

Assume you are expert in javascript programming and a good guide to explain in form of comments to the beginner Providing codes from beginner to advance where beginner will be fully satisfied by providing all the content in one shot for the topic "Taking inputs in a single or multi line with different datatypes creatively arranged" only provide javascript code

Sure! Here's a **JavaScript code file** that **explains how to take inputs** in **single-line and multi-line formats** with **different datatypes**, using creative methods — like `prompt()`, destructuring, `split()`, type conversions, and `readline` for advanced usage.

```
// =====  
// 📌 JavaScript Input Handling: Beginner to Advanced  
// =====  
  
// -----  
// ✅ Basic Input from Single Prompt (String)  
// -----  
// Using prompt() to get a simple string input  
let name = prompt("Enter your name:");  
console.log("Hello, " + name + "!");  
  
// -----  
// ✅ Input and Convert to Number  
// -----  
let age = parseInt(prompt("Enter your age:")); // Converts string to integer  
console.log("You are " + age + " years old.");  
  
// -----  
// ✅ Taking Multiple Inputs in a Single Line (as Strings)  
// -----  
let fullInput = prompt("Enter your city and country (e.g., Hyderabad, India):");  
let [city, country] = fullInput.split(","); // Destructuring after splitting  
console.log("You live in " + city.trim() + ", " + country.trim() + ".");
```

```
// -----
// ✅ Taking Multiple Inputs with Different DataTypes
// -----
let rawInput = prompt("Enter your score (number), isPassed (true/false), and grade (char)
separated by spaces:");
let [scoreStr, isPassedStr, grade] = rawInput.split(" ");
let score = Number(scoreStr);
let isPassed = (isPassedStr === "true");
console.log(`Score: ${score}, Passed: ${isPassed}, Grade: ${grade}`);

// -----
// ✅ Advanced: Taking Multi-Line Inputs in Node.js (Terminal)
// -----
// If you're running JavaScript in Node.js, you can use the readline module

// Uncomment the following block if using Node.js
/*
const readline = require('readline');
const rl = readline.createInterface({
  input: process.stdin,
  output: process.stdout
});

let inputs = [];

console.log("Enter three values (string, number, boolean) one by one:");

rl.on('line', (input) => {
  inputs.push(input);

  if (inputs.length === 3) {
    let strVal = inputs[0];
    let numVal = parseFloat(inputs[1]);
    let boolVal = inputs[2] === 'true';

    console.log("Collected Values:");
    console.log("String:", strVal);
    console.log("Number:", numVal);
    console.log("Boolean:", boolVal);
    rl.close();
  }
});
*/
```

```
// -----
// ✅ Creative Input Example: CSV Format Parsing
// -----
let csvInput = prompt("Enter a list of numbers separated by commas:");
let numberList = csvInput.split(",").map(num => parseFloat(num));
console.log("Parsed Numbers:", numberList);

// -----
// ✅ JSON Input Parsing (for advanced structured input)
// -----
let jsonString = prompt('Enter user data as JSON: {"name": "John", "age": 30}');
let userData = JSON.parse(jsonString);
console.log("User Info:", userData);

// =====
// 🎯 Summary
// - prompt(): Used in browser for basic inputs.
// - split(): Breaks input string into parts.
// - parseInt/parseFloat: Converts string to number.
// - JSON.parse(): Parses structured input.
// - readline (Node.js): For multi-line terminal input.
// =====
```

✅ **Copy-paste this into your browser's DevTools (Console tab)** for browser prompts or use **Node.js** for the advanced part with **readline**.

Let me know if you want the same thing in the browser GUI format or with form elements!