**Day 1 Thoery RnD:**

**Advantages Of Javascript:**

- Speed

- Simplicity

- Popularity

- Rich Interfaces

**Limitations of JavaScript:**

- Client-side JavaScript does not allow the reading or writing of files. It has been kept for the security reason.

- JavaScript could not used for networking applications because there is no such support available.

- JavaScript doesn't have any multithreading or multiprocessor capabilities.

**JavaScript Development Tools:**

- Atom

- Visual Studio

- Goorm

- Webstrom

- Sublime Text

- Eclipse

**Enabling JavaScript in Chrome:**

- Goto Settings -> Privacy & Security -> Site Setting -> JavaScript -> Mark "Sites can use Javascript".

**Warning for Non-JavaScript Browsers:**

- <noscript>

Your browser does not support JavaScript!

</noscript>

**Defer in script:**

- The defer attribute is a boolean attribute. If the defer attribute is set, it specifies that the script is downloaded in parallel to parsing the page, and executed after the page has finished parsing.

**Async in Script:**

- Async scripts are executed as soon as the script is loaded, so it doesn't guarantee the order of execution (a script you included at the end may execute before the first script file).

**Reserved Keywords in JavaScript:**

- abstract - arguments - await\* - boolean

- break - byte - case - catch

- char - class\* - const - continue

- debugger - default - delete - do

- double - else - enum\* - eval

- export\* - extends\* - false - final

- finally - float - for - function

- goto - if - implements - import\*

- in - instanceof - int - interface

- let\* - long - native - new

- null - package - private - protected

- public - return - short - static

- super\* - switch - synchronized - this

- throw - throws - transient - true

- try - typeof - var - void

- volatile - while - with - yield

**What is an operator Javascript:**

- In JavaScript, an operator is a special symbol used to perform operations on operands (values and variables). For example, 2 + 3; // 5. Here + is an operator that performs addition, and 2 and 3 are operands.

**Arthematic Operators:**

+ Addition

- Subtraction

\* Multiplication

\*\* Exponentiation (ES2016)

/ Division

% Modulus (Remainder)

++ Increment

-- Decrement

**Comparison Operators:**

Equal ( == ) Returns true if the operands are equal.

Not equal ( != ) Returns true if the operands are not equal.

Strict equal ( === ) Returns true if the operands are equal and of the same type. See also Object.is and sameness in JS.

Greater than( > ) Returns true if the left operand is greater than the right operand.

Greater than/equal to( >= ) Returns true if the left operand is greater than or equal to the right operand.

Less than( < ) Returns true if the left operand is less than the right operand.

Less than/equal to( >= ) Returns true if the left operand is less than or equal to the right operand.

**JavaScript Bitwise Operators:**

& AND Sets each bit to 1 if both bits are 1.

| OR Sets each bit to 1 if one of two bits is 1.

^ XOR Sets each bit to 1 if only one of two bits is 1.

~ NOT Inverts all the bits.

<< Zero fill left shift Shifts left by pushing zeros in from the right and let the leftmost bits fall off.

>> Signed right shift Shifts right by pushing copies of the leftmost bit in from the left, and let the rightmost bits fall off.

>>> Zero fill right shift Shifts right by pushing zeros in from the left, and let the rightmost bits fall off.

**JavaScript Assignment Operators:**

= x = y x = y

+= x += y x = x + y

-= x -= y x = x - y

\*= x \*= y x = x \* y

/= x /= y x = x / y

%= x %= y x = x % y

<<= x <<= y x = x << y

>>= x >>= y x = x >> y

>>>= x >>>= y x = x >>> y

&= x &= y x = x & y

^= x ^= y x = x ^ y

|= x |= y x = x | y

\*\*= x \*\*= y x = x \*\* y