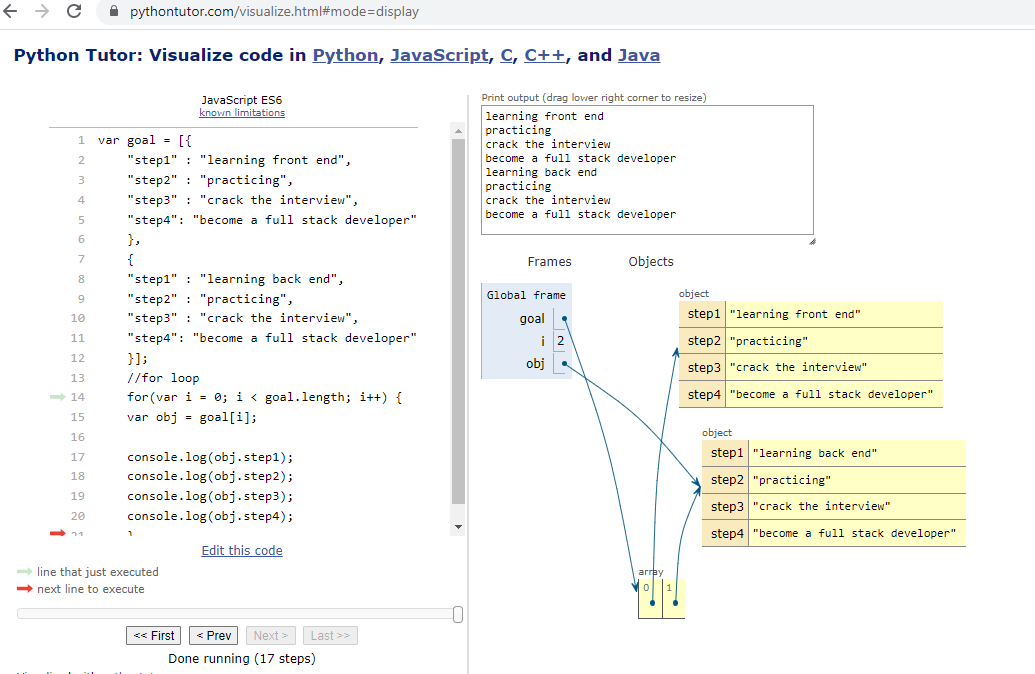
**JSON iterate over all for loops( for, for in)**

**FOR LOOP:**

|  |  |
| --- | --- |
| 1 | var goal = [{ |
| 2 | "step1" : "learning front end", |
| 3 | "step2" : "practicing", |
| 4 | "step3" : "crack the interview", |
| 5 | "step4": "become a full stack developer" |
| 6 | }, |
| 7 | { |
| 8 | "step1" : "learning back end", |
| 9 | "step2" : "practicing", |
| 10 | "step3" : "crack the interview", |
| 11 | "step4": "become a full stack developer" |
| 12 | }]; |
| 13 | //for loop |
| 14 | for(var i = 0; i < goal.length; i++) { |
| 15 | var obj = goal[i]; |
| 16 |  |
| 17 | console.log(obj.step1); |
| 18 | console.log(obj.step2); |
| 19 | console.log(obj.step3); |
| 20 | console.log(obj.step4); |
| 21 | } |



**FOR IN LOOP:**

var goal = [{

"step1" : "learning front end",

"step2" : "practicing",

"step3" : "crack the interview",

"step4": "become a full stack developer"

},

{

"step1" : "learning back end",

"step2" : "practicing",

"step3" : "crack the interview",

"step4": "become a full stack developer"

}];

//for in loop

for (var i in goal) {

var obj = goal[i];

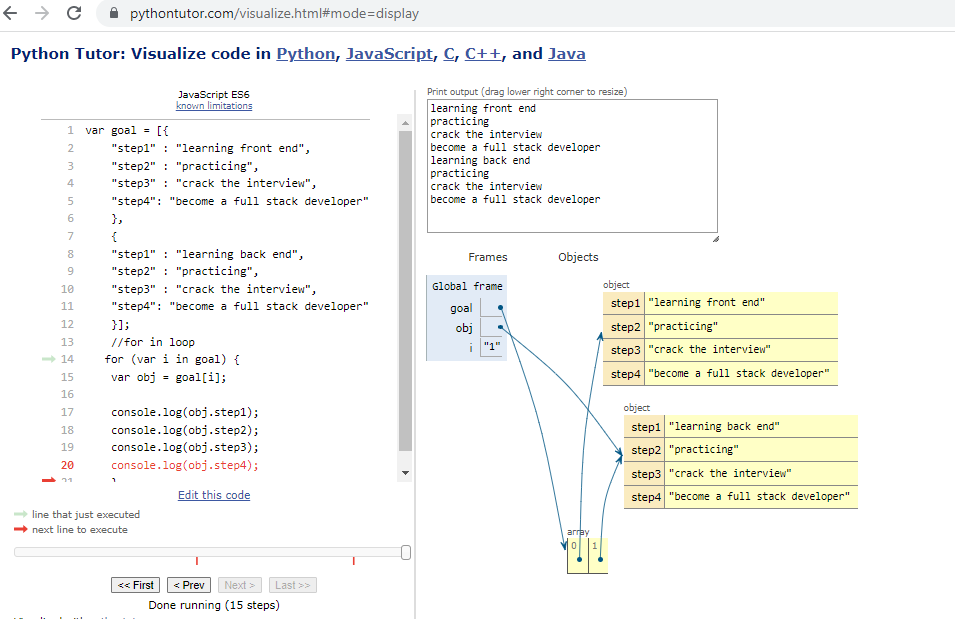
console.log(obj.step1);

console.log(obj.step2);

console.log(obj.step3);

console.log(obj.step4);

}



**Create your own resume data in JSON format**

let myResume={

    "content": {

    "name": "M PADMAPRIYA",

    "email": "mpadmapriya96@gamil.com",

    "phone": 9952473429,

    "degree": "B.E",

    "department": "Electronics and communication engineering"

    },

    "location":

        {

    "address": "M185 master manickam street",

    "city": "kanchipuram",

    "state": "Tamilnadu",

    "country": "India"

    },

    "work experience": [

    {

    "company 1": "TSMT Technology India pvt ltd",

    "position": "Team Leader",

    "duration": "2017-2020",

    "Role": "Team leader of production department in JIO, NOKIA, MI TV assembly and testing process ",

    "company 2": "Salcomp Manufacturing India pvt ltd",

    "position": "Line manager",

    "duration": "2021-2021",

    "Role": "Line manager of Material and scrap process ",

    "company 3": "Ravel electronics pvt ltd",

    "position": "production Engineer",

    "duration": "2022-2022",

    "Role": "production engineer of assembly, SMT, packing process ",

    },

    ],

    "education": [

    {

    "institution": "SKR engineering college",

    "department": "Electronics and communication engineering",

    "duration": "2013-2017",

    "gpa": 7.0,

    }

    ],

    "skills": [

    {

    "name": "javascript",

    "level": "beginer",

    ]

    }

    ],

    "languages": [

    {

    "language": "Tamil,Enlish",

    }

    ],

    }

    console.log(myResume);

**JavaScript: window, document and screen**

**Window**

The JavaScript window object sits at the top of the JavaScript Object hierarchy and represents the browser window. The window object is supported by all browsers. All global JavaScript objects , functions, and variables automatically become members of the window object. The window is the first thing that gets loaded into the browser . This window object has the majority of the properties like length, innerWidth, innerHeight, name, if it has been closed, its parents, and more.

The window object represents the current browsing context . It holds things like window.location, window.history, window.screen, window.status, or the window.document . Each browser tab has its own top-level window object. Each of these windows gets its own separate global object. window.window always refers to window, but window.parent and window.top might refer to enclosing windows, giving access to other execution contexts. Moreover, the window property of a window object points to the window object itself. So the following ststements all return the same window object:

window.window

window.window.window

window.window.window.window

**Document**

The Document interface represents any web page loaded in the browser and serves as an entry point into the web page's content, which is the DOM tree. When an HTML document is loaded into a web browser , it becomes a document object. It is the root node of the HTML document. The document actually gets loaded inside the window object and has properties available to it like title, URL, cookie, etc. HTML documents, served with the "text/html" content type, also implement the HTMLDocument interface, whereas XML and SVG documents implement the XMLDocument interface.

**Screen**

Screen is a small information object about physical screen dimensions . It can be used to display screen width, height, colorDepth, pixelDepth etc. It is not mandatory to write window prefix with screen object. It can be written without window prefix.

**Properties:**

screen.width

screen.height

screen.availWidth

screen.availHeight

screen.colorDepth

screen.pixelDepth