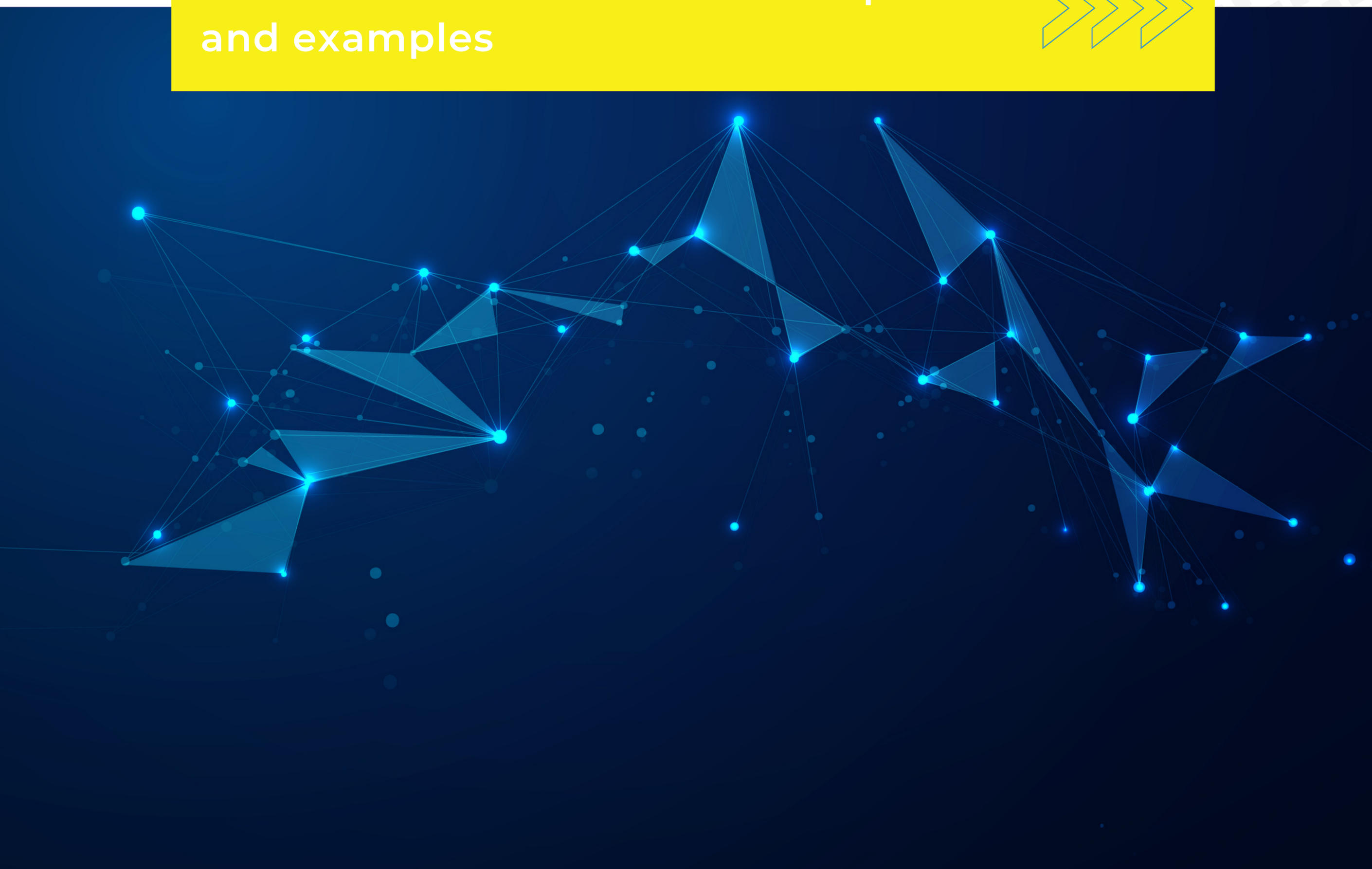


2024

JAVASCRIPT DATE FUNCTIONS

Your Ultimate Guide!

Date methods with clear descriptions
and examples



Mastering JavaScript Date Functions: Your Ultimate Guide!

JavaScript provides a powerful `Date` object for handling dates and times. Whether you're building complex applications or just need to manipulate dates and times, understanding these functions is crucial. Let's dive into all the Date methods with clear descriptions and examples!

1. Creating Dates

1.1 `new Date()`

- Creates a new date object with the current date and time.
- Example:

```
const now = new Date();  
  
console.log(now); // Outputs current date and time
```

1.2 `new Date(year, month, day, hours, minutes, seconds, milliseconds)`

- Creates a new date object with specified date and time.
- Example:

```
const specificDate = new Date(2024, 8, 6, 15, 30, 0, 0);  
  
console.log(specificDate); // Outputs: Tue Aug 06 2024 15:30:00 GMT+0000 (UTC)
```

1.3 `new Date(milliseconds)`

- Creates a new date object as milliseconds since January 1, 1970.
- Example:

```
const epochDate = new Date(0);  
  
console.log(epochDate); // Outputs: Thu Jan 01 1970 00:00:00 GMT+0000 (UTC)
```

1.4 `new Date(dateString)`

- Creates a new date object from a date string.

- Example:

```
const dateString = new Date("2024-08-06T15:30:00");  
console.log(dateString); // Outputs: Tue Aug 06 2024 15:30:00 GMT+0000 (UTC)
```

2. Getting Date Components

2.1 `getDate()`

- Returns the day of the month (1-31).

- Example:

```
const day = now.getDate();  
console.log(day); // Outputs: 6
```

2.2 `getDay()`

- Returns the day of the week (0-6).

- Example:

```
const weekday = now.getDay();  
console.log(weekday); // Outputs: 2 (Tuesday)
```

2.3 `getFullYear()`

- Returns the year (4 digits).

- Example:

```
const year = now.getFullYear();  
console.log(year); // Outputs: 2024
```

2.4 `getMonth()`

- Returns the month (0-11).

- Example:

```
const month = now.getMonth();  
console.log(month); // Outputs: 7 (August)
```

2.5 ``getHours()``

- Returns the hour (0-23).

- Example:

```
const hours = now.getHours();  
console.log(hours); // Outputs: 15
```

2.6 ``getMinutes()``

- Returns the minutes (0-59).

- Example:

```
const minutes = now.getMinutes();  
console.log(minutes); // Outputs: 30
```

2.7 ``getSeconds()``

- Returns the seconds (0-59).

- Example:

```
const seconds = now.getSeconds();  
console.log(seconds); // Outputs: 0
```

2.8 ``getMilliseconds()``

- Returns the milliseconds (0-999).

- Example:

```
const milliseconds = now.getMilliseconds();
```

```
console.log(milliseconds); // Outputs: 0
```

2.9 `getTime()`

- Returns the number of milliseconds since January 1, 1970.

- Example:

```
const time = now.getTime();  
console.log(time); // Outputs: 1725551400000
```

2.10 `getTimezoneOffset()`

- Returns the time difference between UTC and local time, in minutes.

- Example:

```
const timezoneOffset = now.getTimezoneOffset();  
console.log(timezoneOffset); // Outputs: -240 (for UTC-4)
```

2.11 `getUTCDate()`

- Returns the day of the month (1-31) in UTC.

- Example:

```
const utcDay = now.getUTCDate();  
console.log(utcDay); // Outputs: 6
```

2.12 `getUTCDay()`

- Returns the day of the week (0-6) in UTC.

- Example:

```
const utcWeekday = now.getUTCDay();  
console.log(utcWeekday); // Outputs: 2 (Tuesday)
```

2.13 `getUTCFullYear()`

- Returns the year (4 digits) in UTC.

- Example:

```
const utcYear = now.getUTCFullYear();
```

```
console.log(utcYear); // Outputs: 2024
```

2.14 `getUTCMonth()`

- Returns the month (0-11) in UTC.

- Example:

```
const utcMonth = now.getUTCMonth();
```

```
console.log(utcMonth); // Outputs: 7 (August)
```

2.15 `getUTCHours()`

- Returns the hour (0-23) in UTC.

- Example:

```
const utcHours = now.getUTCHours();
```

```
console.log(utcHours); // Outputs: 19 (UTC time)
```

2.16 `getUTCMinutes()`

- Returns the minutes (0-59) in UTC.

- Example:

```
const utcMinutes = now.getUTCMinutes();
```

```
console.log(utcMinutes); // Outputs: 30
```

2.17 `getUTCSeconds()`

- Returns the seconds (0-59) in UTC.

- Example:

```
const utcSeconds = now.getUTCSeconds();  
console.log(utcSeconds); // Outputs: 0
```

2.18 `getUTCMilliseconds()`

- Returns the milliseconds (0-999) in UTC.
- Example:

```
const utcMilliseconds = now.getUTCMilliseconds();  
console.log(utcMilliseconds); // Outputs: 0
```

3. Setting Date Components

3.1 `setDate(day)`

- Sets the day of the month (1-31).
- Example:

```
now.setDate(15);  
console.log(now); // Date set to the 15th
```

3.2 `setFullYear(year, month, day)`

- Sets the year, and optionally the month and day.
- Example:

```
now.setFullYear(2025, 8, 20);  
console.log(now); // Year set to 2025, month to September, day to 20th
```

3.3 `setHours(hours, minutes, seconds, milliseconds)`

- Sets the hour, and optionally the minutes, seconds, and milliseconds.
- Example:

```
now.setHours(10, 45, 30, 0);  
console.log(now); // Time set to 10:45:30
```

3.3 `setMilliseconds(milliseconds)`

- Sets the milliseconds (0-999).
- Example:

```
now.setMilliseconds(500);  
console.log(now); // Milliseconds set to 500
```

3.4 `setMinutes(minutes, seconds, milliseconds)`

- Sets the minutes, and optionally the seconds and milliseconds.
- Example:

```
now.setMinutes(50, 45, 0);  
console.log(now); // Minutes set to 50, seconds to 45
```

3.5 `setMonth(month, day)`

- Sets the month (0-11), and optionally the day.
- Example:

```
now.setMonth(5, 25);  
console.log(now); // Month set to June, day to 25th
```

3.6 `setSeconds(seconds, milliseconds)`

- Sets the seconds, and optionally the milliseconds.
- Example:

```
now.setSeconds(30, 500);  
console.log(now); // Seconds set to 30, milliseconds to 500
```


3.7 `setTime(milliseconds)`

- Sets the time in milliseconds since January 1, 1970.

- Example:

```
now.setTime(1625567400000);  
console.log(now); // Date set to specific milliseconds
```

3.8 `setUTCDate(day)`

- Sets the day of the month (1-31) in UTC.

- Example:

```
now.setUTCDate(20);  
console.log(now); // UTC day set to 20th
```

3.9 `setUTCFullYear(year, month, day)`

- Sets the year, and optionally the month and day, in UTC.

- Example:

```
now.setUTCFullYear(2025, 8, 20);  
console.log(now); // UTC year set to 2025, month to September, day to 20th
```

3.10 `setUTCHours(hours, minutes, seconds, milliseconds)`

- Sets the hour, and optionally the minutes, seconds, and milliseconds, in UTC.

- Example:

```
now.setUTCHours(10, 45, 30, 0);  
console.log(now); // UTC time set to 10:45:30
```

3.11 `setUTCMilliseconds(milliseconds)`

- Sets the milliseconds (0-999) in UTC.

- Example:

```
now.setUTCMilliseconds(500);  
  
console.log(now); // UTC milliseconds set to 500
```

3.12 `setUTCMinutes(minutes, seconds, milliseconds)`

- Sets the minutes, and optionally the seconds and milliseconds, in UTC.

- Example:

```
now.setUTCMinutes(50, 45, 0);  
  
console.log(now); // UTC minutes set to 50, seconds to 45
```

3.13 `setUTCMonth(month, day)`

- Sets the month (0-11), and optionally the day, in UTC.

- Example:

```
now.setUTCMonth(5, 25);  
  
console.log(now); // UTC month set to June, day to 25th
```

3.14 `setUTCSeconds(seconds, milliseconds)`

- Sets the seconds, and optionally the milliseconds, in UTC.

- Example:

```
now.setUTCSeconds(30, 500);  
  
console.log(now); // UTC seconds set to 30, milliseconds to 500
```

4. Parsing and Formatting Dates

4.1 `Date.parse(dateString)`

- Parses a date string and returns the number of milliseconds since January 1, 1970.

- Example:

```
const ms = Date.parse("2024-08-06T15:30:00");  
console.log(ms); // Outputs: 1725551400000
```

4.2 `Date.UTC(year, month, day, hours, minutes, seconds, milliseconds)`

- Returns the number of milliseconds since January 1, 1970, in UTC.

- Example:

```
const utcMs = Date.UTC(2024, 7, 6, 15, 30, 0, 0);  
console.log(utcMs); // Outputs: 1725551400000
```

4.3 `toDateStr()`

- Returns the date portion of a Date object as a string.

- Example:

```
const dateStr = now.toDateString();  
console.log(dateStr); // Outputs: Tue Aug 06 2024
```

4.4 `toISOString()`

- Returns the date as a string in ISO format.

- Example:

```
const isoStr = now.toISOString();  
console.log(isoStr); // Outputs: 2024-08-06T15:30:00.000Z
```

4.5 `toJSON()`

- Returns the date as a string in JSON format.

- Example:

```
const jsonStr = now.toJSON();  
console.log(jsonStr); // Outputs: 2024-08-06T15:30:00.000Z
```

4.6 `toLocaleDateString(locales, options)`

- Returns the date portion of a Date object as a string, using locale-specific formatting.

- Example:

```
const localeDateStr = now.toLocaleDateString('en-US', { weekday: 'long',  
year: 'numeric', month: 'long', day: 'numeric' });  
console.log(localeDateStr); // Outputs: Tuesday, August 6, 2024
```

4.7 `toLocaleString(locales, options)`

- Returns the date and time as a string, using locale-specific formatting.

- Example:

```
const localeStr = now.toLocaleString('en-US', { hour12: true });  
console.log(localeStr); // Outputs: 8/6/2024, 3:30:00 PM
```

4.8 `toLocaleTimeString(locales, options)`

- Returns the time portion of a Date object as a string, using locale-specific formatting.

- Example:

```
const localeTimeStr = now.toLocaleTimeString('en-US', { hour: '2-digit',  
minute: '2-digit' });  
console.log(localeTimeStr); // Outputs: 03:30 PM
```

4.9 `toString()`

- Returns the date as a string.
- Example:

```
const dateStr = now.toString();  
console.log(dateStr); // Outputs: Tue Aug 06 2024 15:30:00 GMT+0000 (UTC)
```

4.10 `toTimeString()`

- Returns the time portion of a Date object as a string.
- Example:

```
const timeStr = now.toTimeString();  
console.log(timeStr); // Outputs: 15:30:00 GMT+0000 (UTC)
```

4.11 `toUTCString()`

- Returns the date as a string, using the UTC time zone.
- Example:

```
const utcStr = now.toUTCString();  
console.log(utcStr); // Outputs: Tue, 06 Aug 2024 15:30:00 GMT
```

Understanding and utilizing these JavaScript Date functions will enhance your ability to manage dates and times effectively in your projects. Which method did you find most interesting? Share your thoughts and experiences below! 📌