Introduction

The article is important because the MVC architecture previously was only available on the server side scripts, we develop the client side scripting with minimal requirements and reliable performance.

Background

To the present time the controller API are used by many programming toolkits like ASP .NET, JSP or Spring Boot, which reside on server and, thus, aren't as flexible as the client side routines implemented in IdahoJS framework.

Using the code

To use the code we first have to instantiate the controllers and optionally persistent components in the runtime of client side:

```
<!DOCTYPE HTML>
<html>
    <head>
        <script src="idaho.js"></script>

        <script>
            idahoControllers.push ({
                path: "index_lang.html",
                args: [ "lang_en_us" ], /* optional */
                html_selector: "#doc_element",
                controller: function (args) { }
            });

            idahoControllers.push ({
                path: "index_lang.html",
                args: [ "lang_en_uk" ], /* optional */
                html_selector: "#doc_element_uk",
                controller: function (args) { }
            });

            idahoObjects.push ({
                html_selector: "#doc_element",
                url: "api.php", /* optional */
                get value() { return idahoGetValue(this); },
                set value(v) { idahoSetValue(this, v); }
            });
        </script>
    </head>
    <body>
        <div id="doc_element">
            Text in English (US)
        </div>
```

```
        <div id="doc_element_uk">
            Text in English (UK)
        </div>

        <script>
            idahoInitialize();
        </script>
    </body>
</html>
```

From the code above `idahoControllers` and `idahoObjects` are controller and persistent object descriptors which define the type of behavior according to the browser URL and the URL for the API to store and retrieve the value from the database or other storages.

Points of Interest

We have learned the way of implementing client side controllers within the example of multi-lingual page or even more.

History

29th May 2023 - Initial release.