**Devyn Heuer**

**JavaScript 1**

**1. Name the three ways to declare a variable?**

Let

Var

Const

**2. Which of the three variable declarations should you avoid and why?**

This is dependent on your need of the variable.

Let variables are only available in their block. So if you required the variable else where in your code then Let would not be the best option.

Var variables are global variables. In the case that you require a variable relative only to a function that it is inside you would then rather use Let.

Const variables can not be changed. In the case that you require your variable to change based off of users input then a Const variable type would not work.

So it depends on the use of the variable in the code.

**3. What rules should you follow when naming variables?**

Variables names should be descriptive and specific.

Camel Case the variable names. E.g. myExponent.

Do not start the variable with a digit.

**4. What should you look out for when using the + operator with numbers and**

**strings?**

You want to look out for the data type being used. If you’re looking for a numeric operation you will use the + operator on the integers/floats/numbers. However this does not work with a string and a numeric value as it will consider the whole line as a string.

**5. How does the % operator work?**

It returns the remainder of two numbers that have been divided.

**6. Explain the difference between == and ===.**

The ‘==’ operator does the type conversion of the operands before comparison, whereas the ‘===’ operator compares the values as well as the data types of the operands.

**7. When would you receive a NaN result?**

When your result is not a valid number, or if one of your operations are undefined. You can also get NaN if you divide 0 by 0.

**8. How do you increment and decrement a number?**

You can increment a number by one by adding ‘++’ to the end/beginning of the variable name. E.g. var myNumber = 6, myNumber++ will make it 7. To decrement a number by one you add ‘–‘ to the end/beginning of the name, --myNumber.

Alternatively you can do a ‘+=’ or ‘-=’ to change the number by an amount of your choosing.

**9. Explain the difference between prefixing and post-fixing increment/decrement**

**operators.**

The post-fixing returns the original value of the variable before it gets incremented or decremented.

Prefixing returns the value of the variable after it has been incremented or decremented.

**10. What is operator precedence and how is it handled in JS?**

This determines the order in which the operators are evaluated. Operators with higher precedence are evaluated first. ‘\*’ > ‘+’.

**11. How do you log information to the console?**

console.log().

**12. What does unary plus operator do to string representations of integers?**

It can convert string representations of integers and floats, as well as the non-string values true, false, and null.

**13. What are the eight data types in JavaScript?**

Boolean, undefined, null, number, bigint, string, symbol, and object.

**14. Which data type is NOT primitive?**

The object data type.

**15. What is the relationship between null and undefined?**

Null: It is the intentional absence of the value. It is one of the primitive values of JavaScript. Undefined: It means the value does not exist in the compiler.

**16. What is the difference between single, double, and backtick quotes for strings?**

There is not a definitive difference between them. However some special cases require the use of two of the options. For example having a apostrophe in a string will require your string to be encased with double quotes and the apostrophe as a single quote. Sometimes backticks are used in order for the programmer to add variables into the string. `Hello+${myName}`.

**17. What is the term for embedding variables/expressions in a string?**

String interpolation.

**18. Which type of quote lets you embed variables/expressions in a string?**

Backticks with Template Literals.

**19. How do you embed variables/expressions in a string?**

Using string concatenation with + symbol.

Using template literals with backticks ( ` ` ) symbol.

**20. How do you escape characters in a string?**

JavaScript uses a backslash in front as an escape character. ‘\’

**21. What is the difference between the slice/substring/substr string methods?**

slice(start, end) extracts parts of a string and returns the extracted parts in a new string.

substr(start, length) extracts parts of a string, beginning at the character at the specified position, and ending at the length from that character specified in the second parameter.

substring(indexStart, indexEnd) It returns the portion of the string that starts at indexStart and ends the character immediately preceding indexEnd.

**22. What are the three logical operators and what do they stand for?**

|| Or operator.

&& And operator.

! Not operator.

**23. What are the comparison operators?**

==.

!=.

===.

!==.

>.

<.

>=.

<=.

**24. What are truthy and falsy values?**

A truthy value is a value that is considered true when encountered in a Boolean context.

All values are truthy unless defined as falsy.

**25. What are the falsy values in JavaScript?**

false, 0, -0, 0n, “”, null, undefined and NaN.

**26. What are conditionals?**

They control behavior in JavaScript and determine whether or not pieces of code can run.

**27. What is the syntax for an if/else conditional?**

if(condition){

run code

} else {

Run code

}

**28. What is the syntax for a switch statement?**

switch(expression){

case value:

code

break;

case value2:

code

break;

default:

}

**29. What is the syntax for a ternary operator?**

Ternary operator takes three operands: a condition followed by a question mark ( ? ), then an expression to execute if the condition is truthy followed by a colon ( : ), and finally the expression to execute if the condition is falsy.

**30. What is nesting?**

When you write code inside of other code. Like nesting if/else if statements.

**31. What are functions useful for?**

They allow you to define a block of code, give it a name and then execute it as many times as you want. This makes for cleaner repeatable code.

**32. How do you invoke a function?**

You use a call() method. Where call will be the name of your function followed by the parenthesis and possibly parameters depending on the function.

**33. What are anonymous functions?**

Functions that do not have a name.

**34. What is function scope?**

Each function creates a new scope. Variables defined inside a function are not accessible (visible) from outside the function.

**35. What are return values?**

The return values provides for passing one value back to the code that called it after everything in the function that needs to run has finished running.

**36. What are arrow functions?**

This is the new way to write anonymous function expressions. It is similar to the lambda functions in Python.