

```
1 <template>
2   <div>
3     <div>
4       <v-card class="mx-auto" max-width="344" outlined>
5         <v-card-title>DynamoDb tutorial</v-card-title>
6         <v-card-actions>
7           <div class="d-flex flex-column justify-center">
8             <v-btn color="primary" @click="addItem()">Add item to table</v-btn>
9             <br />
10            <v-btn color="primary" @click="deleteItem()">Delete item</v-btn>
11            <br />
12            <v-btn color="primary" @click="getItem()">Get item out table</v-btn>
13            <br />
14            <v-btn color="primary" @click="updateItem()">Update item</v-btn>
15          </div>
16        </v-card-actions>
17      </v-card>
18    </div>
19  </div>
20 </template>
21
22 <script>
23 const AWS = require("aws-sdk");
24 // Set the region where your identity pool exists
25 AWS.config.region = "eu-west-1";
26
27 // Configure the credentials provider to use your identity pool
28 AWS.config.credentials = new AWS.CognitoIdentityCredentials({
29   IdentityPoolId: "eu-west-1:c413ea15-7a3c-4f9d-b133-afc5b06ccaae"
30 });
31
32 export default {
33   data() {
34     return {
35       credentials: null
36     };
37   },
38   computed: {},
39   created() {
40     if (this.credentials) {
41       // have credentials
42       console.log("in created(), have credentials");
43     } else {
44       // no credentials
45       console.log("in created(), getting credentials...");
46       this.credentials = this.getCredentials();
47     }
48   },
49   methods: {
50     getCredentials() {
51       console.log("in getCredentials()");
52
53       var promise = AWS.config.credentials.getPromise();
54       promise.then(
55         function() {
56           console.log("result...");
57           var creds = {
58             accessKeyId: AWS.config.credentials.accessKeyId,
59             secretAccessKey: AWS.config.credentials.secretAccessKey,
60             sessionToken: AWS.config.credentials.sessionToken
```

```
61     };
62     console.log("got credentials: " + JSON.stringify(creds, null, 2));
63     return creds;
64   },
65   function(err) {
66     console.log("err..." + JSON.stringify(err, null, 2));
67     return null;
68   }
69 );
70 },
71 addItem() {
72   console.log("in addItem()");
73   // adds given item
74   var docClient = new AWS.DynamoDB.DocumentClient({
75     apiVersion: "2012-08-10"
76   });
77
78   var params = {
79     TableName: "TestDynamo",
80     Item: {
81       phone: "123456789",
82       name: "Bob"
83     }
84   };
85
86   if (this.credentials) {
87     console.log("in addItem(), have credentials, adding item...");
88     docClient.put(params, function(err, data) {
89       if (err) {
90         console.log("Error adding item", JSON.stringify(err, null, 2));
91       } else {
92         console.log("Success adding item", JSON.stringify(data, null, 2));
93       }
94     });
95   } else {
96     console.log("in addItem(), credentials null, getting credentials...");
97
98     var that = this;
99     var promise = AWS.config.credentials.getPromise();
100    promise.then(
101      function() {
102        console.log("in addItem(), got credentials, adding item...");
103        var creds = {
104          accessKeyId: AWS.config.credentials.accessKeyId,
105          secretAccessKey: AWS.config.credentials.secretAccessKey,
106          sessionToken: AWS.config.credentials.sessionToken
107        };
108        that.credentials = creds;
109        docClient.put(params, function(err, data) {
110          if (err) {
111            console.log("Error adding item", JSON.stringify(err, null, 2));
112          } else {
113            console.log(
114              "Success adding item",
115              JSON.stringify(data, null, 2)
116            );
117          }
118        });
119      },
120      function(err) {
```

```
121     console.log(  
122         "err getting credentials..." + JSON.stringify(err, null, 2)  
123     );  
124     that.credentials = null;  
125     }  
126     );  
127     }  
128 },  
129 deleteItem() {  
130     // deletes item for given key  
131     console.log("in deleteItem()");  
132  
133     var docClient = new AWS.DynamoDB.DocumentClient({  
134         apiVersion: "2012-08-10"  
135     });  
136  
137     var params = {  
138         TableName: "TestDynamo",  
139         Key: {  
140             phone: "123456789"  
141         }, ReturnValues: "NONE"  
142     };  
143  
144     if (this.credentials) {  
145         console.log("in deleteItem(), got credentials....");  
146  
147         docClient.delete(params, function(err, data) {  
148             if (err) {  
149                 console.log(  
150                     "error deleting item, err: " + JSON.stringify(err, null, 2)  
151                 );  
152             } else {  
153                 console.log(  
154                     "success deleting item: " + JSON.stringify(data, null, 2)  
155                 );  
156             }  
157         });  
158     } else {  
159         console.log("in deleteItem(), no credentials....getting credentials");  
160         var that = this;  
161         var promise = AWS.config.credentials.getPromise();  
162         promise.then(  
163             function() {  
164                 console.log("in deleteItem(), got credentials, deleting item...");  
165                 var creds = {  
166                     accessKeyId: AWS.config.credentials.accessKeyId,  
167                     secretAccessKey: AWS.config.credentials.secretAccessKey,  
168                     sessionToken: AWS.config.credentials.sessionToken  
169                 };  
170                 that.credentials = creds;  
171                 docClient.delete(params, function(err, data) {  
172                     if (err) {  
173                         console.log(  
174                             "error deleting item, err: " + JSON.stringify(err, null, 2)  
175                         );  
176                     } else {  
177                         console.log(  
178                             "success deleting item: " + JSON.stringify(data, null, 2)  
179                         );  
180                     }  
181                 }  
182             )  
183         }  
184     }  
185 }  
186 }  
187 }  
188 }  
189 }  
190 }
```

```
181     });
182   },
183   function(err) {
184     console.log(
185       "err getting credentials..." + JSON.stringify(err, null, 2)
186     );
187     that.credentials = null;
188   }
189 );
190 }
191 },
192 updateItem() {
193   // updates (overwrites) existing item, adds item if not exist, returns updated
attributes
194   console.log("in updateItem()");
195
196   var docClient = new AWS.DynamoDB.DocumentClient({
197     apiVersion: "2012-08-10"
198   });
199
200   var params = {
201     TableName: "TestDynamo",
202     Key: { phone: "123456789" },
203     UpdateExpression: "set #n = :x",
204     ExpressionAttributeNames: { "#n": "name" },
205     ExpressionAttributeValues: {
206       ":x": "Peter"
207     },
208     ReturnValues: "ALL_OLD"
209   };
210
211   if (this.credentials) {
212     console.log("in updateItem(), got credentials....");
213
214     docClient.update(params, function(err, data) {
215       if (err) {
216         console.log(
217           "error updating item, err: " + JSON.stringify(err, null, 2)
218         );
219       } else {
220         console.log(
221           "success updating item: " + JSON.stringify(data, null, 2)
222         );
223       }
224     });
225   } else {
226     console.log("in updateItem(), no credentials....getting credentials");
227     var that = this;
228     var promise = AWS.config.credentials.getPromise();
229     promise.then(
230       function() {
231         console.log("in updateItem(), got credentials, updating item...");
232         var creds = {
233           accessKeyId: AWS.config.credentials.accessKeyId,
234           secretAccessKey: AWS.config.credentials.secretAccessKey,
235           sessionToken: AWS.config.credentials.sessionToken
236         };
237         that.credentials = creds;
238         docClient.update(params, function(err, data) {
239           if (err) {
```

```
240         console.log(  
241             "error updating item, err: " + JSON.stringify(err, null, 2)  
242         );  
243     } else {  
244         console.log(  
245             "success updating item: " + JSON.stringify(data, null, 2)  
246         );  
247     }  
248 });  
249 },  
250 function(err) {  
251     console.log(  
252         "err getting credentials..." + JSON.stringify(err, null, 2)  
253     );  
254     that.credentials = null;  
255 }  
256 );  
257 }  
258 },  
259 getItem() {  
260     console.log("in getItem()");  
261  
262     var docClient = new AWS.DynamoDB.DocumentClient({  
263         apiVersion: "2012-08-10"  
264     });  
265  
266     var params = {  
267         TableName: "TestDynamo",  
268         Key: {  
269             phone: "123456789"  
270         }  
271     };  
272  
273     if (this.credentials) {  
274         console.log("in getItem(), got credentials....");  
275  
276         docClient.get(params, function(err, data) {  
277             if (err) {  
278                 console.log(  
279                     "error getting item, err: " + JSON.stringify(err, null, 2)  
280                 );  
281             } else {  
282                 console.log(  
283                     "success getting item: " + JSON.stringify(data, null, 2)  
284                 );  
285                 if(data==null){  
286                     console.log('data is null');  
287                 }else{  
288                     console.log('data is not null');  
289                 }  
290             }  
291         });  
292     });  
293 } else {  
294     console.log("in getItem(), no credentials....getting credentials");  
295     var that = this;  
296     var promise = AWS.config.credentials.getPromise();  
297     promise.then(  
298         function() {  
299             console.log("in addItem(), got credentials, getting item...");
```

```
300     var creds = {
301         accessKeyId: AWS.config.credentials.accessKeyId,
302         secretAccessKey: AWS.config.credentials.secretAccessKey,
303         sessionToken: AWS.config.credentials.sessionToken
304     };
305     that.credentials = creds;
306     docClient.get(params, function(err, data) {
307         if (err) console.log(err);
308         else console.log(data);
309     });
310 },
311 function(err) {
312     console.log(
313         "err getting credentials..." + JSON.stringify(err, null, 2)
314     );
315     that.credentials = null;
316 }
317 );
318 }
319 }
320 }
321 };
322 </script>
323
```