

مدة اليوم: 6 ساعات

المستوى: مبتدئ إلى متوسط

(اختياري) Firebase (Compat API), JavaScript, HTML, Bootstrap: التكنولوجيا

Firestore الساعه الأولى: مقدمة + إعداد

مفاهيم:

- ما هو Firestore؟
- للبروتوكولات (Auth, Firestore, Hosting) خدمات Firestore
- إنشاء وإضافة Firestore Project كيفية إنشاء Web App

المثال:

```
<script src="https://www.gstatic.com/firebasejs/10.12.0/firebase-app-compat.js"></script>
<script>
  const firebaseConfig = {
    apiKey: "...",
    authDomain: "...",
    projectId: "..."
  };
  firebase.initializeApp(firebaseConfig);
</script>
```

HTML. ويربطه بصفحة Firestore Project ينشئ الطالب: **التاسك**.

firebase-app. سكربت + firebaseConfig تركيب: **الحل**.

2 الساعه: Firestore (Add/Get)

مفاهيم:

- Collection - Document - Fields
- مفهوم NoSQL

المثال:

```
const db = firebase.firestore();

db.collection("tasks").add({ title: "Buy milk", done: false });

db.collection("tasks").get().then(snapshot => {
  snapshot.forEach(doc => console.log(doc.data()));
});
```

UL لإضافة مهمة وعرضها في input + button إنشاء: **التاسك**

الحل:

- document.getElementById("taskInput").value
- db.collection("tasks").add(...)
- ul.innerHTML += ...

3 الساعة: Real-time Updates

مفاهيم:

- onSnapshot() vs get()

المثال:

```
db.collection("tasks").onSnapshot(snapshot => {
  list.innerHTML = "";
  snapshot.forEach(doc => {
    const task = doc.data();
    const li = document.createElement("li");
    li.textContent = task.title;
    list.appendChild(li);
  });
});
```

عند إضافة مهمة console للتسجيل في onSnapshot استخدام: **التاسك**

الحل: استخدام .forEach(doc => console.log(doc.data()))

4 الساعة: Firebase Authentication (Google)

مفاهيم:

- Google Sign-in
- مع مهام المستخدم UID ربط

المثال:

```
const auth = firebase.auth();
const provider = new firebase.auth.GoogleAuthProvider();

auth.signInWithPopup(provider).then(result => {
  console.log(result.user.displayName);
});
```

التاسك:

- عمل login button
- عرض اسم المستخدم بعد الدخول

الحل: `document.getElementById("userInfo").textContent = user.displayName`

الساعة 5: المشروع العملي

To-Do App:

- Google Login
- إضافة/حذف مهام
- عرض فقط مهام
- Realtime update

التاسك:

- إضافة Checkbox + Line-through
- عمل Logout

الحل:

- `db.collection("tasks").doc(id).update({ done: !done })`
 - CSS: `text-decoration: line-through`
-

6 الساعة: Challenge + Deploy

التحدي:

- ترتيب المهام حسب الوقت
- المهام المنتهية في نهاية القائمة

الحل:

- `orderBy("done")`
- `orderBy("createdAt")`

المستعملة API اليوم ينتهي بمشروع كامل ومراجعة لكل ال