

JSON Syntax & jq Terminal Query Cheat Sheet

RFC 8259 JSON syntax specifications, formatting gotchas, and essential jq command line recipes.

Live Interactive JSON Formatter & Validator: <https://lotsofnetwork.com/json-formatter>

Data Type	Valid Example	RFC 8259 Strict Syntax Constraint
String	"valid string"	Must use double quotes ("). Single quotes ('invalid') cause syntax error
Number	42, -17, 3.1415	Integers and floats allowed. Hexadecimal (0xFF) and leading zeros (01) invalid
Boolean	true false	Strictly lowercase true or false. True, False, TRUE are invalid in JSON
Null	null	Strictly lowercase null. nil, None, and undefined are invalid syntax
Array	["a", 1, false]	Ordered elements. Trailing commas ([1, 2,]) are strictly forbidden
Object	{"key": "val"}	Key-value map. Keys MUST be double-quoted strings ({key: 1} is invalid)

Essential jq Command Line Query Recipes

Query Objective	Terminal Command Syntax
Pretty-print formatted JSON	<code>curl -s https://api.com jq .</code>
Extract single top-level field	<code>jq '.status'</code>
Unquote string for Bash variables	<code>TOKEN=\$(jq -r '.access_token' res.json)</code>
Iterate over array of objects	<code>jq '.users[] {id, email}'</code>
Filter array by attribute condition	<code>jq '[] select(.role == "admin")'</code>
Map attribute across entire array	<code>jq '[] .items[] .price'</code>
Get total length of array or object	<code>jq '.users length'</code>
List all top-level keys in object	<code>jq 'keys'</code>