

SIMPLE QUERIES



Perform Simple and Compound Queries in Cloud Firestore

- Cloud Firestore เตรียมวิธีการสอบถามข้อมูล (query) เอาไว้ให้แล้ว
- Query function ของ Cloud Firestore สามารถใช้กับ `get()` หรือ `onSnapshot()` ได้

Example Data

- เพิ่มตัวอย่างข้อมูลก่อน
- 01-query.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/8.2.4/firebase-app.js"></script>
8     <script src="https://www.gstatic.com/firebasejs/8.2.4/firebase-firestore.js"></script>
9
10  </head>
11  <body>
12
13    <button id="expData">Create Example Data</button>
14
15
16    <script src="01-query.js"></script>
17  </body>
18 </html>
```

01-query.js

```
1 // Your web app's Firebase configuration
2   var firebaseConfig = {
3     apiKey: "AIzaSyD5gqz_VK_sXupiVizX8fc60Azs12mo4-8",
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();
```

01-query.js (ต่อ)






```
14
15     const expData = document.querySelector("#expData");
16
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     });
28
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 });
61
```

01-query.html

Create Example Data

Cloud FirestoreConsole

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

Simple queries

จากตัวอย่างก่อนหน้านี้ พบว่ามีการใช้ where() method ไปแล้ว

- ใช้เพื่อทำการ filter ข้อมูลต่าง ๆ เพื่อให้ได้ข้อมูลตรงตามความต้องการ
- where() ประกอบด้วย 3 parameters คือ
 - fieldPath คือชื่อของฟิลด์ที่เราต้องการเอามาสร้างเงื่อนไขในการสอบถาม (query)
 - opStr คือ operating string คือ เงื่อนไขที่ใช้ในการเปรียบเทียบกันระหว่าง fieldPath กับ value ในตอนนี้มีให้ใช้ด้วยกัน 5 operating คือ

“<”, “<=”, “==”, “>”, “>=”

- ตอนใช้งานจริง ให้ใส่ “ ” ลงไปด้วย
- value คือ ค่าที่เราต้องการเอาไปเปรียบเทียบกับ fieldPath โดยปกติ จะเป็นค่าที่ได้มาจากผู้ใช้งาน
- where(fieldPath, opStr, value)

Example

- ตัวอย่างเช่น ต้องการหาเมืองที่อยู่ในรัฐที่มีชื่อย่อว่า CA

01-query.html

```
11 <body>
12
13     <button id="expData">Create Example Data</button>
14     <button id="simpleQuery">Simple Query</button>
15
16
17 <script src="01-query.js"></script>
18 </body>
```


01-query.js

```
62  const simpleQuery = document.querySelector("#simpleQuery");
63  simpleQuery.addEventListener("click", function () {
64      database.collection("cities")
65          .where("state", "==", "CA")
66          .get()
67          .then(function (querySnapshot) {
68              querySnapshot.forEach(function (doc) {
69                  console.log(doc.data());
70              })
71          })
72  });
```

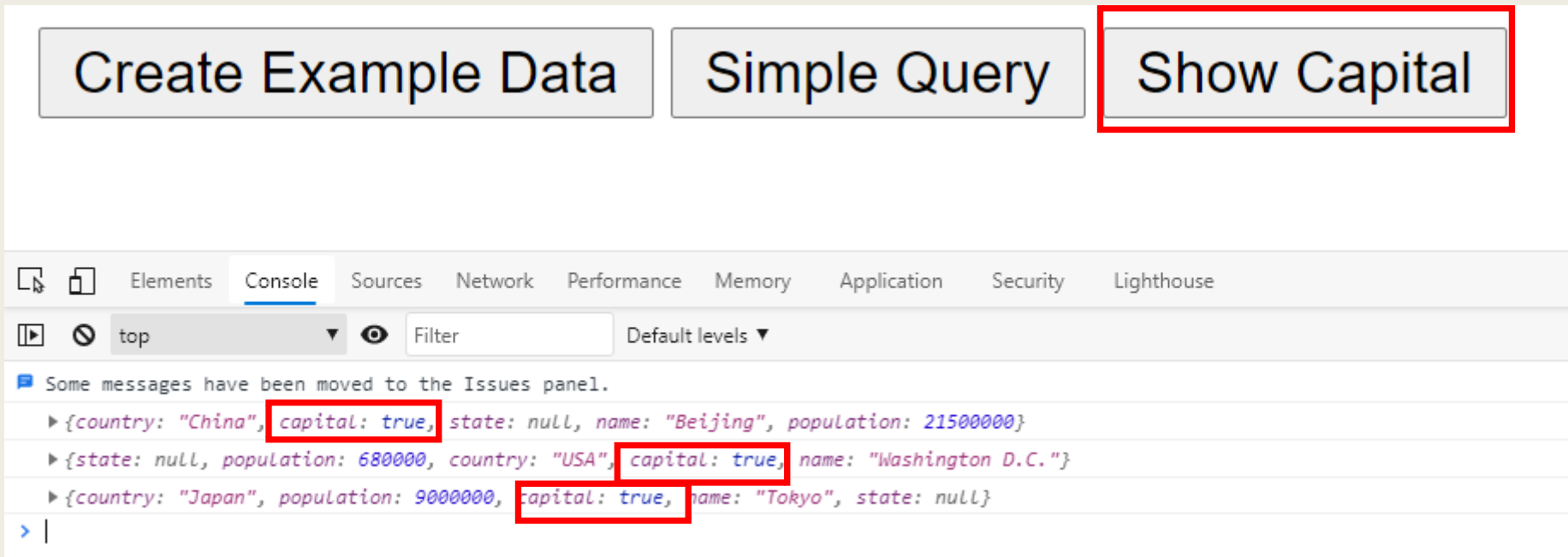
where(fieldPath, opStr, value)

Result



ทบทวน

ให้นักศึกษาเพิ่มปุ่มขึ้นมาอีกปุ่ม โดยเมื่อกดแล้ว จะแสดงข้อมูลเฉพาะเมืองที่เป็นเมืองหลวงเท่านั้น (capital = true)



ทบทวน

ให้นักศึกษาทำการแสดงผลลัพธ์ เมืองที่มีประชากรมากกว่าที่ผู้ใช้กรอกเข้ามา

Population More than

50000000

search

Elements Console Sources Network Performance Memory Application Security Lighthouse Adblock Plus

top Filter Default levels

{state: null, country: "Japan", capital: true, population: 90000000, name: "Tokyo"}

{country: "China", capital: true, name: "Beijing", state: null, population: 215000000}

Compound queries

การตัวอย่างก่อนหน้านี้เรามี where() แค่ 1 ตัวเท่านั้น

- แต่หากเรามีเงื่อนไขมากกว่า 1 เงื่อนไข ก็ยังสามารถทำได้
- เราสามารถใช้คำสั่ง where() ต่อไปได้เรื่อย ๆ โดยในแต่ละเงื่อนไข จะเป็นการทำ AND กัน (เงื่อนไขทุกอันต้องเป็นจริง)
- ทั้งนี้ หากใน where() ที่มากกว่า 1 where() มีการใช้ “==” ร่วมกับ range comparison (“<”, “<=”, “>” or “>=”) จะต้องมีการทำ index ก่อน
- และ เราจะสามารถใช้ range comparison ได้กับแค่ field เดียวเท่านั้น

ตัวอย่างเช่น ต้องการหาเมืองที่อยู่ในประเทศ USA และไม่ใช่เมืองหลวง

- 01-query.html

```
19         <button id="compoundQuery">Compound Query</button>
20
21
22     <script src="01-query.js"></script>
23 </body>
```

01-query.js

```
101  const compoundQuery = document.querySelector("#compoundQuery");
102  compoundQuery.addEventListener("click", function () {
103      database.collection("cities")
104          .where("capital", "==", false)
105          .where("country", "==", "USA")
106          .get()
107          .then(function (querySnapshot) {
108              querySnapshot.forEach(function (doc) {
109                  console.log(doc.data());
110              })
111          })
112  });
```

Result

Compound Query

The screenshot shows a web browser's developer console with the 'Console' tab selected. The console displays two JSON objects. The first object is for Los Angeles, and the second is for San Francisco. Both objects have 'capital: false' and 'country: USA' highlighted with red boxes.

```
{state: "CA", name: "Los Angeles", country: "USA", population: 390000, capital: false}
capital: false
country: "USA"
name: "Los Angeles"
population: 390000
state: "CA"
__proto__: Object

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


Order and Limit Data with Cloud Firestore

ก่อนหน้านี้ เราได้ทำการใช้คำสั่ง `.where()` เพื่อเลือกว่าต้องการข้อมูลจาก document ไหน จาก collection ไດมาแล้ว

- Cloud Firestore ยังสามารถให้เราทำการเรียงข้อมูลของผลลัพธ์ตามที่เราต้องการได้

`(orderBy())`

- และสามารถเลือกจำนวนของผลลัพธ์ตามที่ต้องการได้เช่นกัน

`(limit())`

.orderBy()

- orderBy(fieldPath, directionStr) method ประกอบด้วย 2 parameter
- fieldPath คือ ชื่อของ field ที่เราต้องการเรียงข้อมูล
- directionStr คือ ทิศทางที่เราต้องการให้เรียงข้อมูล
 - asc เรียงจากน้อยไปมาก เป็นค่า default
 - desc เรียงจากมากไปน้อย

.limit()

- limit(limit) method ประกอบด้วย 1 parameter
- limit คือ จำนวนข้อมูลที่ต้องการเอามาใช้งาน

Example

ตัวอย่างเช่น ต้องการ 3 เมืองแรก โดยเรียงลำดับตามตัวอักษร

- 01-query.html

```
19      <button id="compoundQuery">Compound Query</button>
20      <button id="treeCities">Get First 3 Cities</button>
21
22
23      <script src="01-query.js"></script>
```

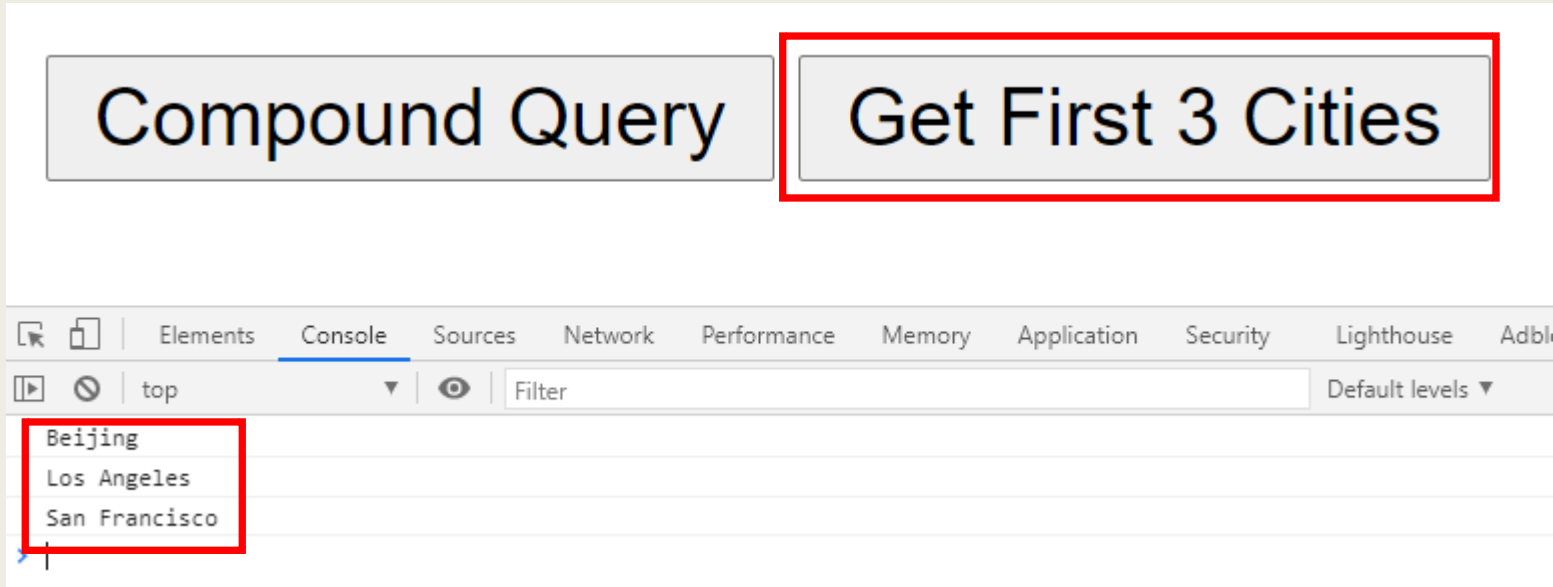
- 01-query.js

```
114     const treeCities = document.querySelector("#treeCities");
115     treeCities.addEventListener("click", function () {
116         database.collection("cities")
117             .orderBy("name")
118             .limit(3)
119             .get()
120             .then(function (querySnapshot) {
121                 querySnapshot.forEach(function (doc) {
122                     console.log(doc.data().name);
123                 })
124             })
125     });
```

Result

Compound Query

Get First 3 Cities



The screenshot displays a web application interface with a console log. The console is open, showing three log entries: "Beijing", "Los Angeles", and "San Francisco". The "Get First 3 Cities" button is highlighted with a red box, and the console output is also highlighted with a red box.

Log Entry
Beijing
Los Angeles
San Francisco

Example

ตัวอย่างเช่น ต้องการ 3 เมืองสุดท้าย โดยเรียงลำดับตามตัวอักษร

- 01-query.html

```
19      <button id="compoundQuery">Compound Query</button>  
20      <button id="treeCities">Get First 3 Cities</button>  
21      <button id="treeCities2">Get Last 3 Cities</button>  
22
```

- 01-query.js

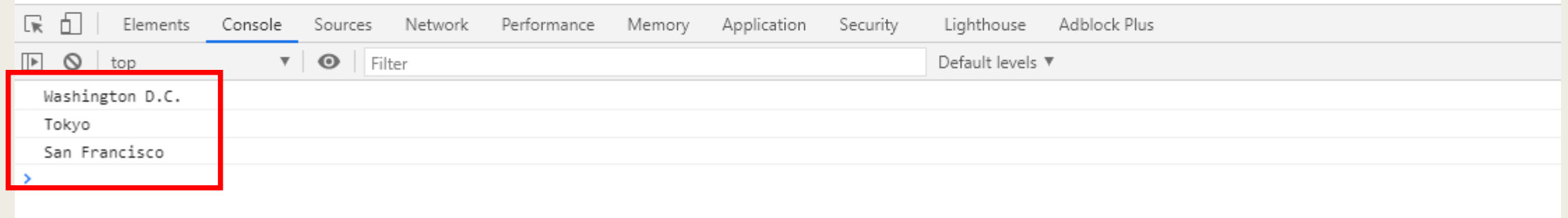
```
127  const treeCities2 = document.querySelector("#treeCities2");
128  treeCities2.addEventListener("click", function () {
129      database.collection("cities")
130          .orderBy("name", 'desc')
131          .limit(3)
132          .get()
133          .then(function (querySnapshot) {
134              querySnapshot.forEach(function (doc) {
135                  console.log(doc.data().name);
136              })
137          })
138  });
```


Result

Compound Query

Get First 3 Cities

Get Last 3 Cities



Example

ทั้งนี้ เราสามารถใช้ `.where()` ร่วมกับ `.orderBy()` `.limit()` ได้ด้วย

- ตัวอย่างเช่น ต้องการข้อมูลชื่อเมือง และจำนวนประชากร 2 อันดับแรก ที่มีประชากรมากกว่า 100000 โดยเรียงข้อมูลตามจำนวนประชากรจากมากไปหาน้อย
- 01-query.html

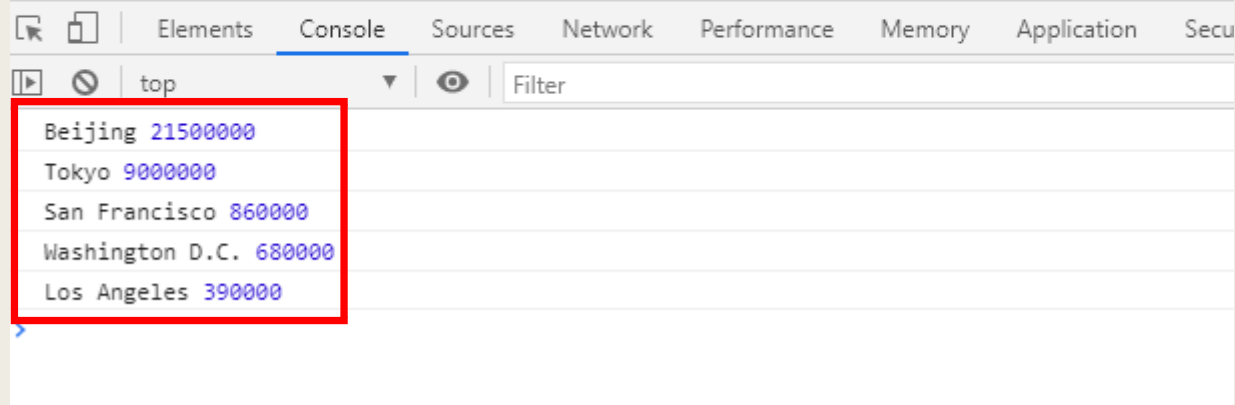
```
19 <button id="compoundQuery">Compound Query</button>
20 <button id="treeCities">Get First 3 Cities</button>
21 <button id="treeCities2">Get Last 3 Cities</button><br><br>
22 <button id="get2CitiesWithPop">Get 2 Cities With Population</button>
23
```

- 01-query.js

```
140     const get2CitiesWithPop = document.querySelector("#get2CitiesWithPop");
141     get2CitiesWithPop.addEventListener("click", function () {
142         database.collection("cities")
143             .where("population", ">", 100000)
144             .orderBy("population", 'desc')
145             .get()
146             .then(function (querySnapshot) {
147                 querySnapshot.forEach(function (doc) {
148                     console.log(doc.data().name, doc.data().population);
149                 })
150             })
151     });
```

Result

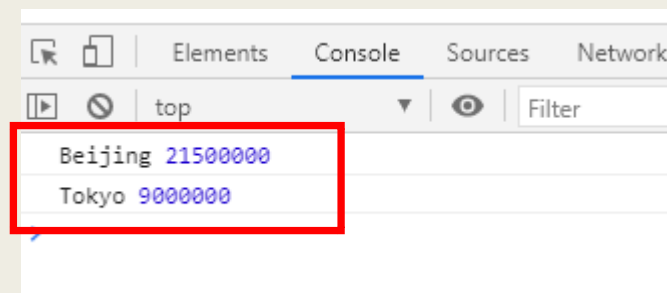
Get 2 Cities With Population



Beijing	21500000
Tokyo	9000000
San Francisco	860000
Washington D.C.	680000
Los Angeles	390000

หรือ ทดลองใส่ limit เข้าไป

```
.where("population", ">", 100000)  
.orderBy("population", 'desc')  
.limit(2)  
.get()
```



Beijing	21500000
Tokyo	9000000

ทดลอง

ให้นักศึกษาทำการแสดงผลลัพธ์ โดยมีปุ่มด้วยกันทั้งหมด 3ปุ่ม

- Show All Data เมื่อกดปุ่ม จะทำการดึงรายชื่อเมือง ประเทศ และจำนวนประชากรออกมาแสดงผล
- Order By Name with ASC เมื่อกดปุ่ม จะทำการดึงรายชื่อเมือง ประเทศ และจำนวนประชากรออกมาแสดงผล โดยเรียงผลลัพธ์ตามชื่อเมืองจาก A to Z
- Order By Name with DESC เมื่อกดปุ่ม จะทำการดึงรายชื่อเมือง ประเทศ และจำนวนประชากรออกมาแสดงผล โดยเรียงผลลัพธ์ตามชื่อเมืองจาก Z to A

Show All Data

Order By Name with ASC

Order By Name with DESC

Citty Name: BeijingCountry: ChinaPopulation: 21500000

Citty Name: Washington, D.C. Country: USAPopulation: 680000

Citty Name: Los AngelesCountry: USAPopulation: 3900000

Citty Name: San FranciscoCountry: USAPopulation: 860000

Citty Name: TokyoCountry: JapanPopulation: 9000000

Show All Data

Order By Name with ASC

Order By Name with DESC

Citty Name: BeijingCountry: ChinaPopulation: 21500000

Citty Name: Los AngelesCountry: USAPopulation: 3900000

Citty Name: San FranciscoCountry: USAPopulation: 860000

Citty Name: TokyoCountry: JapanPopulation: 9000000

Citty Name: Washington, D.C. Country: USAPopulation: 680000

Show All Data

Order By Name with ASC

Order By Name with DESC

Citty Name: Washington, D.C. Country: USAPopulation: 680000

Citty Name: TokyoCountry: JapanPopulation: 9000000

Citty Name: San FranciscoCountry: USAPopulation: 860000

Citty Name: Los AngelesCountry: USAPopulation: 3900000

Citty Name: BeijingCountry: ChinaPopulation: 21500000

Quiz

ให้นักศึกษาทำการแสดงผลลัพธ์ โดยมีปุ่มด้วยกันทั้งหมด 1 ปุ่ม

- Create Users เมื่อกดแล้วจะทำการสร้างข้อมูล โดยมีข้อกำหนดดังนี้

```
158     const docRef = database.collection("users");
159
160     docRef
161     .add({
162         firstname: "Harry",
163         lastname: "Potter",
164         gender: "Male",
165         birthdate: "07/31/1997",
166         weight : 65,
167         salary : 50000,
168         country : "England"
169     });
170     docRef
171     .add({
172         firstname: "Steve",
173         lastname: "Rogers",
174         gender: "Male",
175         birthdate: "07/04/1981",
176         weight : 75,
177         salary : 85000,
178         country : "USA"
179     });
```

```
180     docRef
181     .add({
182         firstname: "Ronnachut",
183         lastname: "Jinapangkas",
184         gender: "Male",
185         birthdate: "06/15/1989",
186         weight : 65,
187         salary : 20000,
188         country : "USA"
189     });
190     docRef
191     .add({
192         firstname: "Hermione",
193         lastname: "Granger",
194         gender: "Female",
195         birthdate: "09/19/1979",
196         weight : 55,
197         salary : 65000,
198         country : "England"
199     });
```

```
200 docRef
201   .add({
202     firstname: "Cherprang",
203     lastname: "Areekul",
204     gender: "Female",
205     birthdate: "05/02/1996",
206     weight : 48,
207     salary : 100000,
208     country : "Thailand"
209   });
210 docRef
211   .add({
212     firstname: "Anne",
213     lastname: "Hathaway",
214     gender: "Female",
215     birthdate: "11/12/1982",
216     weight : 55,
217     salary : 81000,
218     country : "USA"
219   });
220 console.log("Created Users");
221 });
```

Quiz 1

จงเขียนคำสั่งเพื่อแสดงข้อมูลผู้ใช้ เฉพาะที่เป็นผู้ชาย เท่านั้น (3 แถว)

```
▶ {firstname: "Ronnachut", gender: "Male", country: "USA", salary: 20000, birthdate: "06/15/1989", ...}  
▶ {salary: 50000, gender: "Male", birthdate: "07/31/1997", weight: 65, firstname: "Harry", ...}  
▶ {gender: "Male", weight: 75, salary: 85000, country: "USA", birthdate: "07/04/1981", ...}  
>
```


Quiz 2

จงเขียนคำสั่งเพื่อแสดงข้อมูลชื่อ น้ำหนัก และวันเกิดของผู้ใช้ เฉพาะที่เป็นผู้ชาย
ที่มีน้ำหนักเท่ากับ 65 kg เท่านั้น (2แถว)

```
name: Ronnachut weight: 65 Birthdate: 06/15/1989
```

```
name: Harry weight: 65 Birthdate: 07/31/1997
```

```
>
```

Quiz 3

จงเขียนคำสั่งเพื่อแสดงข้อมูลชื่อ เงินเดือน และประเทศของผู้ใช้ ทุกคนที่มีเงินเดือนมากกว่า 65000 (3 แถว)

```
name: Anne salary: 81000 country: USA
name: Steve salary: 85000 country: USA
name: Cherprang salary: 100000 country: Thailand
> |
```

Quiz 4

จงเขียนคำสั่งเพื่อแสดงข้อมูลชื่อ นามสกุล และน้ำหนักของผู้ใช้ทุกคน ที่มีน้ำหนักอยู่ในช่วง 55-65 kg (4แถว)

```
name: Hermione Lastname: Granger weight: 55
name: Anne Lastname: Hathaway weight: 55
name: Ronnachut Lastname: Jinapangkas weight: 65
name: Harry Lastname: Potter weight: 65
> |
```

Quiz 5

จงเขียนคำสั่งเพื่อแสดงข้อมูลชื่อ นามสกุล ประเทศ และเงินเดือนของผู้ใช้ทุกคน โดยเรียงข้อมูลด้วยเงินเดือนจากมากไปหาน้อย (6แถว)

name:	Cherprang	Lastname:	Areekul	country:	Thailand	salary:	100000
name:	Steve	Lastname:	Rogers	country:	USA	salary:	85000
name:	Anne	Lastname:	Hathaway	country:	USA	salary:	81000
name:	Hermione	Lastname:	Granger	country:	England	salary:	65000
name:	Harry	Lastname:	Potter	country:	England	salary:	50000
name:	Ronnachut	Lastname:	Jinapangkas	country:	USA	salary:	20000

Quiz 6

จงเขียนคำสั่งเพื่อแสดงข้อมูลชื่อ นามสกุล และอายุของผู้ใช้ทุกคน (6แถว)

name: Ronnachut	Lastname: Jinapangkas	Age: 31
name: Harry	Lastname: Potter	Age: 23
name: Cherprang	Lastname: Areekul	Age: 24
name: Steve	Lastname: Rogers	Age: 39
name: Hermione	Lastname: Granger	Age: 41
name: Anne	Lastname: Hathaway	Age: 38

Quiz 7

จงเขียนคำสั่งเพื่อแสดงข้อมูลชื่อ นามสกุล และน้ำหนักของผู้ใช้ทุกคน ที่น้ำหนักมากกว่า Hermione Granger Granger

name: Ronnachut	Lastname: Jinapangkas	weight: 65
name: Harry	Lastname: Potter	weight: 65
name: Steve	Lastname: Rogers	weight: 75
>		

Quiz 8

จงเขียนคำสั่งเพื่อแสดงข้อมูลชื่อ นามสกุล และน้ำหนักของผู้ใช้ทุกคน ที่น้ำหนักน้อยกว่า Hermione Granger Granger

```
name: Cherprang Lastname: Areekul weight: 48
```

```
>
```