

pjs.display()

Displays HTML content and waits for the user to respond. Once displayed, the screen can be submitted back to the server with the use of the `pui.submit()` client-side browser API.

The content is defined using an [EJS Template](#). The template is populated using the strongly typed fields declared within the global scope of the Node.js module or using the data parameter. Since Profound.js applications are single page applications, the content should not be a full HTML document and must not include `<html>`, `<title>`, and `<body>` tags. It should only include the body content to render. The browser page will not be reloaded.

Parameters

- **HTML Template File**

This parameter is the file path to the **HTML Template File** relative to the `modules` directory, specified as a string. If the path is not qualified, the [pathlist](#) is used to search for the **HTML Template File**.

Alternatively, this parameter can also be specified as an object with one of the following properties:

- `file` - HTML Template File
- `html` - the actual HTML string to use for the HTML template

- **Data** (*optional*)

This parameter is the data object to pass to the template. If omitted, all strongly typed fields declared within the global scope are passed to the template.

- **EJS options** (*optional*)

This parameter is a JavaScript object specifying EJS options as documented on www.ejs.co.

Return Value

The user's response to the screen is returned as a JavaScript object representing the browser's post data.

RPG Equivalent

EXFMT

Examples

HTML Template code

```
<h1> Calculate Square of a Number </h1>

<br/>
Enter a number: <input type="text" name="number" value="<%= number %>" />

<br/>
The square is: <%= square %>

<input type="button" value="Calculate" onclick="pui.submit()" />
<input type="button" value="Exit" onclick="pui.submit('exit')" />
```

Automatically passing declared fields to the template

```
pjs.define("number", { type: "integer" });
pjs.define("square", { type: "integer" });
pjs.define("exit", { type: "boolean" });

while (!exit) {
  pjs.display("calculate_square.html");
  square = number * number;
}
```

Passing data parameter to the template

```
pjs.define("number", { type: "integer" });
pjs.define("square", { type: "integer" });
pjs.define("exit", { type: "boolean" });

while (!exit) {
  pjs.display("calculate_square.html", { number: number, square: square });
  square = number * number;
}
```

Video Tutorial