1. 다음과 같은 obj 객체가 있을 때 이어지는 소스에서 에러인 것과 아닌 것을 구분해라. ( 에러 : F 에러 아닌 것 : T )  
   var obj = { };
   1. var obj.x = 10;
   2. obj[“2”] = 20;
   3. obj.3 = 30;
   4. alert( obj.y );
2. 다음 소스 코드의 중간 실행 결과를 적어라.

var a = (function( num ) {

if(arguments.length === arguments.callee.length) {

a = 10;

return num + a;

}else {

alert( arguments ); -------------- (1)

var a = 20;

return arguments.callee( arguments[2] );

}

})( 50, 100, 60 );

alert( a ); -------------- (2)

1. 다음 소스 코드의 중간 실행 결과를 적어라.  
   function curryingFunc( a ) {

return { x : function() { return ++a; },

y : function() { return a--; },

z : function() { return a += 2; } };

}

var foo = curryingFunc( 10 );

var x = foo.x;

var y = foo.y;

var z = foo.z;

alert( z() ); ------------------------ (1)

alert( y() ); ------------------------ (2)

alert( z() ); ------------------------ (3)

alert( x() ); ------------------------ (4)

1. 다음 소스 코드에서 미완성 된 부분을 작성해라.

**function average( ) {**

// 전달받은 인자들의 평균을 구해서 출력한다.

**alert( /\* 결과 출력 \*/ );**

**}**

ex) 호출 예시 : 아래와 같이 호출 했을 때 올바른 값이 나와야 한다.

- average( 10, 5, 8, 6, 8); // 7.4

- average( 10, 40, 20, 30, 20, 60, 80, 20, 40); // 35.5555…

- average( 10, 40, 20, 30, 20, 60, 80); // 37.1428….