# Trivia Qs:

## i++ a. Incrementor b. Operator c. Iterator d. Decrementor \* Incrementor Incrementors/Decrementors are used to increase/reduce the value of a variable by 1

## var hello = function() { console.log(“I am saying Hello”); }; Is an example of a: a. Method b. Variable c. Object d. Function \* Function A function contains code that will be executed by an event or by a call to the function. In this case, we would call the function by writing a command: hello(); which would display the text "i am saying hello" in the console.

1. var = donaldTrumpkin is an example of a:   
   a. Variable   
   b. String  
   c. Object  
   d. Method  
     
   \*Variable   
   Variables are used to hold values or expressions
2. In the function below, what does the text '(w, l)' represent?   
   var area = function (w, l) {   return w \* l; };  
     
   a. Variables  
   b. Parameters  
   c. Strings  
   d. Object  
     
   \* Parameters  
   A parameter is a variable that is processed by the function to generate a given result. In this case, if we run the function as follows: area(2,3); we will get a result of 6. To further clarify, parameters are the variables in the declaration of a function (w,l) while arguments are the specific values that get passed when running the function (2,3)
3. var i; for (i = 0; i < 2; i++) {   
    console.log(“i is now equal to “ + i);   
    }  
     
    a. Function  
    b. Object  
    c. For Loop  
    d. While Loop

For Loop\*  
 In JavaScript, there are two different kind of Loops:  
 For - Loops through a block of code a specified number of times.  
 While - Loops through a block of code while a specified condition is true.

1. For the object below, what does the word ‘name’ represent?   
     
   var dog = {name: “Daisy”, age: 6};  
     
   a. Property  
   b. Variable   
   c. Value  
   d. String  
     
   \* Property:  
   In this object, 'name' is a property. A property is a type of information that describes the object and is always paired with a value.  
     
   Objects can written in two different ways, depending on how the properties are defined within them:  
     
   1. Object Literal Notation:  
     
   math xmlns=¨http://www.w3.org/1998/Math/MathML¨»«mi mathvariant=¨normal¨»var«/mi»«mo»§nbsp;«/mo»«mi mathvariant=¨normal¨»dog«/mi»«mo»§nbsp;«/mo»«mo»=«/mo»«mo»§nbsp;«/mo»«m  
   2. Object Constructor Notation  
     
   math xmlns=¨http://www.w3.org/1998/Math/MathML¨»«mi mathvariant=¨normal¨»var«/mi»«mo»§nbsp;«/mo»«mi mathvariant=¨normal¨»dog«/mi»«mo»§nbsp;«/mo»«mo»=«/mo»«mo»§nbsp;«/mo»«m  
   In addition to representing properties in two different ways when creating an object, you can also access properties two different ways:  
     
   1. Dot Notation  
     
   math xmlns=¨http://www.w3.org/1998/Math/MathML¨»«mi mathvariant=¨normal¨»var«/mi»«mo»§nbsp;«/mo»«mi mathvariant=¨normal¨»dogName«/mi»«mo»§nbsp;«/mo»«mo»=«/mo»«mo»§nbsp;«/m  
   2. Bracket Notation:  
     
   math xmlns=¨http://www.w3.org/1998/Math/MathML¨»«mi mathvariant=¨normal¨»var«/mi»«mo»§nbsp;«/mo»«mi mathvariant=¨normal¨»dogName«/mi»«mo»§nbsp;«/mo»«mo»=«/mo»«mo»§nbsp;«/m
2. var dogName = dog[“name”]  
     
   a. Object Literal Notation  
   b. Bracket Notation  
   c. Dot Notation  
   d. Object Literal Notation  
     
   \*Bracket Notation  
   When you access a property, you are setting a variable equal to the value of a property in a particular object.  
     
   In this example, we are setting variable dogName equal to the value of the 'name' property of the 'dog' object.
3. var numbers = [1,2,3] is an example of:   
     
   a. Array  
   b. Function  
   c. Object  
   d. Method  
     
   \*Array  
   An array is a special variable which can hold more than one value at a time.
4. var i = 0; while ( i < 2 ) { console.log(“i is now” + i); i++ }  
     
   a. While Loop  
   b. Incrementor  
   c. For Loop  
   d. Function  
     
   \*While Loop  
   In JavaScript, there are two different kind of Loops:  
   For - Loops through a block of code a specified number of times.  
   While - Loops through a block of code while a specified condition is true.
5. var dog = {name: “Daisy”, age: 6};  
     
   Is an example of creating an object using:   
     
   a. Dot notation  
   b. Object Literal Notation  
   c. Bracket Notation  
   d. Object Constructor Notation  
     
   \*Object Literal Notation  
   Objects can written in two different ways, depending on how the properties are defined within them:  
     
   Objects can written in two different ways, depending on how the properties are defined within them:  
     
   1. Object Literal Notation:  
     
   math xmlns=¨http://www.w3.org/1998/Math/MathML¨»«mi mathvariant=¨normal¨»var«/mi»«mo»§nbsp;«/mo»«mi mathvariant=¨normal¨»dog«/mi»«mo»§nbsp;«/mo»«mo»=«/mo»«mo»§nbsp;«/mo»«m  
   2. Object Constructor Notation  
     
   math xmlns=¨http://www.w3.org/1998/Math/MathML¨»«mi mathvariant=¨normal¨»var«/mi»«mo»§nbsp;«/mo»«mi mathvariant=¨normal¨»dog«/mi»«mo»§nbsp;«/mo»«mo»=«/mo»«mo»§nbsp;«/mo»«m  
   In addition to representing properties in two different ways when creating an object, you can also access properties two different ways:  
     
   1. Dot Notation  
     
   math xmlns=¨http://www.w3.org/1998/Math/MathML¨»«mi mathvariant=¨normal¨»var«/mi»«mo»§nbsp;«/mo»«mi mathvariant=¨normal¨»dogName«/mi»«mo»§nbsp;«/mo»«mo»=«/mo»«mo»§nbsp;«/m  
   2. Bracket Notation:
6. switch (GPA) {  
    case 90:  
    letterGrade = “A”;  
    break;  
    case 80:  
    letterGrade = “B”;   
    break;

case 70:   
 letterGrade = “C”;  
 break;  
 case 60:   
 letterGrade = “D”;  
 break;  
 case 50:   
 letterGrade = “F”;  
 break;   
 default:   
 letterGrade = “A+”;

break;  
 }  
  
 a. If…Else Statement  
 b. Ternary Operator

c. Else If Statement

d. Switch Statement

\* Switch Statement

Switch statements are a shorthand way to write if else statements when there are many different cases, and each case has a different outcome.

1. if (name.length > 0) {  
    console.log (“Please enter your name.”)  
    }   
    else {  
    console.log(“Hello “ + name);  
    }  
      
    is an example of:   
      
    a. If Statement  
    b. If…Else Statement  
    c. Else If Statement   
    d. Switch Statement  
     
    \* If…Else Statement   
    If the condition (name.length > 0) is true, statement1 will be executed. Otherwise, statement2 will be executed.
2. var dogName = dog.name  
     
    is an example of accessing an object’s property using:   
      
    a. Dot Notation  
    b. Bracket Notation  
    c. Object Literal Notation  
    d. Object Constructor Notation   
      
    \* Dot Notation   
    When you access a property, you are setting a variable equal to the value of a property in a particular object.  
    In this example, we are setting variable dogName equal to the value of the 'name' property of the 'dog' object.

1. console.log(“You “ + (grade > 50 ? “Passed!” : “Failed!));  
      
    is an example of:   
      
    a. If Statement  
    b. If…Else Statement   
    c. Switch Statement   
    d. Ternary Statement   
      
    \* Ternary Statement   
    Ternary operators are a shorthand way of writing if else statements.
2. Which of the options below are data types:   
      
    a. Number, String, Function.   
    b. String, Boolean, Data,   
    c. Boolean, Number, Sting.  
    d. Number, String, Boolean.  
      
    \*Number, String, Boolean. (Not Sting!)
3. var greeting = function (name) {  
    console.log(“Hello “ + name);  
    };  
      
    is an example of a:   
    a. Method  
    b. Object  
    c. Function  
    d. Variable  
      
    \*Function   
    A function contains code that will be executed by an event or by a call to the function.
4. var dog = (); dog.name = “Daisy”; dog.age = 6;   
      
    is an example of creating an object using:   
      
    a. Object Literal Notation  
    b. Object Constructor Notation  
    c. Dot Notation  
    d. Bracket Notation  
      
    \*Object Constructor Notation   
    Objects can written in two different ways, depending on how the properties are defined within them:  
    1. Object Literal Notation:  
    math xmlns=¨http://www.w3.org/1998/Math/MathML¨»«mi mathvariant=¨normal¨»var«/mi»«mo»§nbsp;«/mo»«mi mathvariant=¨normal¨»dog«/mi»«mo»§nbsp;«/mo»«mo»=«/mo»«mo»§nbsp;«/mo»«m  
    2. Object Constructor Notation  
    math xmlns=¨http://www.w3.org/1998/Math/MathML¨»«mi mathvariant=¨normal¨»var«/mi»«mo»§nbsp;«/mo»«mi mathvariant=¨normal¨»dog«/mi»«mo»§nbsp;«/mo»«mo»=«/mo»«mo»§nbsp;«/mo»«m  
    In addition to representing properties in two different ways when creating an object, you can also access properties two different ways:  
    1. Dot Notation  
    math xmlns=¨http://www.w3.org/1998/Math/MathML¨»«mi mathvariant=¨normal¨»var«/mi»«mo»§nbsp;«/mo»«mi mathvariant=¨normal¨»dogName«/mi»«mo»§nbsp;«/mo»«mo»=«/mo»«mo»§nbsp;«/m  
    2. Bracket Notation:  
    math xmlns=¨http://www.w3.org/1998/Math/MathML¨»«mi mathvariant=¨normal¨»var«/mi»«mo»§nbsp;«/mo»«mi mathvariant=¨normal¨»dogName«/mi»«mo»§nbsp;«/mo»«mo»=«/mo»«mo»§nbsp;«/m
5. Which built-in method returns the length of a string?  
      
    a. length()  
    b. index()  
    c. size()  
    d. value()  
      
    \* length()   
    length() method returns the length of the string.
6. Which of the following function of a String object returns the characters in a string between two indexes into the string?  
      
    a. slice()  
    b. split()  
    c. substr()  
    d. substring()   
      
    \* substring()   
    substring() returns the characters in a string between two indexes into the string.
7. Which built-in method reverses the order of the elements in an array?   
      
    a. changeOrder(order)  
    b. reverse()   
    c. sort(order)   
    d. charCodeAt()   
      
    \* reverse()   
    reverse() method reverses the order of the elements of an array -- the first becomes the last, and the last becomes the first.