**🔢 Number Methods**

| **Method** | **What it does** |
| --- | --- |
| toFixed(n) | Rounds to *n* decimal places, returns string |
| toString() | Converts number to string |
| parseInt() | Parses string into integer |
| parseFloat() | Parses string into float |
| isNaN(val) | Checks if value is NaN |
| isFinite(val) | Checks if value is a finite number |
| Number(val) | Converts to number |

**📌 Example:**

let num = 12.3456;

num.toFixed(2); // "12.35"

Number("123"); // 123

**🔤 String Methods**

| **Method** | **What it does** |
| --- | --- |
| length | Number of characters |
| toUpperCase() / toLowerCase() | Changes case |
| trim() | Removes whitespace |
| includes(sub) | Checks if string contains substring |
| indexOf(sub) | Returns index of first match |
| replace(old, new) | Replaces text |
| split(sep) | Converts string to array |
| slice(start, end) | Cuts string part |
| charAt(i) | Returns character at index |
| startsWith() / endsWith() | Self-explanatory |
| repeat(n) | Repeats the string n times |

**📌 Example:**

"hello".toUpperCase(); // "HELLO"

"hello world".includes("world"); // true

**✅ Boolean Functions**

| **Function** | **What it does** |
| --- | --- |
| Boolean(val) | Converts any value to true/false |
| !!val | Shortcut for Boolean conversion |
| true.toString() | "true" |
| false.valueOf() | false |

**📌 Example:**

Boolean(0); // false

!!"hello"; // true

**📦 Array Methods**

| **Method** | **What it does** |
| --- | --- |
| push(val) | Add to end |
| pop() | Remove from end |
| shift() | Remove from start |
| unshift(val) | Add to start |
| length | Number of items |
| slice(start, end) | Gets a portion |
| splice() | Add/remove items |
| indexOf(val) | Get index |
| includes(val) | Check if exists |
| join(sep) | Convert to string |
| map(fn) | Create new array from values |
| filter(fn) | Get items matching a condition |
| reduce(fn) | Reduce to a single value |
| forEach(fn) | Loop through |
| sort() | Sorts elements (can be customized) |
| find(fn) | Finds first match |

**📌 Example:**

[1, 2, 3].map(x => x \* 2); // [2, 4, 6]

**🧱 Object Methods**

| **Method** | **What it does** |
| --- | --- |
| Object.keys(obj) | Array of property names |
| Object.values(obj) | Array of values |
| Object.entries(obj) | Array of [key, value] pairs |
| Object.assign(target, source) | Copies properties |
| hasOwnProperty(key) | Checks if key exists |
| delete obj.key | Deletes property |
| JSON.stringify(obj) | Converts object to JSON string |
| JSON.parse(jsonStr) | Converts JSON string to object |

**📌 Example:**

const user = { name: "Krishna", age: 17 };

Object.keys(user); // ["name", "age"]

**🔄 Type Conversion Functions (Global)**

| **Function** | **Converts To** |
| --- | --- |
| String(val) | String |
| Number(val) | Number |
| Boolean(val) | Boolean |
| parseInt(val) | Integer |
| parseFloat(val) | Decimal |
| typeof | Tells data type |
| instanceof | Checks constructor |

**🔍 Others**

* typeof → check the type (typeof "hello" → "string")
* isNaN() → checks if value is Not-a-Number
* Array.isArray(val) → check if it's an array
* new Date() → creates date object

Would you like a **PDF note version**, or **interactive flashcards** of these?