# Lexical grammar

An overview of JavaScript's lexical grammar including topics such as:

- Spaces
- Line terminators
- Comments
- Identifiers
- Keywords
- Literals

References

- https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Lexical_grammar

# Lexical grammar - Spaces

- Valid spaces include horizontal and vertical tabulation and regular spaces

# Lexical grammar - Line terminators

- Line terminators are normally a line feed/line break \n or a carriage return \r
- If no line terminators are used, contiguous statements will result in a syntax error.
    - For example:
        - let a = "Hello" let b = "world!!!"
- A semi-colon can be used at the end of each statement to avoid syntax errors from the lack of line terminators, but it will make the code less readable.
    - For example:
        - let a = "Hello";let b = "world!!!";

# Lexical grammar - Comments

- Line comments
    - Used to add comments that span a single line
    - The line comment starts with double slash ( // )
    - For example:
        - let a = "Hello" // This is a line comment
- Block comments
    - Used to add comments that span one or more lines
    - The comment starts with a slash and star ( /* ) and ends with a star and slash ( */ )
    - For example:
        - let a = "Hello" /* This is a line comment */

# Lexical grammar - Identifiers

Identifies link values with a name, for example, a variable, a function, object keys, class names. For example

- const amount = 1; // Variable declaration (may also be `let` or `var`)
- function myFunction() {} // Function declaration
- const person = { name: "John Doe" }; // Object keys
- // Class declaration
- class Person {
      #priv = "value"; // Private property
  }

# Lexical grammar - Keywords

- Keywords are special words with specific meaning for JavaScript.
- For example:
    - const, var, let, function, if, else, while, for
- Reference
    - https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Lexical_grammar#reserved_words

# Lexical grammar - Literals

- Individual words that represent values in JavaScript
- For example:
    - "Hello"
    - 'World!!!'
    - 123
    - true
    - false
    - 123.45