  **function myFunc(){**  **alert(“myFunc 함수를 호출했습니다.”);**  **return 3;**  **}**  **function testFunc(){**  **var x=myFunc();**  **alert(x);**  **}**  **testFunc();** |
| 7 | 출력 결과를 예상하세요. 2  <script language=”javascript”>  **var x=0;**  **function a(){**  **return;**  **x=1;**  **}**  **function b(){**  **x++;**  **a();**  **x++;**  **}**  **b();**  **document.write(x);**  </script> |
| 8 | 원하는 단을 입력받아 출력하는 함수를 작성하세요.  Ex) 3을 입력시 3단을 출력하는 함수 |
| 9 | 함수의 정의가 아래과 같을 때 실행 결과에 대하여 틀리게 설명한 것은?4  function myFunc(x , y){ --------------- (1)  var result; -------------------- (2)  result=x+y; -------------------- (3)  return result; -------------------- (4)  alert(“계산결과는”+result); -------------- (5)  }  (1) 매개변수가 2개이며, 리턴값이 있는 함수를 정의한 것이다.  (2) result 는 (2)에서 선언된 변수이므로, myFunc() 함수의 지역변수라고 할 수 있다.  (3) 이 함수를 호출 할 때는 myFunc(10, 20) 과 같이 매개변수가 2개 있는 형태로 호출해야 한다.  (4) return 문에 의해 결과 값을 반환한 후 실행부는 아랫줄의 alert() 을 실행한다.  (5) alert()은 실행되어 질 수 없다. |