JavaScript String to Number Conversion Notes

Overview

- JavaScript handles string-to-number conversion in arithmetic operations, but behavior varies with operators.

Key Concepts

- **Automatic Conversion**:

  - Strings in arithmetic operations (`-`, `*`, `/`) are converted to numbers.

  - Example: `var profit = "200" - "150";` → `profit` is 50 (number).

  - For `+`, JavaScript concatenates, converting numbers to strings.

  - Example: `var result = "200" + 150;` → `result` is `"200150"`.

- **Prompt Issues**:

  - `prompt()` returns a string, leading to concatenation with `+`.

  - Example: `var currentAge = prompt("Enter your age.");  var qualifyingAge = currentAge + 1;` → If input is `"52"`, `qualifyingAge` is `"521"`.

- **Explicit Conversion**:

  - `parseInt()`: Converts string to integer, truncates decimals.

  - Example: `var myInteger = parseInt("1.9999");` → `myInteger` is 1.

  - `parseFloat()`: Converts string to number, preserves decimals.

  - Example: `var myFractional = parseFloat("1.9999");` → `myFractional` is `1.9999`.

- **Error Handling**:

  - Non-numeric strings in math operations return `NaN`.

  - Example: `var profit = "200" - "duck";` → `NaN`.

- **Best Practice**:

  - Use `parseInt` or `parseFloat` explicitly to avoid confusion, even when JavaScript auto-converts.