# Chrome DevTools Quick Reference

Elements, Console, Network, Performance, debugging

## Elements

### Inspecting & Editing

| | |
|---|---|
| **Right-click -> Inspect** | Open Elements panel for element |
| **Double-click tag/attribute** | Edit HTML inline |
| **Delete key** | Delete selected node |
| **Ctrl+Z** | Undo DOM change |
| **H key** | Toggle visibility of selected element |
| **Drag node** | Move element in DOM tree |

### Styles Panel

| | |
|---|---|
| **element.style {}** | Add inline styles to element |
| **Click property value** | Edit CSS value live |
| **Checkbox next to rule** | Toggle CSS property on/off |
| **:hov button** | Force element pseudo-states (:hover, :focus) |
| **.cls button** | Add/remove CSS classes |
| **Color swatch** | Open color picker |
| **Computed tab** | View final computed CSS values |

## Console

### Console API

```
console.log("info");          console.warn("warning");
console.error("error");       console.table(arrayOrObj);
console.group("label");       console.groupEnd();
console.time("t"); /*...*/ console.timeEnd("t");
```

### Console Utilities

| | |
|---|---|
| **$0** | Currently selected element in Elements |
| **$('sel') / $$('sel')** | querySelector / querySelectorAll shorthand |
| **copy(obj)** | Copy object as string to clipboard |
| **clear()** | Clear console output |
| **monitor(fn)** | Log calls to function fn |
| **monitorEvents(el, 'click')** | Log events on element |
| **keys(obj) / values(obj)** | Object keys / values |
| **$_** | Result of last evaluated expression |

### Filtering

| | |
|---|---|
| **Log levels dropdown** | Filter by verbose/info/warn/error |
| **Filter text box** | Search console output |
| **-url:extension** | Exclude messages by source URL |
| **context: dropdown** | Filter by iframe/worker context |

## Network

### Network Panel Features

| | |
|---|---|
| **Filter bar** | Filter by type: XHR, JS, CSS, Img, Doc, WS |
| **Search (Ctrl+F)** | Search across all request/response bodies |
| **Preserve log** | Keep log across page navigation |
| **Disable cache** | Bypass browser cache while DevTools open |
| **Throttling dropdown** | Simulate Slow 3G, Fast 3G, Offline |
| **Block request URL** | Right-click request -> Block URL |

### Request Details Tabs

| | |
|---|---|
| **Headers** | Request/response headers, status code |
| **Payload** | POST body, query parameters |
| **Preview** | Formatted response (JSON, HTML, image) |
| **Response** | Raw response body |
| **Timing** | DNS, connect, TLS, TTFB, download |
| **Initiator** | Stack trace that triggered request |
| **Cookies** | Cookies sent/received |

### Copy & Export

| | |
|---|---|
| **Right-click -> Copy as cURL** | Copy request as cURL command |
| **Copy as fetch** | Copy as fetch() JavaScript |
| **Export HAR** | Export all requests as HAR file |
| **Copy response** | Copy response body to clipboard |

## Sources

### Breakpoints

| | |
|---|---|
| **Click line number** | Toggle line breakpoint |
| **Right-click -> Conditional** | Break only when expression is true |
| **Right-click -> Logpoint** | Log without pausing execution |
| **DOM breakpoint** | Break on subtree/attribute/removal |
| **XHR/Fetch breakpoint** | Break when URL contains string |
| **Event listener breakpoint** | Break on specific event types |

### Debugger Controls

| | |
|---|---|
| **F8 / Ctrl+** | Resume / Pause execution |
| **F10** | Step over next function call |
| **F11** | Step into function call |
| **Shift+F11** | Step out of current function |
| **Ctrl+Shift+P -> "never pause"** | Disable all breakpoints |

### Debug Panels

| | |
|---|---|
| **Watch** | Monitor expression values |
| **Scope** | Local, closure, global variables |
| **Call Stack** | Function call chain |
| **Snippets** | Save and run reusable JS code |

## Performance

### Recording

| | |
|---|---|
| **Record button (Ctrl+E)** | Start/stop recording performance |
| **Reload button** | Record page load performance |
| **Screenshots checkbox** | Capture visual timeline |
| **CPU throttle dropdown** | Simulate 4x/6x CPU slowdown |
| **Network throttle** | Simulate slow network during recording |

### Reading the Flame Chart

| | |
|---|---|
| **Main track** | JavaScript execution (flame chart) |
| **Network track** | Network requests timeline |
| **Frames track** | FPS and frame durations |
| **Timings track** | FCP, LCP, DCL, Load markers |
| **Yellow bars** | JavaScript (scripting) |
| **Purple bars** | Layout / rendering |
| **Green bars** | Painting / compositing |

### Bottom-Up & Summary

| | |
|---|---|
| **Summary tab** | Time breakdown: scripting, rendering, etc. |
| **Bottom-Up tab** | Most expensive functions first |
| **Call Tree tab** | Root-to-leaf call hierarchy |
| **Event Log tab** | Chronological event list |

## Application

### Storage

| | |
|---|---|
| **Local Storage** | View/edit key-value pairs per origin |
| **Session Storage** | View/edit session-scoped storage |
| **IndexedDB** | Inspect object stores and records |
| **Cookies** | View/edit/delete cookies per domain |
| **Cache Storage** | Inspect Service Worker caches |
| **Clear storage** | Bulk clear selected storage types |

### Service Workers & Manifest

| | |
|---|---|
| **Service Workers** | View registration, status, push/sync |
| **Update on reload** | Force SW update on each reload |
| **Bypass for network** | Skip SW and go to network |
| **Manifest** | View web app manifest details |
| **Offline checkbox** | Simulate offline mode |

## Lighthouse

### Running Audits

| | |
|---|---|
| **Mode: Navigation** | Full page load audit |
| **Mode: Timespan** | Audit user interactions over time |
| **Mode: Snapshot** | Audit current page state |
| **Categories** | Performance, Accessibility, Best Practices, SEO |
| **Device** | Mobile or Desktop simulation |

### Key Metrics

| | |
|---|---|
| **FCP (First Contentful Paint)** | Time to first visible content |
| **LCP (Largest Contentful Paint)** | Time to largest visible element |
| **TBT (Total Blocking Time)** | Sum of long task blocking time |
| **CLS (Cumulative Layout Shift)** | Visual stability score |
| **SI (Speed Index)** | How quickly content is visually populated |
| **INP (Interaction to Next Paint)** | Responsiveness to user input |

## Shortcuts

### Opening DevTools

| | |
|---|---|
| **F12 / Ctrl+Shift+I** | Toggle DevTools open/close |
| **Ctrl+Shift+J** | Open Console panel |
| **Ctrl+Shift+C** | Open Elements + inspect mode |
| **Ctrl+Shift+M** | Toggle device toolbar (responsive) |

### Navigation & Search

| | |
|---|---|
| **Ctrl+Shift+P** | Command Menu (run any action) |
| **Ctrl+P** | Open file (Go to File) |
| **Ctrl+Shift+F** | Search across all sources |
| **Ctrl+G** | Go to line number in Sources |
| **Ctrl+[ / Ctrl+]** | Switch between panels |

### Editing & Console

| | |
|---|---|
| **Ctrl+Shift+D** | Toggle DevTools dock position |
| **Ctrl+L (Console)** | Clear console output |
| **Shift+Enter (Console)** | Multi-line input |
| **Esc** | Toggle console drawer |
| **Ctrl+K (Console)** | Clear console |

## Mobile Debug

### Device Toolbar

| | |
|---|---|
| **Ctrl+Shift+M** | Toggle device toolbar |
| **Device dropdown** | Preset phone/tablet dimensions |
| **Responsive mode** | Freely resize viewport |
| **DPR dropdown** | Change device pixel ratio |
| **Throttling** | Simulate mobile CPU + network |
| **Show media queries** | Visualize CSS breakpoints |

### Remote Debugging (Android)

| | |
|---|---|
| **1. Enable USB debugging** | Settings -> Developer Options on device |
| **2. Connect USB** | Connect Android device to computer |
| **3. chrome://inspect** | Open in desktop Chrome |
| **4. Click Inspect** | Open DevTools for mobile page |

Requires Chrome on both desktop and Android device

# Chrome DevTools Quick Reference

### Sensors Override

| | |
|---|---|
| **`Geolocation`** | Override GPS coordinates |
| **`Orientation`** | Simulate device orientation |
| **`Touch`** | Simulate touch events |
| **`Idle detection`** | Override idle detection API |

## Common Patterns

### Debug Network Issues

```
// In Console: monitor all fetch requests
const origFetch = window.fetch;
window.fetch = (...args) => {
  console.log('fetch:', args);
  return origFetch(...args);
};
```

### Performance Workflow

| | |
|---|---|
| **`1. Lighthouse audit`** | Identify top performance issues |
| **`2. Performance recording`** | Find long tasks in flame chart |
| **`3. Coverage tab`** | Find unused CSS/JS (Ctrl+Shift+P -> Coverage) |
| **`4. Network waterfall`** | Identify blocking resources |
| **`5. Rendering tab`** | Visualize paint/layout (Ctrl+Shift+P -> Rendering) |

### Useful Console Snippets

```
// List all event listeners on element
getEventListeners($0);

// Monitor layout shifts
new PerformanceObserver(l => l.getEntries().forEach(
  e => console.log('CLS:', e)
)).observe({ type: 'layout-shift', buffered: true });
```