

RETRIEVE DATA



Get Data with Cloud Firestore

ใน Cloud Firestore สามารถดึงข้อมูล และสอบถามข้อมูลได้ 2 วิธี

1. เรียกใช้ method เพื่อดึงข้อมูล

2. คอยมอนิเตอร์ว่ามีการเปลี่ยนแปลงของข้อมูลหรือไม่

- แต่ก่อนที่จะไปดูวิธีการดึงข้อมูล
- อยากให้ใส่ตัวอย่างข้อมูลก่อน

01-getdata.html

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset="UTF-8">
5      <title>Get Data</title>
6
7      <script src="https://www.gstatic.com/firebasejs/8.2.2/firebase-app.js"></script>
8      <script src="https://www.gstatic.com/firebasejs/8.2.2/firebase-firestore.js"></script>
9  </head>
10 <body>
11     <button id="createExampleDataButton">Create Example Data</button>
12
13     <script src="01-getdata.js"></script>
14 </body>
15 </html>
```

01-getdata.js

```
1  // Your web app's Firebase configuration
2  var firebaseConfig = {
3      apiKey: "AIzaSyD5gqz_VK_sXupiVizX8fc6OAzs12mo4-8",
4      authDomain: "my-project-b8ca5.firebaseio.com",
5      projectId: "my-project-b8ca5",
6      storageBucket: "my-project-b8ca5.appspot.com",
7      messagingSenderId: "866862940608",
8      appId: "1:866862940608:web:95db1d1039685a3f960825"
9  };
10 // Initialize Firebase
11 firebase.initializeApp(firebaseConfig);
12
13 var database = firebase.firestore();
14
15 const createExampleDataButton = document.querySelector("createExampleDataButton");
16
```

01-getdata.js

```
17 expData.addEventListener("click", function () {
18     var citiesRef = database.collection("cities");
19
20
21     citiesRef.doc("SF").set({
22         name: "San Francisco",
23         state: "CA",
24         country: "USA",
25         capital: false,
26         population : 860000
27     });
```

```
29     citiesRef.doc("LA").set({
30         name: "Los Angeles",
31         state: "CA",
32         country: "USA",
33         capital: false,
34         population : 390000
35     });
36
```

```
37     citiesRef.doc("DC").set({
38         name: "Washington D.C.",
39         state: null,
40         country: "USA",
41         capital: true,
42         population : 680000
43     });
44
45     citiesRef.doc("TOK").set({
46         name: "Tokyo",
47         state: null,
48         country: "Japan",
49         capital: true,
50         population : 9000000
51     });
52
53     citiesRef.doc("BJ").set({
54         name: "Beijing",
55         state: null,
56         country: "China",
57         capital: true,
58         population : 21500000
59     });
60 });
```

หรือถ้าอยากเขียนแบบสั้นๆ ให้ตรวจสอบจาก onclick ที่ปุ่มก็ได้

01-getdata.html

```
11 <body>
12
13     <button id="expData" onclick="fn1()">Create Example Data</button>
14     <script src="01-getdata.js"></script>
15 </body>
```











01-getdata.js

```
15 function fn1 () {
16
17     var citiesRef = database.collection("cities");
18 }
```

01-get data.html

Create Example Data

Firestore console

 my-project-b8ca5	 cities  	 BJ 
 เริ่มต้นคอลเล็กชัน	 เพิ่มเอกสาร	 เริ่มต้นคอลเล็กชัน
cities >	BJ >	 เพิ่มช่อง
samples	DC LA SF TOK	capital: true country: "China" name: "Beijing" population: 21500000 state: null

Get a Document

การทำงานนี้ จะใช้คำสั่ง `get()` ซึ่งเป็นการรับข้อมูลเพียงครั้งเดียว
ในที่นี้หมายถึง เราไม่ต้องการรับรู้ว่ ที่ DB ของเรานั้น มีการเปลี่ยนแปลงอะไรบ้าง
`get()` จะทำการดึงข้อมูลปัจจุบัน ณ ขณะเวลานั้น ๆ ออกมาให้

01-getdata.html

```
11 <body>
12
13     <button id="expData">Create Example Data</button>
14     <button id="getDocument">Get A Document</button>
15
16 <script src="01-getdata.js"></script>
17 </body>
```


01-getdata.js

```
62     const getDocument = document.querySelector("#getDocument");
63
64     getDocument.addEventListener("click", function() {
65         var docRef = database.collection("cities").doc("SF");
66
67         docRef.get()
68             .then(function (doc) {
69                 if (doc && doc.exists) {
70                     console.log("Doc ID", doc.id, "Document data: ", doc.data());
71                 } else {
72                     console.log("No such document!");
73                 }
74             })
75             .catch(function (error) {
76                 console.log("Error getting document: ", error);
77             });
78     });
```

01-get data.html

Create Example Data

Get A Document

doc.id

```
Doc ID SF Document data: ▼{country: "USA", population: 860000, name: "San Francisco", state: "CA", capital: false} ⓘ  
  capital: false  
  country: "USA"  
  name: "San Francisco"  
  population: 860000  
  state: "CA"
```

doc.data()

- จากรูป เราจะเห็นว่า ข้อมูลที่ส่งกลับมานั้น อยู่ในรูปแบบของ JSON ซึ่งเราสามารถอ้างถึง key ที่ต้องการ เพื่อให้ได้ value ออกมา เช่น ต้องการ USA ให้อ้างไปที่ key ที่ชื่อว่า country
- doc.data().country => USA

Quiz

Create Example Data

BJ

DC

LA

SF

```
capital: true  
country: "China"  
name: "Beijing"  
population: 21500000  
state: null
```

```
capital: true  
country: "USA"  
name: "Washington D.C."  
population: 680000  
state: null
```

```
capital: false  
country: "USA"  
name: "Los Angeles"  
population: 390000  
state: "CA"
```

```
capital: false  
country: "USA"  
name: "San Francisco"  
population: 860000  
state: "CA"
```



Firebase

Get multiple documents from a collection

- เราสามารถดึงข้อมูลจากหลาย document ได้ ด้วยการใช้คำสั่งเพียงคำสั่งเดียว
- เช่น หากเราต้องการข้อมูลเฉพาะ ที่เป็นเมืองหลวงเท่านั้น

01-getdata.html

```
11 <body>
12
13     <button id="expData">Create Example Data</button>
14     <button id="getDocument">Get A Document</button>
15     <button id="getMultipleDoc">Get Multiple Document</button>
16
17     <script src="01-getdata.js"></script>
18 </body>
```

01-getdata.js

```
80     const getMultipleDoc = document.querySelector("#getMultipleDoc");
81
82     getMultipleDoc.addEventListener("click", function() {
83         database.collection("cities")
84             .where("capital", "==", true)
85             .get()
86             .then(function (querySnapshot) {
87                 querySnapshot.forEach(function (doc) {
88                     console.log(doc.id, " => ", doc.data());
89                 })
90             })
91             .catch(function (error) {
92                 console.log("Error getting document: ", error);
93             });
94     });
```

01-get data.html

Create Example Data

Get A Document

Get Multiple Document

```
BJ => ▶ {name: "Beijing", country: "China", state: null, population: 21500000, capital: true}
```

```
DC => ▶ {capital: true, population: 680000, country: "USA", state: null, name: "Washington D.C."}
```

```
TOK => ▶ {country: "Japan", state: null, capital: true, name: "Tokyo", population: 9000000}
```

Get all documents in a collection

- จากตัวอย่างก่อนหน้านี้ ทำให้เราสามารถประยุกต์ใช้ได้
- หากเราต้องการ document ทั้งหมด ที่อยู่ใน collection นั้น

01-getdata.html

```
L1 <body>
L2
L3     <button id="expData">Create Example Data</button>
L4     <button id="getDocument">Get A Document</button>
L5     <button id="getMultipleDoc">Get Multiple Document</button>
L6     <button id="getAllDoc">Get All Document</button>
L7
L8     <script src="01-getdata.js"></script>
L9 </body>
```

01-getdata.js

```
96     const getAllDoc = document.querySelector("#getAllDoc");
97
98     getAllDoc.addEventListener("click", function () {
99         database.collection("cities")
100             .get()
101             .then(function (querySnapshot) {
102                 querySnapshot.forEach(function (doc) {
103                     console.log(doc.id, " => ", doc.data());
104                 });
105             });
106     });
```


01-get data.html

Create Example Data

Get A Document

Get Multiple Document

Get All Document

```
BJ => ▶ {state: null, population: 21500000, capital: true, name: "Beijing", country: "China"}
```

```
DC => ▶ {population: 680000, state: null, country: "USA", capital: true, name: "Washington D.C."}
```

```
LA => ▶ {country: "USA", capital: false, population: 390000, name: "Los Angeles", state: "CA"}
```

```
SF => ▶ {country: "USA", capital: false, name: "San Francisco", population: 860000, state: "CA"}
```

```
TOK => ▶ {capital: true, population: 9000000, country: "Japan", name: "Tokyo", state: null}
```

Get RealtimeUpdates

- คุณสมบัติหนึ่งของ Firebase คือการเป็นฐานข้อมูลที่สามารถทำงานแบบ realtime ได้
- การทำงานไม่จำเป็นต้องปรับแต่งฐานข้อมูลแต่อย่างใด ทำแค่เพียงเรียกใช้ function ที่มีการทำงานแบบ realtime เท่านั้น

Get RealtimeUpdates with Cloud Firestore

- เราจะใช้ `.onSnapshot()` method ในการมอนิเตอร์ document ที่เราสนใจ
- หากข้อมูลใน document นั้นเกิดการเปลี่ยนแปลงเมื่อไหร่ จะมีการเรียก callback มาที่ snapshot ทันที

01-getdata.html

```
11 <body>
12
13     <button id="expData">Create Example Data</button>
14     <button id="getDocument">Get A Document</button>
15     <button id="getMultipleDoc">Get Multiple Document</button>
16     <button id="getAllDoc">Get All Document</button>
17     <button id="getRealtimeDoc">Get Realtime Document</button>
18
19     <script src="01-getdata.js"></script>
```

Get Realtime Document

```
Current ID: SF Current data: {capital: false, state: "CA", population: 860000, country: "USA", name: "San Francisco"}  
  capital: false  
  country: "USA"  
  name: "San Francisco"  
  population: 860000  
  state: "CA"
```

cities	SF
+ เพิ่มเอกสาร	+ เริ่มต้นคอลเล็กชัน
BJ	+ เพิ่มช่อง
DC	capital: false
LA	country: "THAILAND"
SF	name: "San Francisco"
TOK	population: 860000
	state: "CA"

ในหน้า Firebase console
เปลี่ยน value เป็น
"Thailand"

```
Current ID: SF Current data: {capital: false, state: "CA", population: 860000, country: "USA", name: "San Francisco"}  
  capital: false  
  country: "USA"  
  name: "San Francisco"  
  population: 860000  
  state: "CA"  
  __proto__: Object
```

```
Current ID: SF Current data: {name: "San Francisco", capital: false, state: "CA", population: 860000, country: "THAILAND"}  
  capital: false  
  country: "THAILAND"  
  name: "San Francisco"  
  population: 860000  
  state: "CA"
```

Listen to multiple documents in a collection

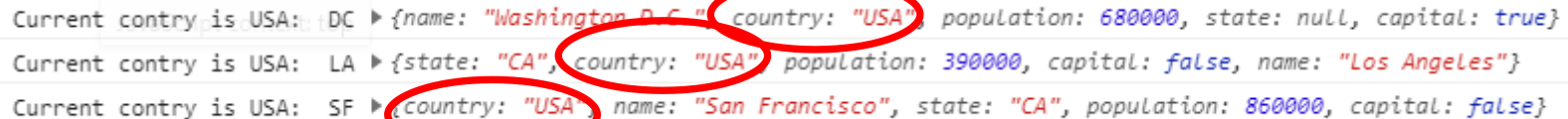
- เราจะใช้ .onSnapshot() method แทนการใช้ get()
- ในที่นี้ เราต้องการคอยตรวจสอบ document เฉพาะที่มี country เป็น USA

01-getdata.html

```
13 <button id="expData">Create Example Data</button>
14 <button id="getDocument">Get A Document</button>
15 <button id="getMultipleDoc">Get Multiple Document</button>
16 <button id="getAllDoc">Get All Document</button>
17 <button id="getRealtimeDoc">Get Realtime Document</button>
18 <button id="listenMultipleDoc">Listen Multiple Document</button>
```

01-getdata.js

```
118     const listenMultipleDoc = document.querySelector("#listenMultipleDoc");
119
120     listenMultipleDoc.addEventListener("click", function () {
121         database.collection("cities")
122             .where("country", "==", "USA")
123             .onSnapshot(function (querySnapshot) {
124                 querySnapshot.forEach(function (doc) {
125                     console.log("Current contry is USA: ", doc.id, doc.data());
126                 });
127             });
128     });
```



```
Current contry is USA: DC {name: "Washington D.C.", country: "USA", population: 680000, state: null, capital: true}
Current contry is USA: LA {state: "CA", country: "USA", population: 390000, capital: false, name: "Los Angeles"}
Current contry is USA: SF {country: "USA", name: "San Francisco", state: "CA", population: 860000, capital: false}
```

Quiz

จงแก้ไขคำสั่ง เพื่อให้สามารถแสดงผลออกมาดังนี้

- เมื่อทำการกดปุ่ม get All Document, State CA ให้ทำการดึง document ทั้งหมดที่ state มีค่าเท่ากับ CA มาแสดงผลในรูป

get All Document, State CA

```
LA => ▼{state: "CA", capital: false, name: "Los Angeles", country: "USA", population: 390000} ⓘ  
  capital: false  
  country: "USA"  
  name: "Los Angeles"  
  population: 390000  
  state: "CA"  
  ▶ __proto__: Object  
  
SF => ▼{country: "USA", capital: false, population: 860000, state: "CA", name: "San Francisco"} ⓘ  
  capital: false  
  country: "USA"  
  name: "San Francisco"  
  population: 860000  
  state: "CA"  
  ▶ __proto__: Object
```

Snapshot จะทำการรับข้อมูลทุก ๆ ครั้ง ที่มีการเปลี่ยนแปลงเกิดขึ้นใน document

- สถานะการเปลี่ยนแปลงของข้อมูลใน document
 - added
 - modified
 - removed

View changes between snapshots

การทำงานส่วนนี้ค่อนข้างมีประโยชน์ในการตรวจสอบการเปลี่ยนแปลงที่เกิดขึ้นระหว่างที่มีการทำ query snapshot สถานะของ document มีด้วยกัน 3 สถานะ added, removed และ modified

01-getdata.html

```
11 <body>
12
13     <button id="expData">Create Example Data</button>
14     <button id="getDocument">Get A Document</button>
15     <button id="getMultipleDoc">Get Multiple Document</button>
16     <button id="getAllDoc">Get All Document</button>
17     <button id="getRealtimeDoc">Get Realtime Document</button>
18     <button id="listenMultipleDoc">Listen Multiple Document</button>
19     <button id="viewChangeButton">View Changes between snapshots</button>
20
21 <script src="01-getdata.js"></script>
22 </body>
```

ทดสอบแก้ไข, เพิ่ม และ ลบ ข้อมูลใน Firebase Console

cities	SF
<div>+ เพิ่มเอกสาร</div> <div>BJ</div> <div>DC</div> <div>LA</div> <div>SF</div> <div>TOK</div>	<div>+ เริ่มต้นคอลเล็กชัน</div> <div>+ เพิ่มช่อง</div> <div>capital: false</div> <div>country: "USA"</div> <div>name: "San Francisco"</div> <div>population: 860000</div> <div>state: "CA"</div>

```
Modified city: {country: "USA", population: 860000, name: "San Francisco", capital: false, state: "CA"}
New city: {state: "CA"}
Removed city: {state: "CA"}
```

เก็บคะแนนท้ายคาบ

- เพิ่ม code สำหรับเพิ่มข้อมูลใน document id BJ
- เมื่อทำการกดปุ่ม "EDIT BJ CITY" ให้ทำการแก้ไขข้อมูล และ Get ค่า Data มาแสดงดังรูปด้านขวา

Name:

Country:

Population:

EDIT BJ City

+ เพิ่มเอกสาร	+ เริ่มต้นคอลเล็กชัน
BJ >	+ เพิ่มช่อง
DC	capital: true
LA	country: "China"
SF	name: "Beijing"
TOK	population: 21500000
	state: null

Name:

Country:

Population:

EDIT BJ City

Edited BJ Document data: `{name: "5", population: "5", capital: true, state: null, country: "5"}`

+ เพิ่มเอกสาร	+ เริ่มต้นคอลเล็กชัน
BJ >	+ เพิ่มช่อง
DC	capital: true
LA	country: "5"
SF	name: "5"
TOK	population: "5"
	state: null