**JavaScript Coding Quiz - Multiple Choice Questions (MCQ)**

**1. What will be the output of the following code snippet?**

|  |
| --- |
| var x = 5;  console.log(x + "5"); |

a) 55  b) 10  c) "55"  d) "5 + 5"

**2. What is the correct way to check if a variable x is of type "string" in JavaScript?**

a) *if (x === "string")* b) *if (typeOf x === "string")* c)*if (typeof x === "string")* d) *if (x.type === "string")*

**3. What will be the output of the following code snippet?**

|  |
| --- |
| var num = 10;  function increment() {  num++;  }  increment();  console.log(num); |

a) 9  b) 10  c) 11  d) NaN

**4. What is the output of the following code snippet?**

console.log(2 + 3 + "5");

a) 55  b) 10  c) 235  d) "55"

**5. What is the output of the following code snippet?**

|  |
| --- |
| var x = 10;  function foo() {  var x = 5;  console.log(x);  }  foo();  console.log(x); |

a) 10, 10  b) 10, 5  c) 5, 10  d) 5, 5

**6. What will be the output of the following code snippet?**

|  |
| --- |
| var name = "John";  function sayName() {  console.log(name);  var name = "Jane";  }  sayName(); |

a) John  b) Jane  c) Undefined  d) ReferenceError

**7. What will be the output of the following code snippet?**

|  |
| --- |
| var fruits = ["apple", "banana", "orange"];  console.log(fruits[1]); |

a) "apple"  b) "banana"  c) "orange"  d) undefined

**8. What will be the output of the following code snippet?**

|  |
| --- |
| var arr = [1, 2, 3, 4, 5];  var slicedArr = arr.slice(1, 4);  console.log(slicedArr); |

a) [1, 2, 3]  b) [2, 3, 4]  c) [2, 3, 4, 5]  d) [1, 2, 3, 4]

**9. How do you convert a string "123" to a number in JavaScript?**

a) parseInt("123")  b) convertNumber("123")  c) toNumber("123")  d) number("123")

**10. What will be the output of the following code snippet?**

|  |
| --- |
| var numbers = [1, 2, 3, 4, 5];  var sum = 0;  numbers.forEach(function (num) {  sum += num;  });  console.log(sum); |

a) 1  b) 15  c) 10  d) [1, 2, 3, 4, 5]

**11. What will be the output of the following code snippet?**

|  |
| --- |
| var numbers = [1, 2, 3, 4, 5];  var squaredNumbers = numbers.map(function (num) {  return num \* num;  });  console.log(squaredNumbers); |

a) [1, 2, 3, 4, 5]  b) [2, 4, 6, 8, 10]  c) [1, 4, 9, 16, 25]  d) [1, 8, 27, 64, 125]

**12. What will be the output of the following code snippet?**

|  |
| --- |
| var numbers = [1, 2, 3, 4, 5];  var evenNumbers = numbers.filter(function (num) {  return num % 2 === 0;  });  console.log(evenNumbers); |

a) [1, 3, 5]  b) [2, 4]  c) [1, 2, 3, 4, 5]  d) [true, false, true, false, true]

**13. What will be the output of the following code snippet?**

|  |
| --- |
| function outer() {  var x = 10;  function inner() {  console.log(x);  }  return inner;  }  var closureFunction = outer();  closureFunction(); |

a) 10  b) 20  c) undefined d) ReferenceError

**14. What will be the output of the following code snippet?**

|  |
| --- |
| var x = 5;  function outer() {  function inner() {  console.log(x);  }  var x = 10;  return inner;  }  var closureFunction = outer();  closureFunction(); |

a) 5  b) 10  c) 20  d) ReferenceError

**15. What will be the output of the following code snippet?**

|  |
| --- |
| function delayLog() {  for (var i = 1; i <= 5; i++) {  setTimeout(function () {  console.log(i);  }, 1000);  }  }  delayLog(); |

a) 1, 2, 3, 4, 5  b) 5, 5, 5, 5, 5  c) 6, 6, 6, 6, 6  d) 1, 6, 6, 6, 6

**16. What will be the output of the following code snippet?**

|  |
| --- |
| function foo() {  return new Promise(function (resolve, reject) {  setTimeout(function () {  resolve("Promise resolved.");  }, 1000);  });  }  console.log("Start");  foo().then(function (result) {  console.log(result);  });  console.log("End"); |

a) Start, Promise resolved., End  b) Start, End, Promise resolved.

c) Promise resolved., Start, End  d) Start, End, Promise resolved.

**17. What will be the output of the following code snippet?**

|  |
| --- |
| async function foo() {  return "Async function.";  }  console.log("Start");  foo().then(function (result) {  console.log(result);  });  console.log("End"); |

a) Start, Async function., End  b) Start, End, Async function.

c) Async function., Start, End  d) Start, End, Async function.

**18. What will be the output of the following code snippet?**

|  |
| --- |
| function\* fibonacciGenerator() {  let a = 0;  let b = 1;  while (true) {  yield a;  [a, b] = [b, a + b];  }  }  const fibonacci = fibonacciGenerator();  console.log(fibonacci.next().value);  console.log(fibonacci.next().value);  console.log(fibonacci.next().value);  console.log(fibonacci.next().value);  console.log(fibonacci.next().value); |

a) 0, 1, 1, 2, 3  b) 0, 1, 2, 3, 4  c) 1, 1, 2, 3, 5  d) 0, 1, 1, 3, 5

**19. What will be the output of the following code snippet?**

|  |
| --- |
| Promise.resolve("First resolved.")  .then(function (result) {  console.log(result);  return Promise.resolve("Second resolved.");  })  .then(function (result) {  console.log(result);  return Promise.resolve("Third resolved.");  })  .then(function (result) {  console.log(result);  return Promise.reject(new Error("Fourth rejected."));  })  .catch(function (error) {  console.log(error.message);  }); |

a) First resolved., Second resolved., Third resolved., Fourth rejected.

b) First resolved., Second resolved., Third resolved., Error: Fourth rejected.

c) First resolved., Second resolved., Third resolved., Promise {<rejected>: Error: Fourth rejected.}

d) First resolved., Second resolved., Third resolved., Promise {<rejected>: "Fourth rejected."}

**20. What will be the output of the following code snippet?**

|  |
| --- |
| async function asyncFunction() {  throw new Error("Error in async function.");  }  console.log("Start");  asyncFunction().catch(function (error) {  console.log(error.message);  });  console.log("End"); |

a) Start, Error in async function., End  b) Start, End, Error in async function.

c) Error in async function., Start, End  d) Start, End, Error in async function.