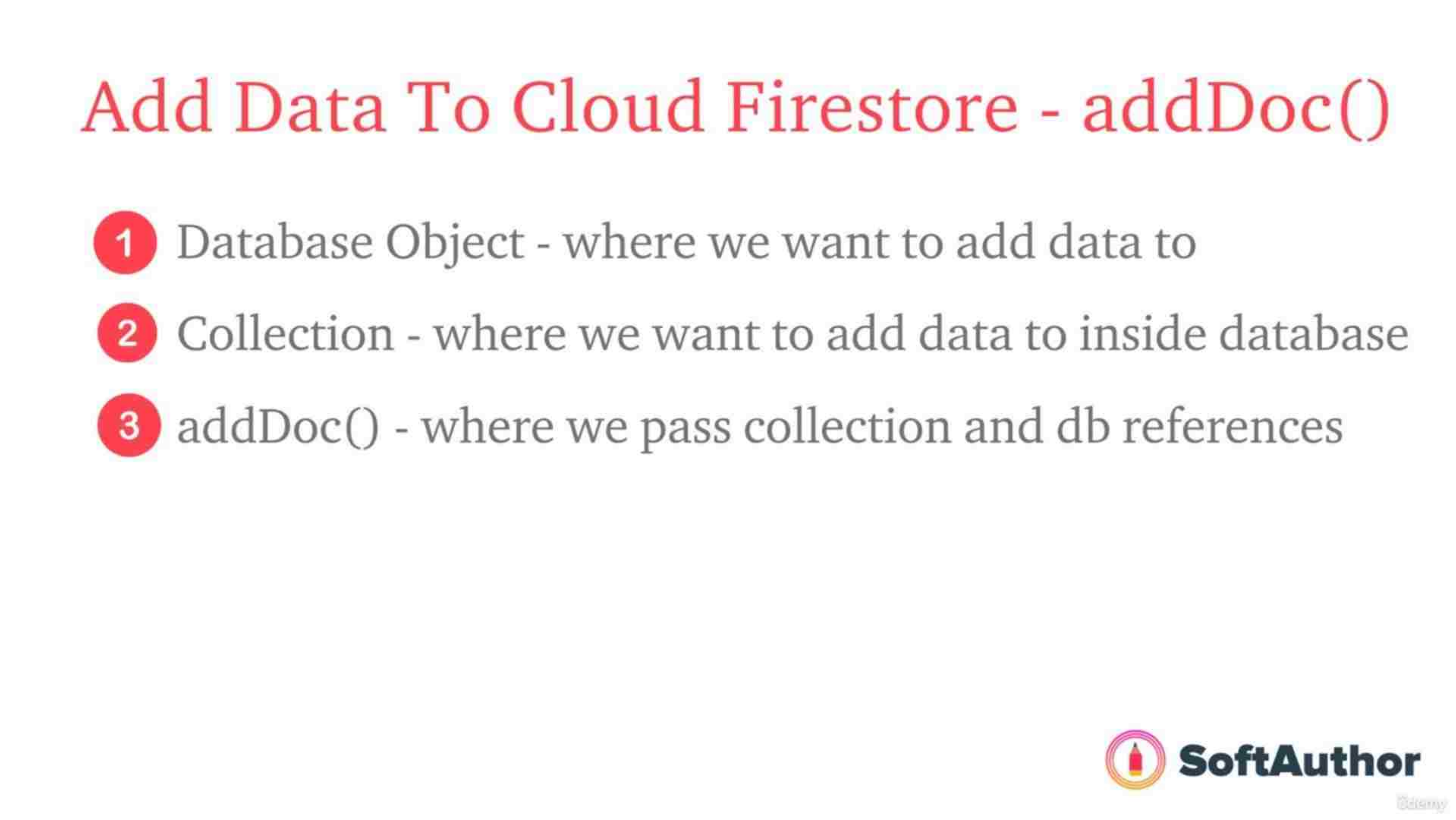
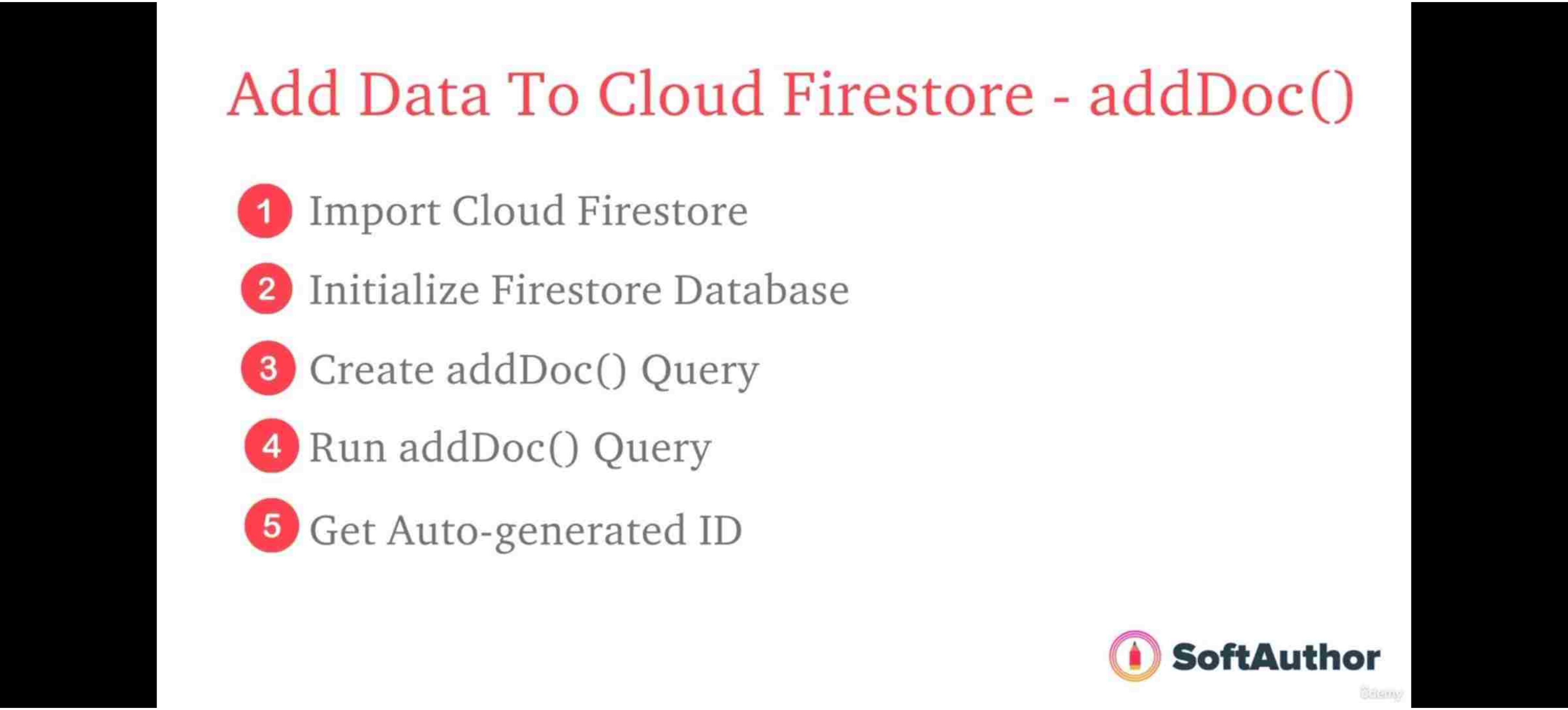


Add Data To Cloud Firestore - addDoc()

1. Database Object - where we want to add data to
2. Collection - where we want to add data to inside database
3. addDoc() - where we pass collection and db references

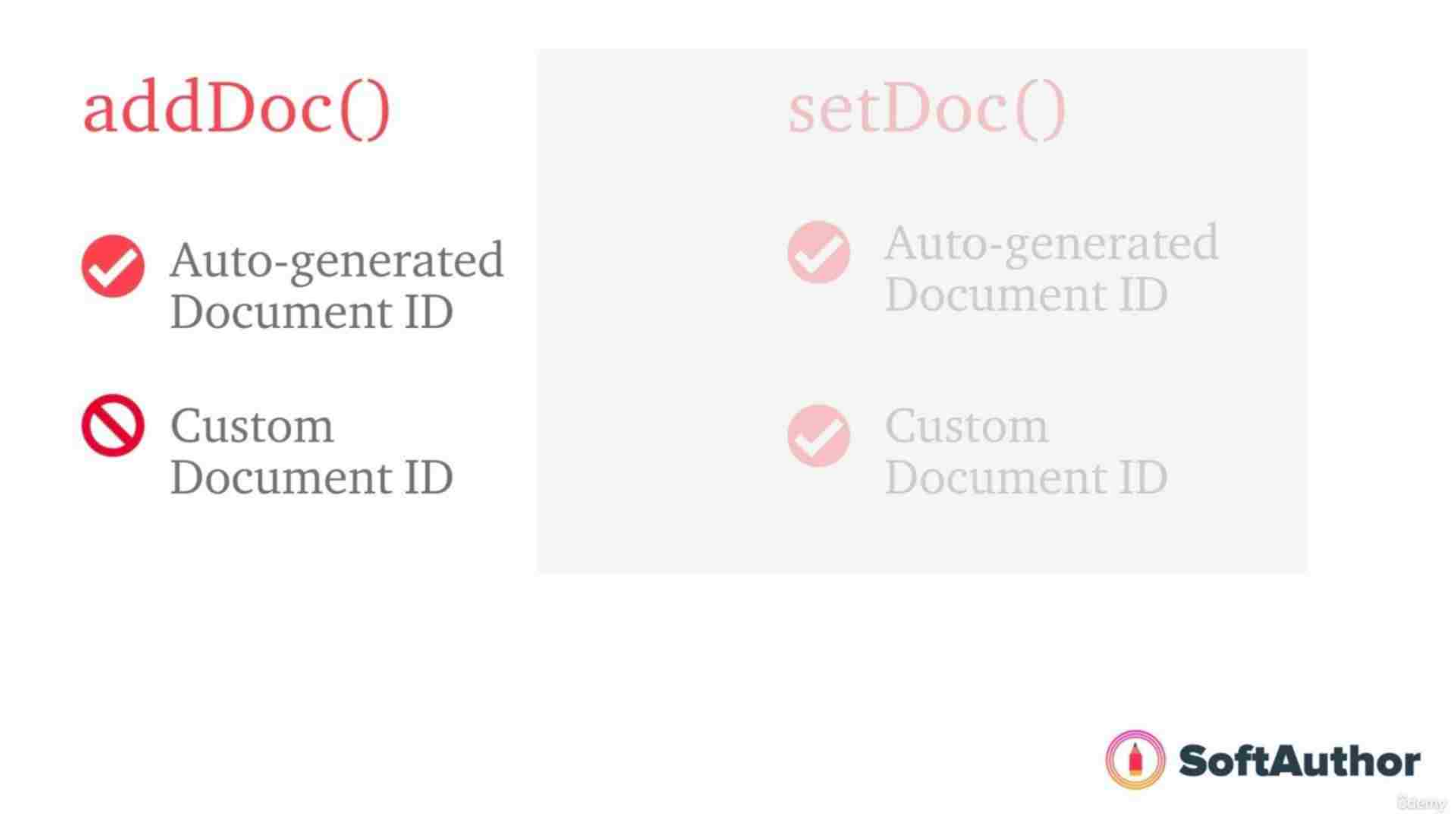
SoftAuthor



Add Data To Cloud Firestore - addDoc()

1. Import Cloud Firestore
2. Initialize Firestore Database
3. Create addDoc() Query
4. Run addDoc() Query
5. Get Auto-generated ID

SoftAuthor



addDoc()

✓ Auto-generated Document ID

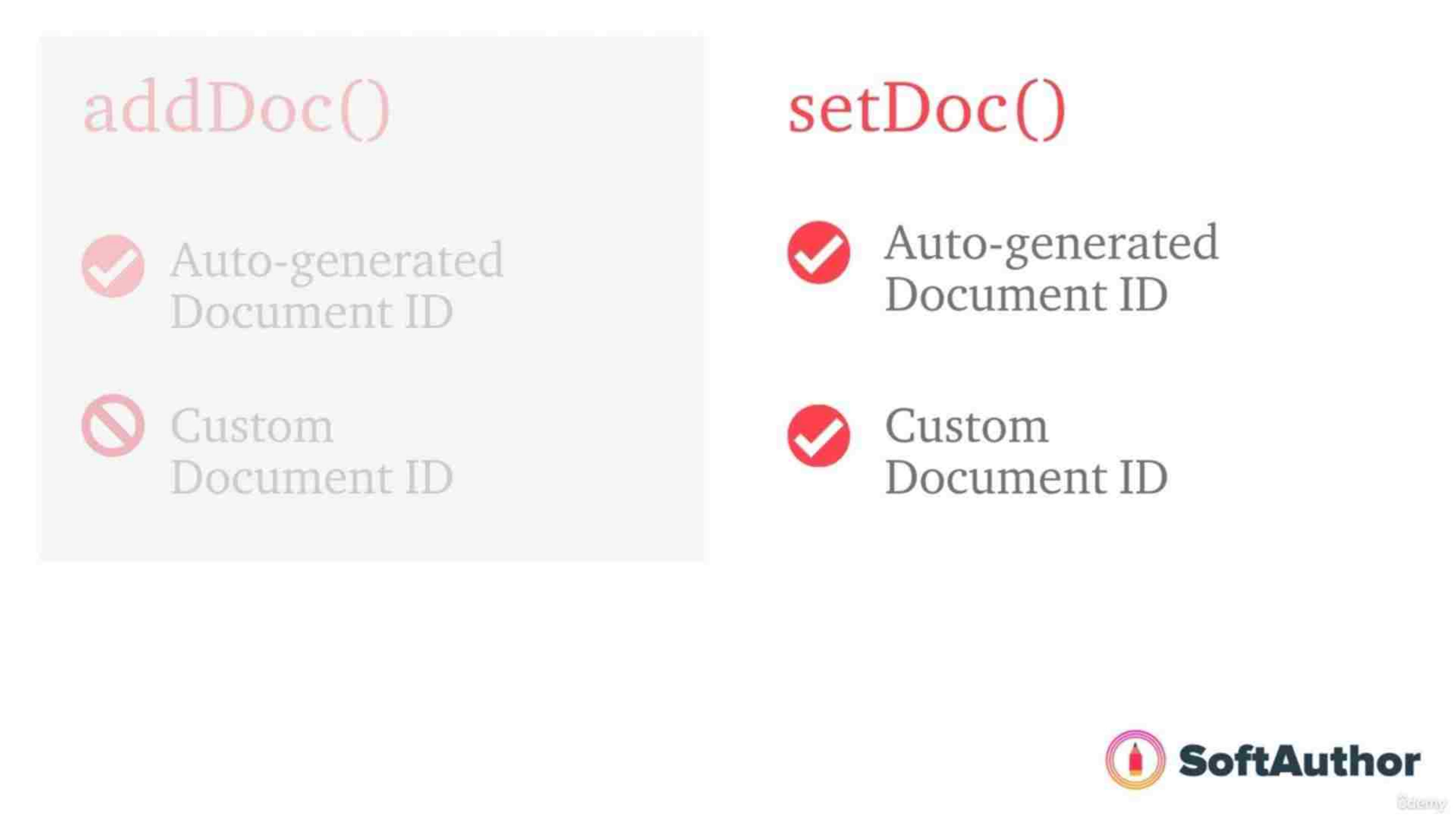
🚫 Custom Document ID

setDoc()

✓ Auto-generated Document ID

✓ Custom Document ID

SoftAuthor

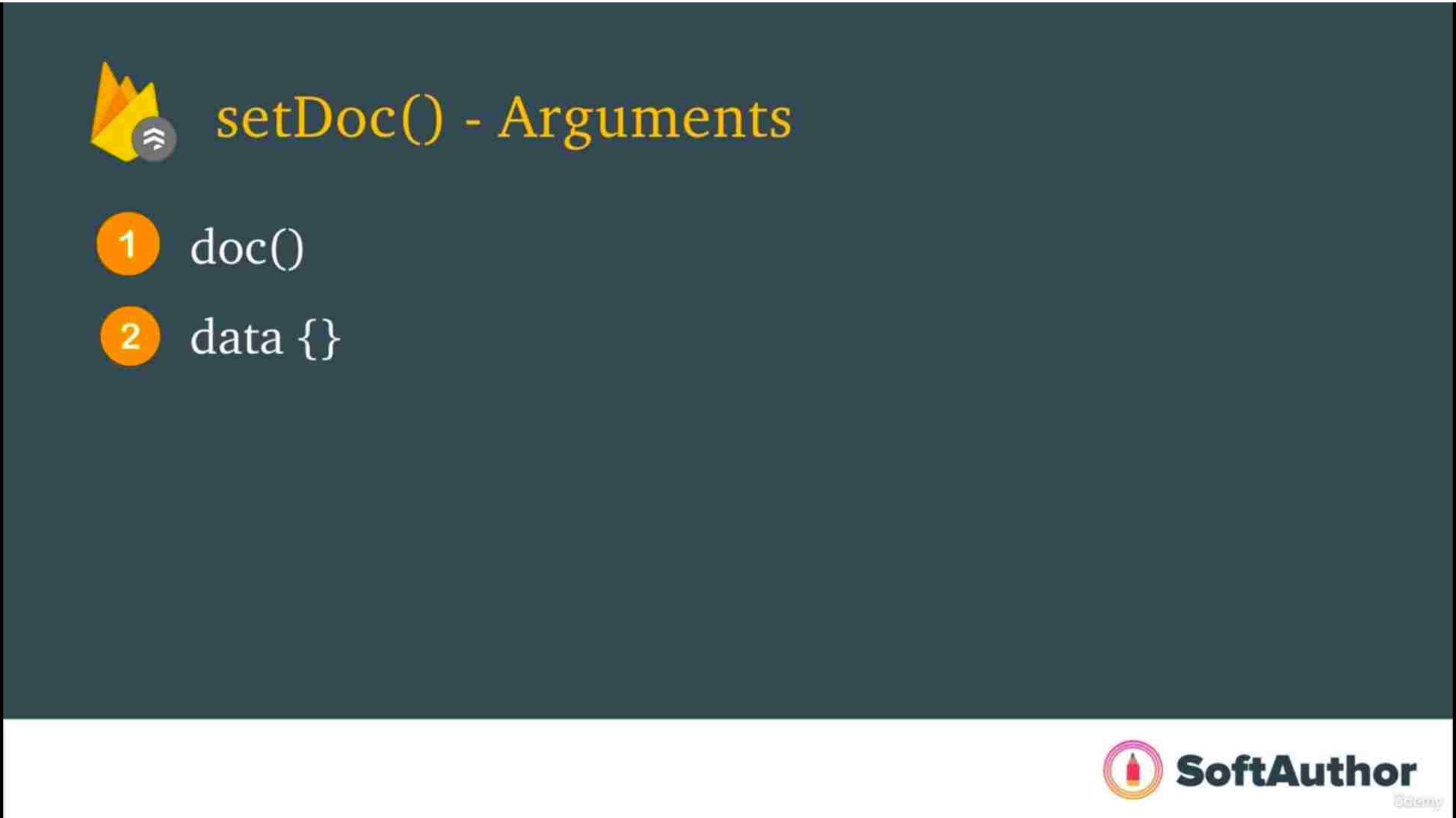


addDoc()

- ✓ Auto-generated Document ID
- 🚫 Custom Document ID

setDoc()

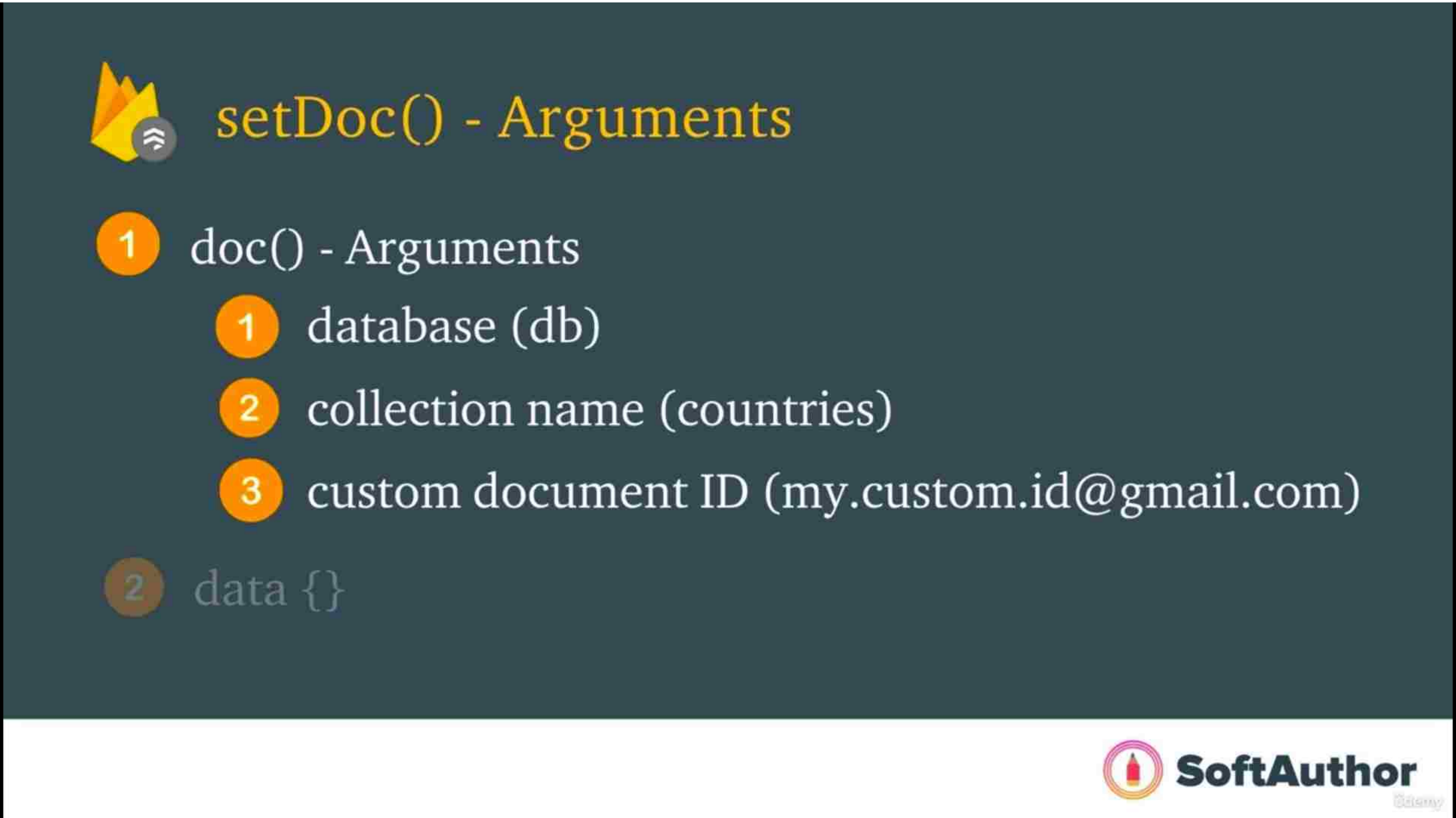
- ✓ Auto-generated Document ID
- ✓ Custom Document ID



setDoc() - Arguments

1. doc()
2. data {}

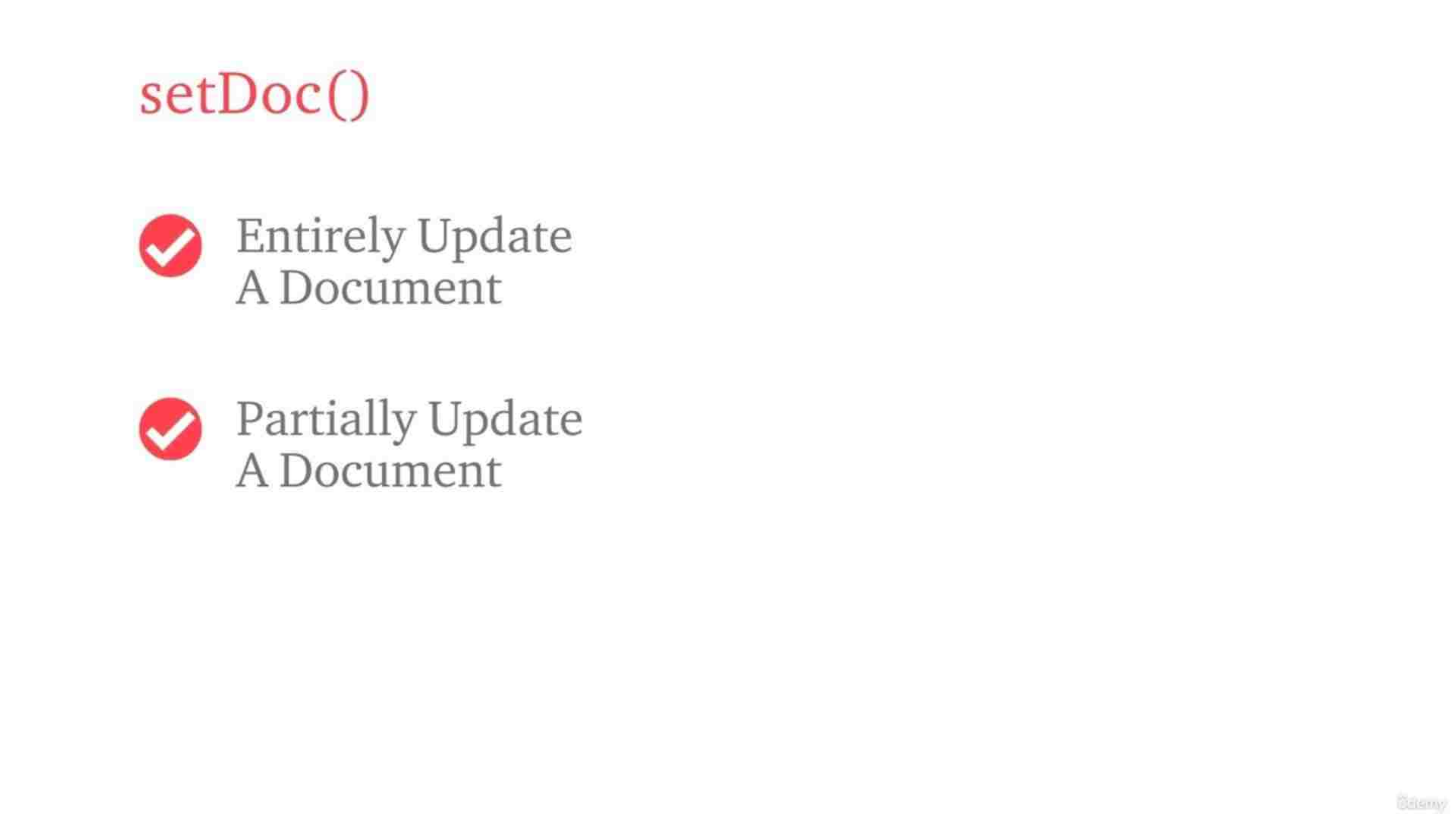
SoftAuthor



setDoc() - Arguments

1. doc() - Arguments
   1. database (db)
   2. collection name (countries)
   3. custom document ID (my.custom.id@gmail.com)
2. data {}

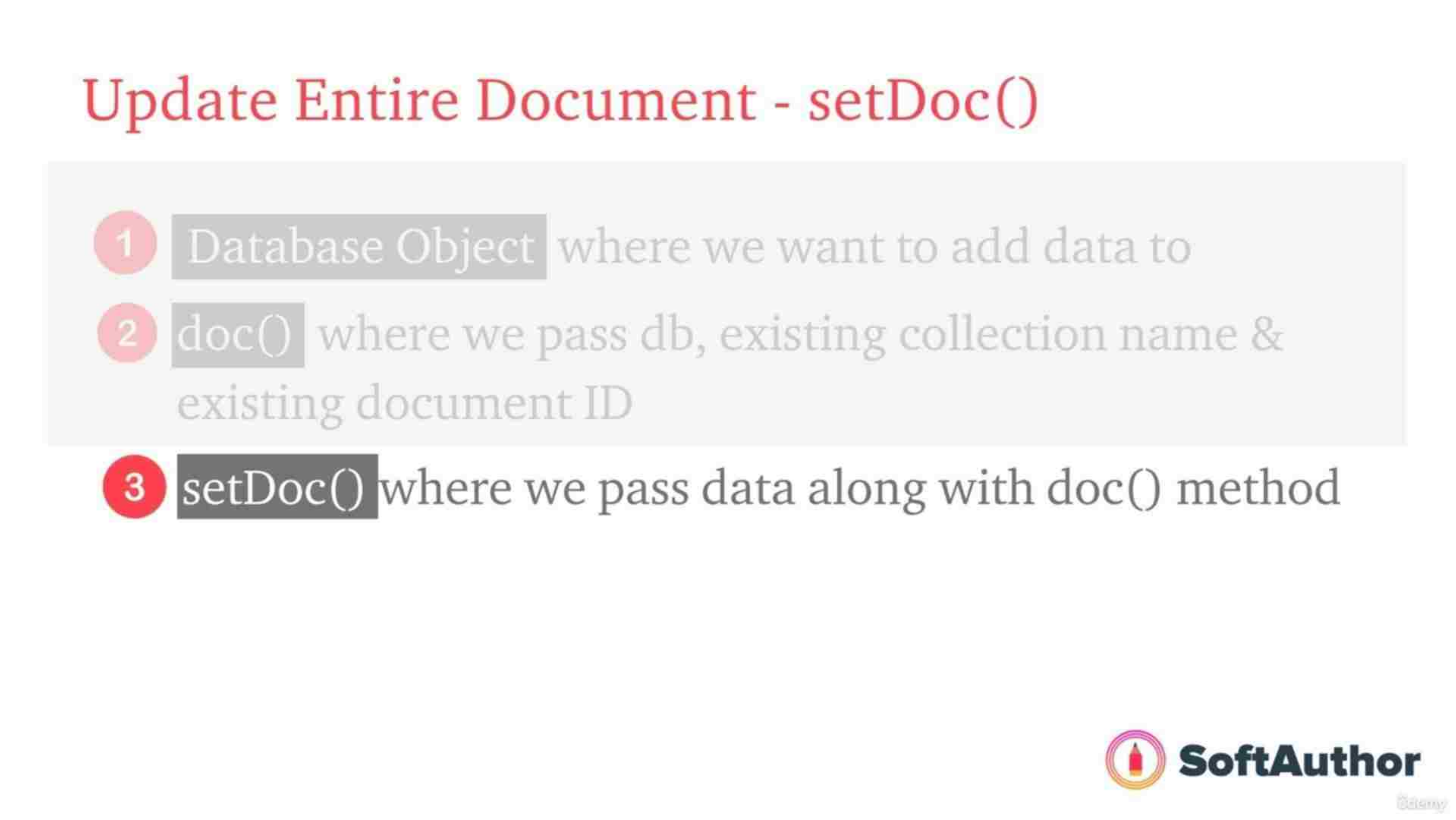
SoftAuthor



setDoc()

✓ Entirely Update A Document

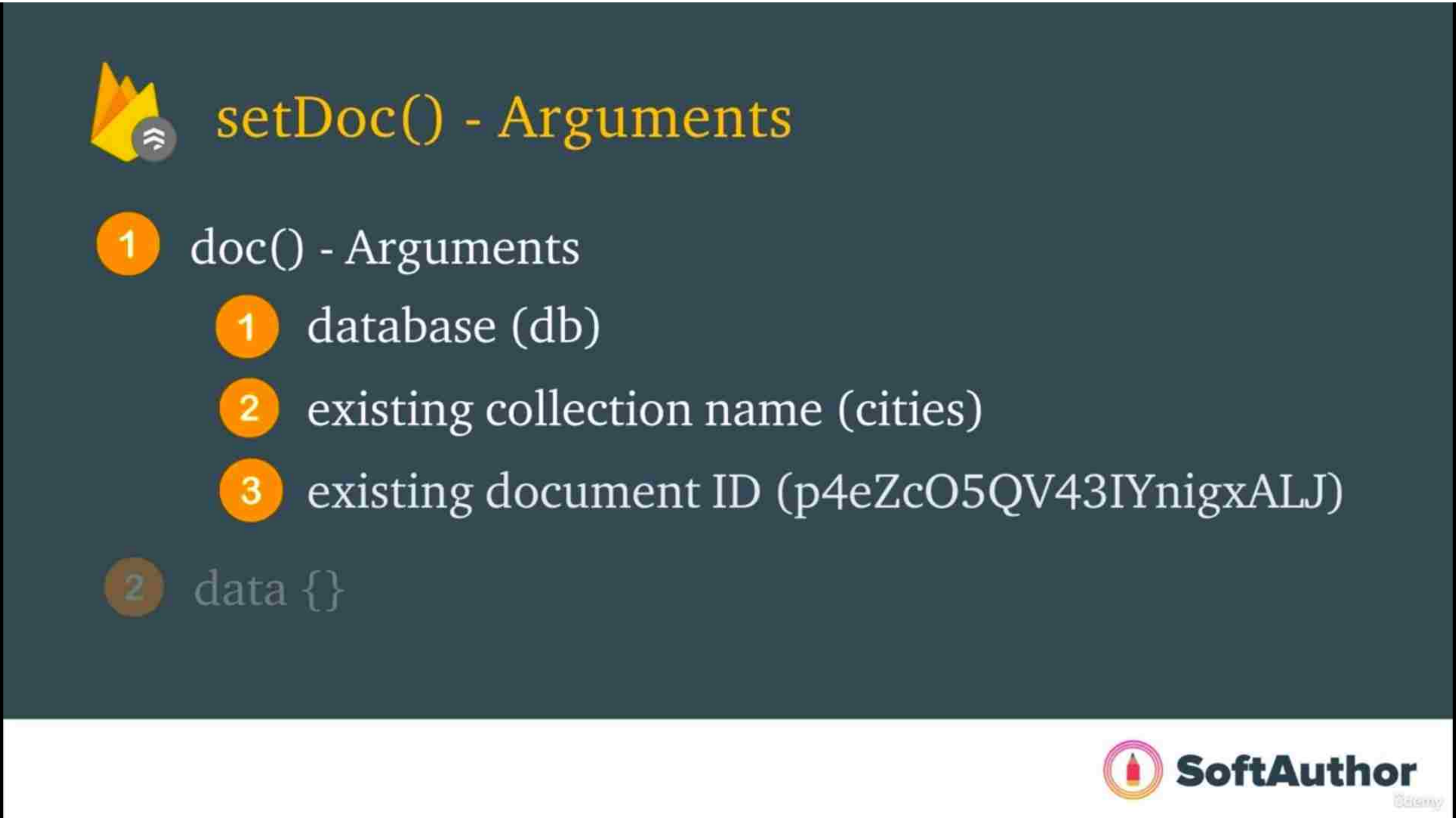
✓ Partially Update A Document



Update Entire Document - setDoc()

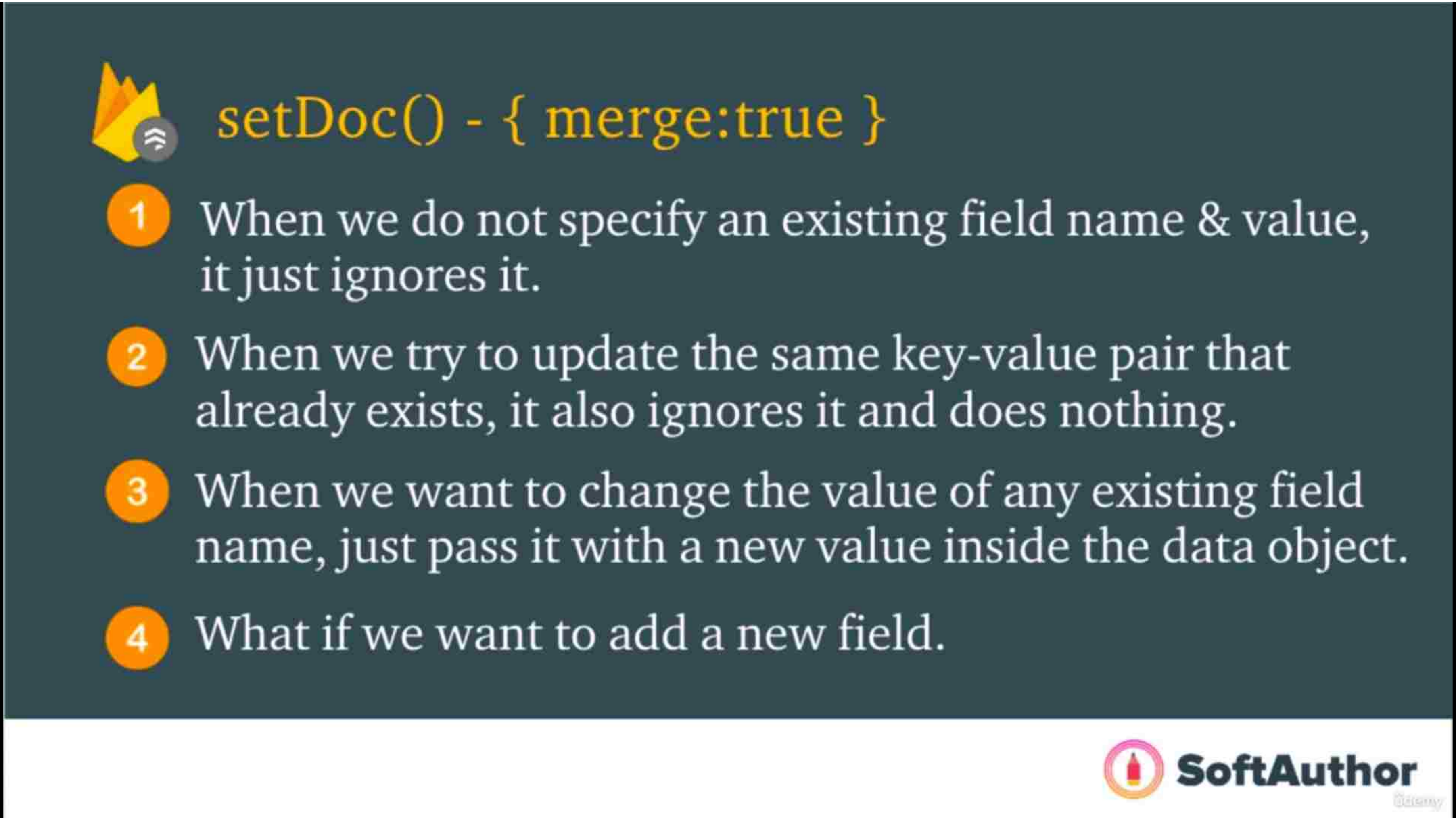
1. Database Object where we want to add data to
2. doc() where we pass db, existing collection name & existing document ID
3. setDoc() where we pass data along with doc() method

SoftAuthor



setDoc() - Arguments

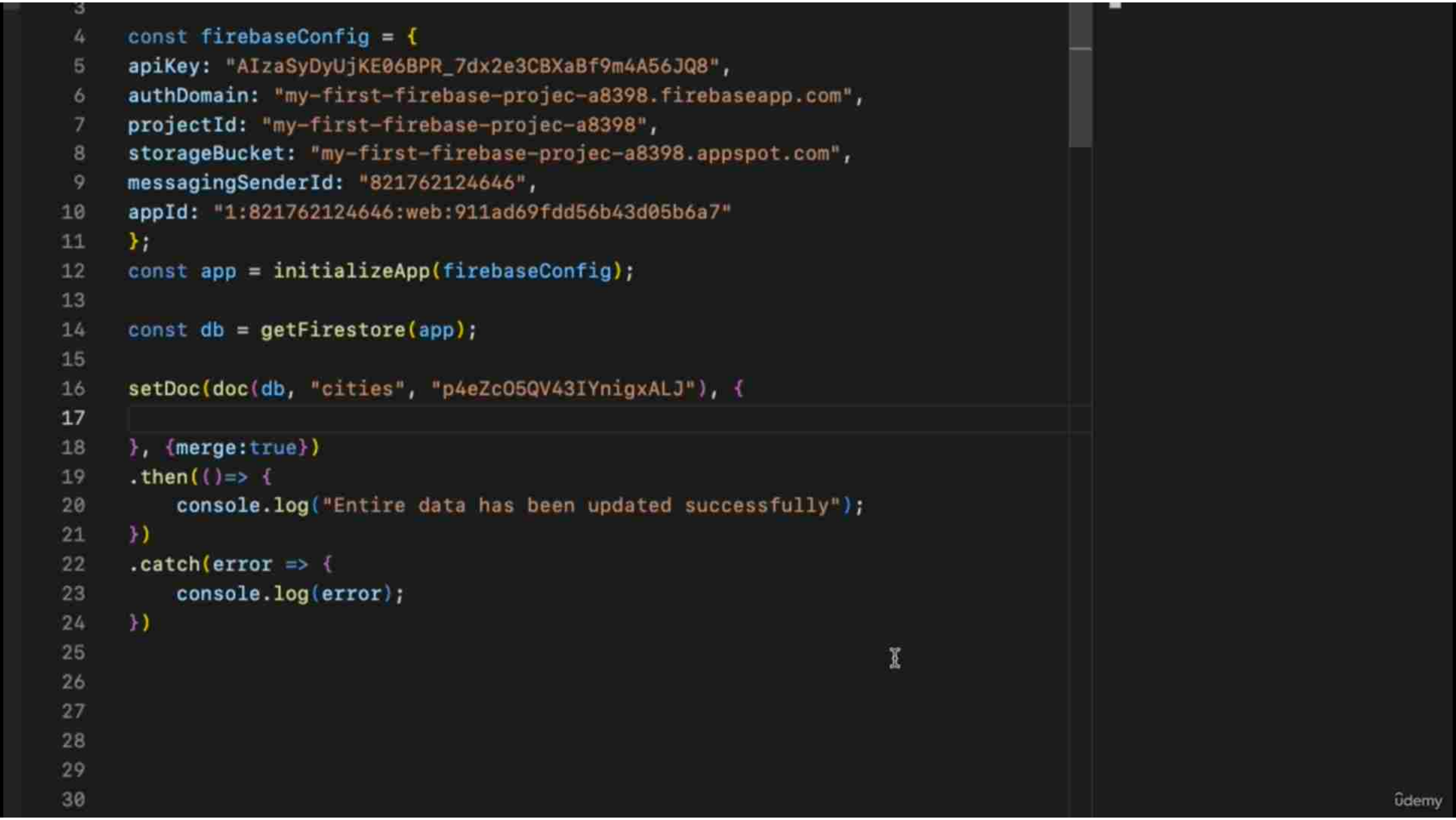
1. doc() - Arguments
   1. database (db)
   2. existing collection name (cities)
   3. existing document ID (p4eZcO5QV43IYnigxALJ)
2. data {}



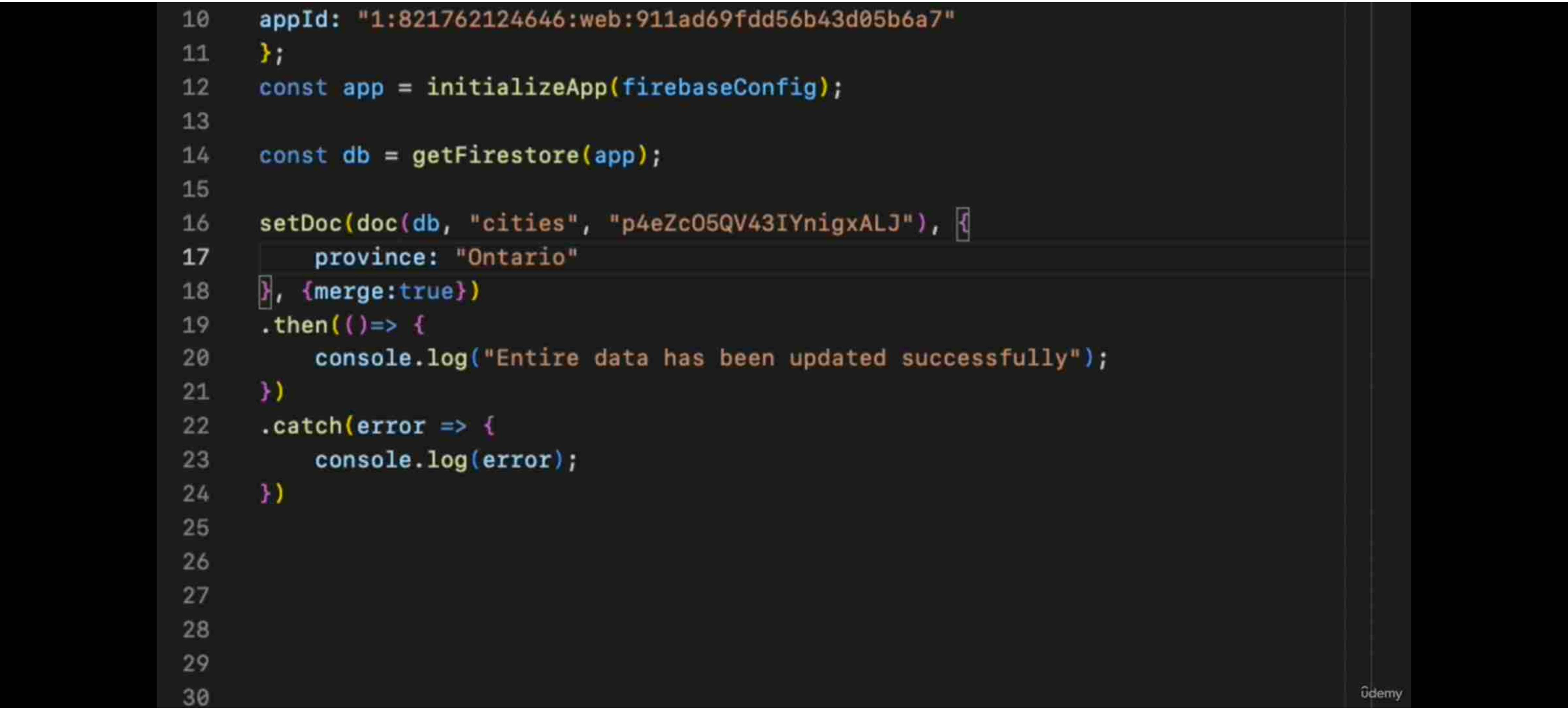
setDoc() - { merge:true }

1. When we do not specify an existing field name & value, it just ignores it.
2. When we try to update the same key-value pair that already exists, it also ignores it and does nothing.
3. When we want to change the value of any existing field name, just pass it with a new value inside the data object.
4. What if we want to add a new field.

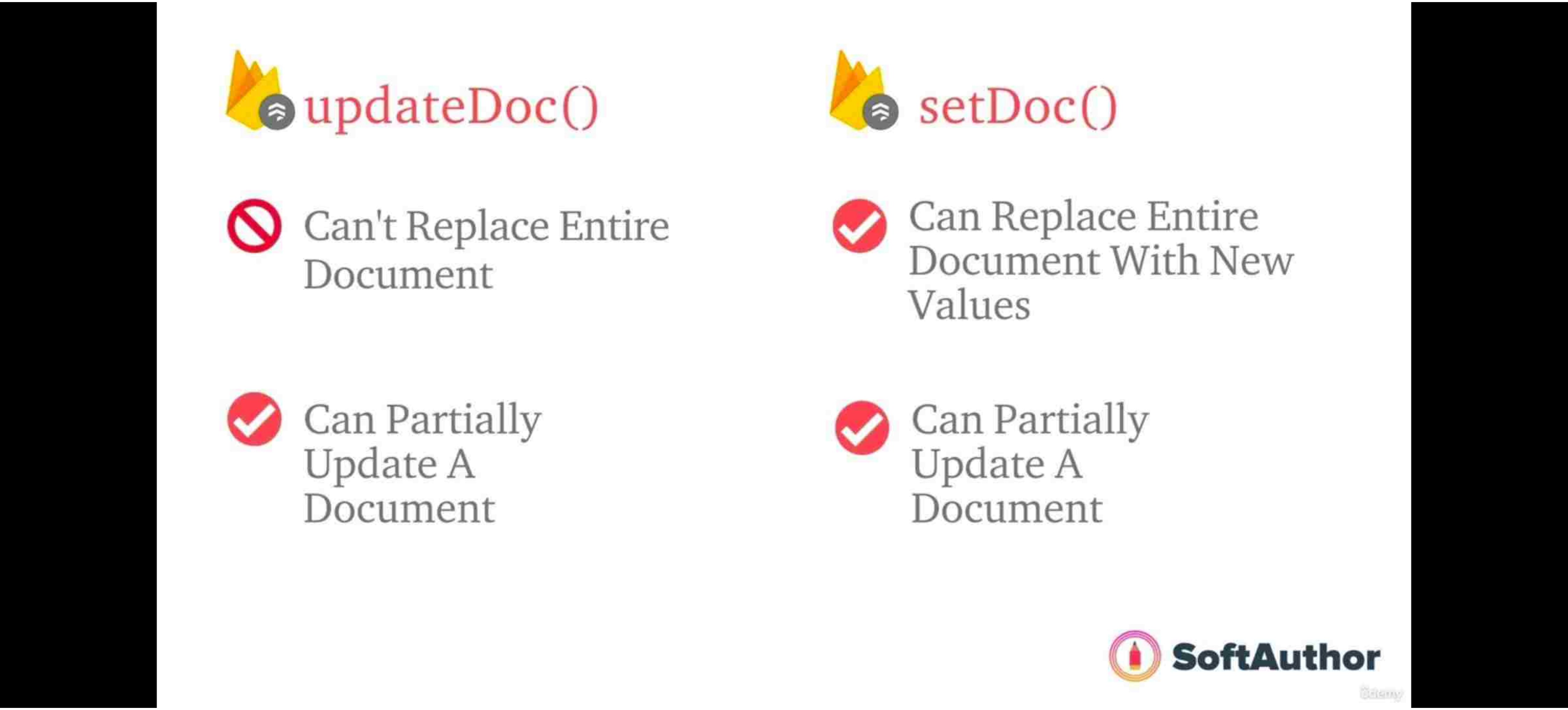
SoftAuthor



```
const firebaseConfig = {
apiKey: "AIzaSyDyUjKE06BPR_7dx2e3CBXaBf9m4A56JQ8",
authDomain: "my-first-firebase-projec-a8398.firebaseapp.com",
projectId: "my-first-firebase-projec-a8398",
storageBucket: "my-first-firebase-projec-a8398.appspot.com",
messagingSenderId: "821762124646",
appId: "1:821762124646:web:911ad69fdd56b43d05b6a7"
};
const app = initializeApp(firebaseConfig);

const db = getFirestore(app);

setDoc(doc(db, "cities", "p4eZcO5QV43IYnigxALJ"), {

}, {merge:true})
.then(()=> {
    console.log("Entire data has been updated successfully");
})
.catch(error => {
    console.log(error);
})
```



```
appId: "1:821762124646:web:911ad69fdd56b43d05b6a7"
};
const app = initializeApp(firebaseConfig);

const db = getFirestore(app);

setDoc(doc(db, "cities", "p4eZcO5QV43IYnigxALJ"), {
    province: "Ontario"
}, {merge:true})
.then(()=> {
    console.log("Entire data has been updated successfully");
})
.catch(error => {
    console.log(error);
})
```



updateDoc()

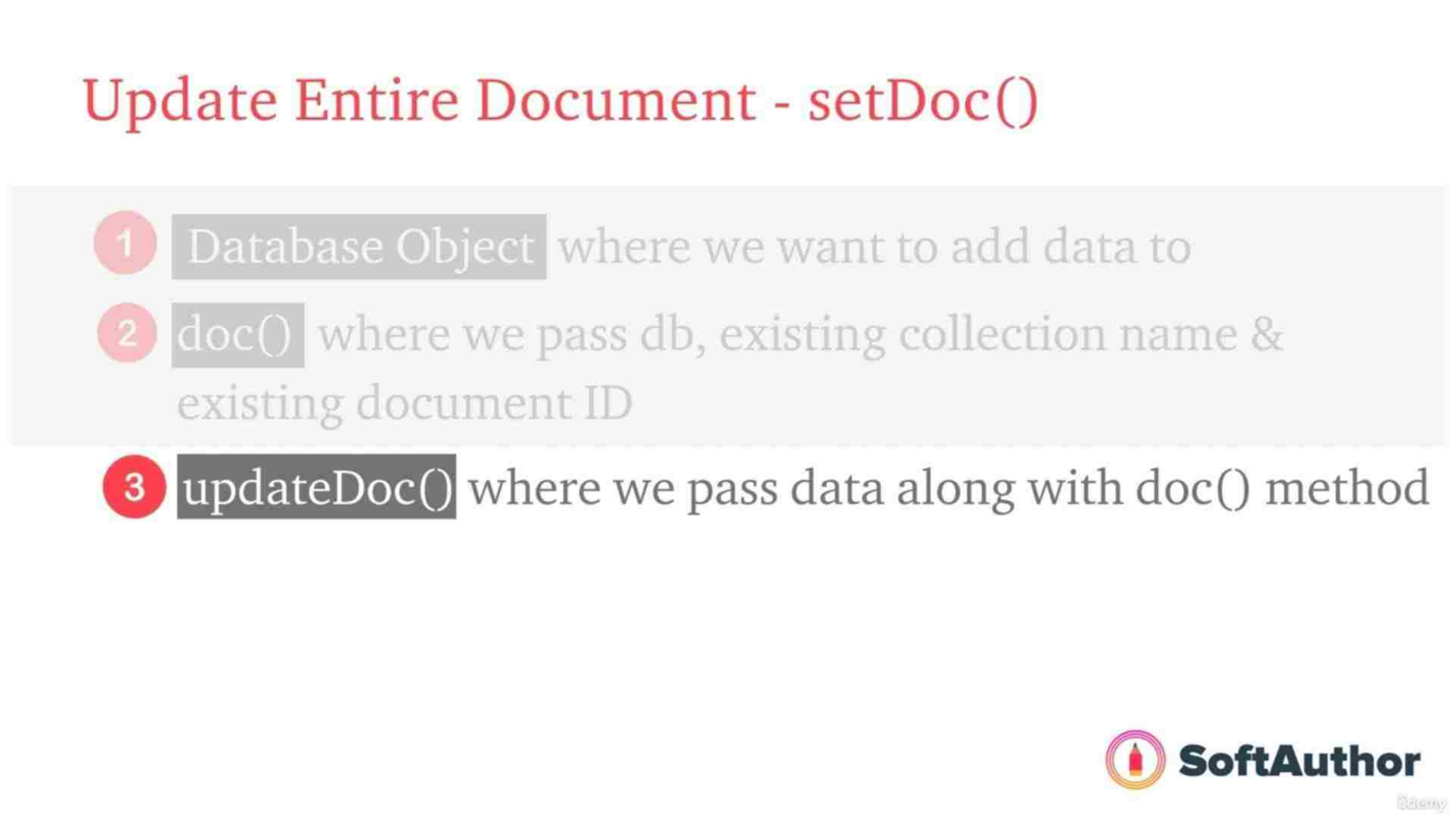
⊘ Can't Replace Entire Document

✓ Can Partially Update A Document

setDoc()

✓ Can Replace Entire Document With New Values

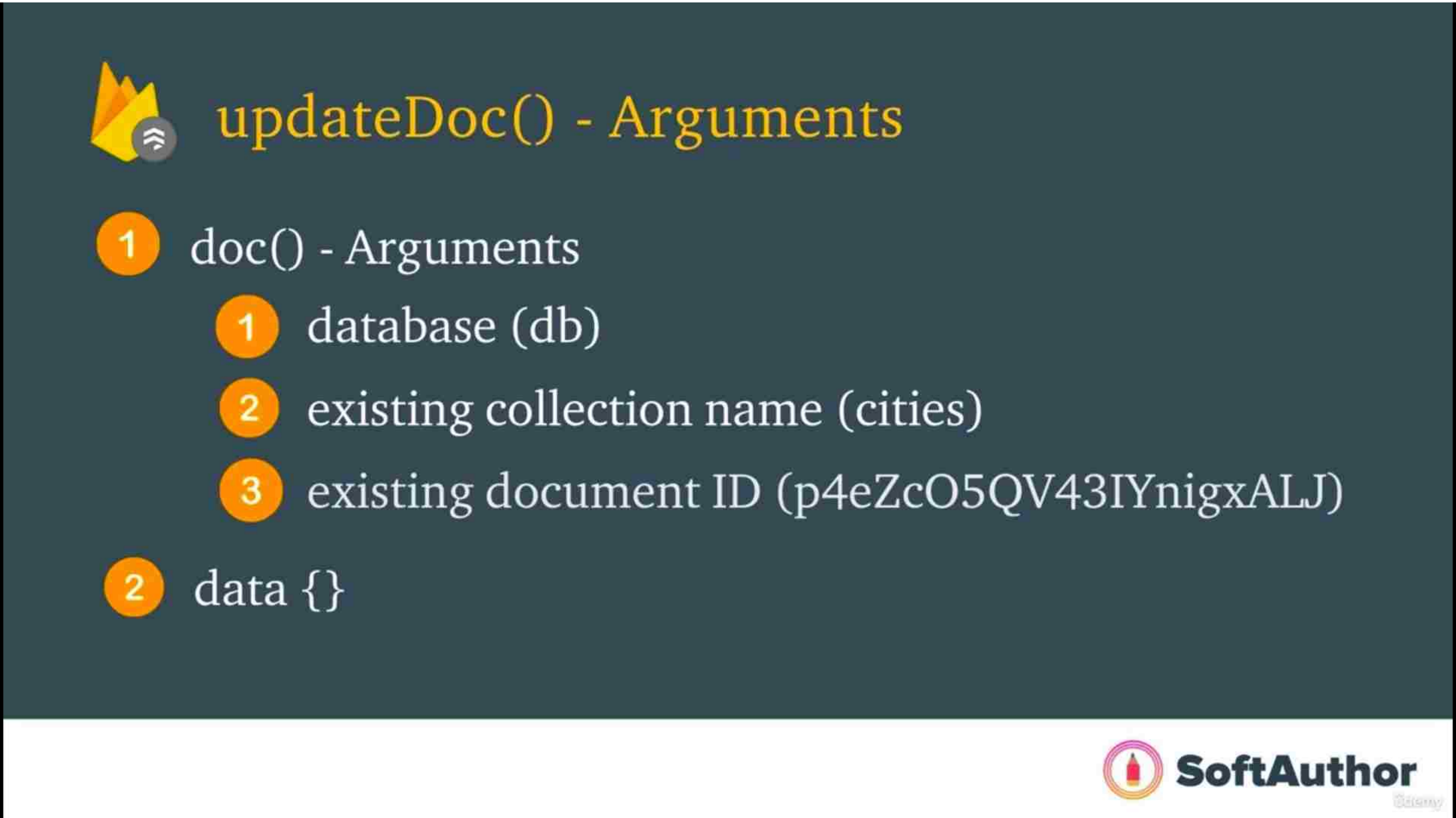
✓ Can Partially Update A Document



Update Entire Document - setDoc()

1. Database Object where we want to add data to
2. doc() where we pass db, existing collection name & existing document ID
3. updateDoc() where we pass data along with doc() method

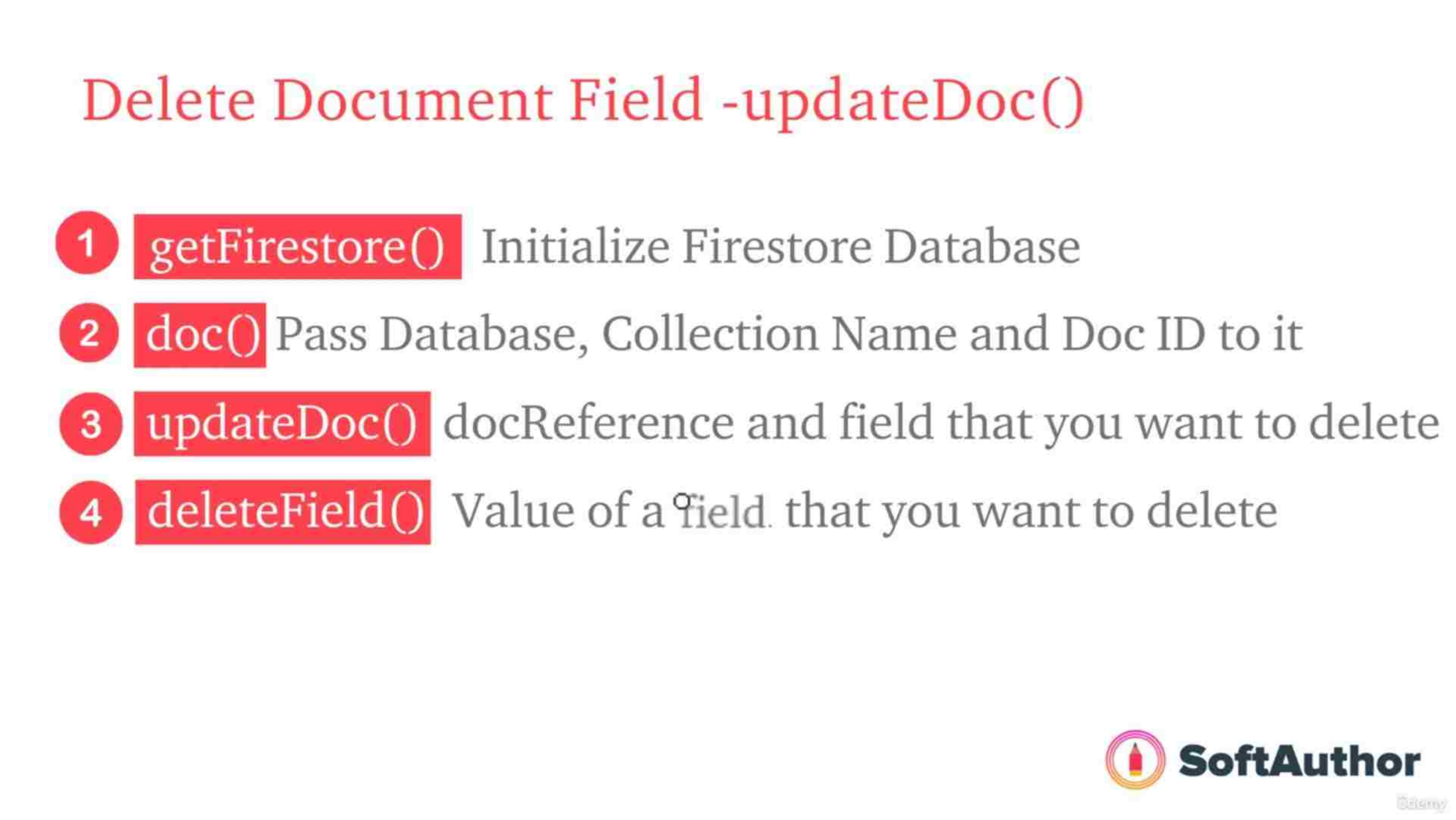
SoftAuthor



updateDoc() - Arguments

1. doc() - Arguments
   1. database (db)
   2. existing collection name (cities)
   3. existing document ID (p4eZcO5QV43IYnigxALJ)
2. data {}

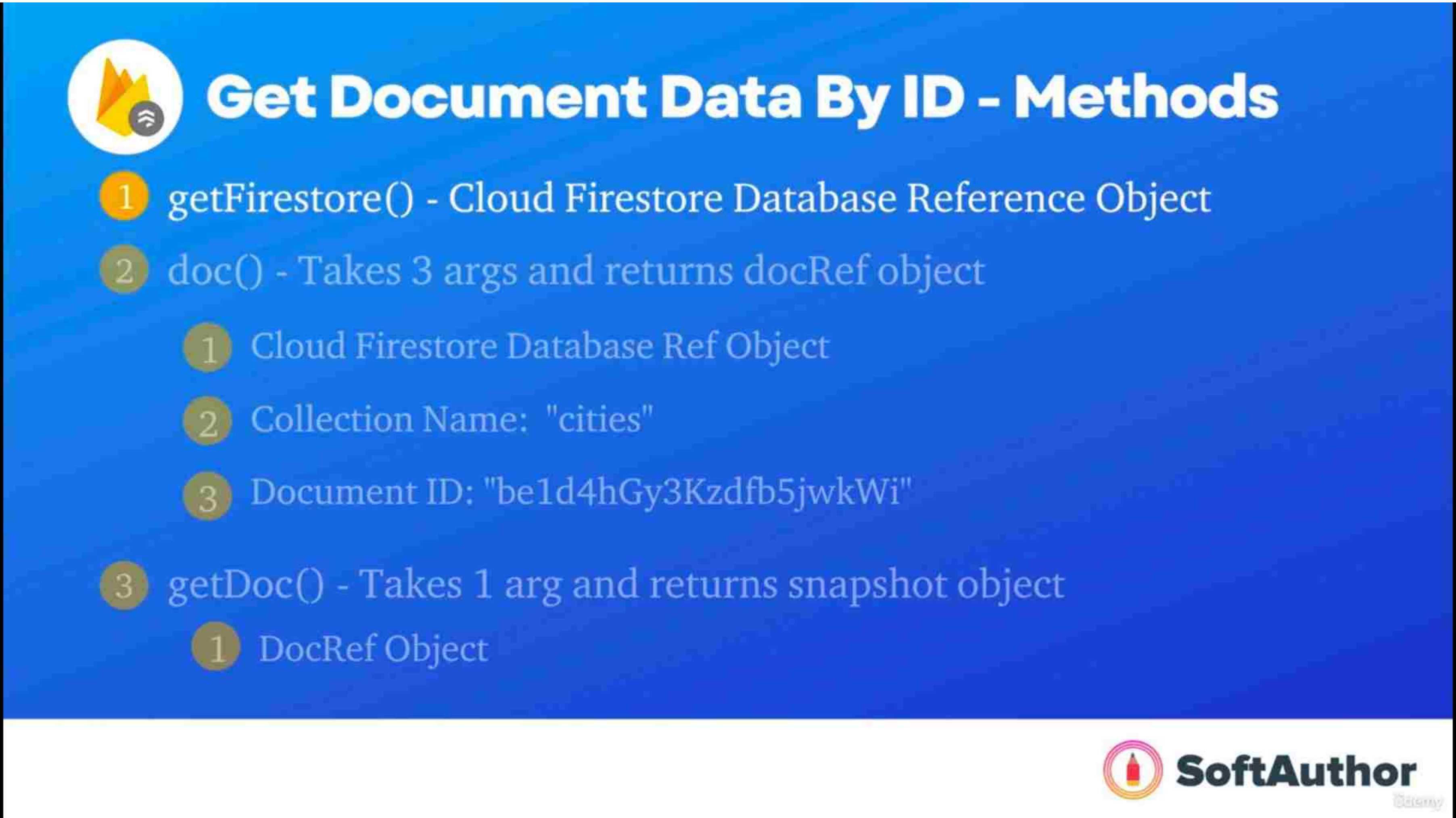
SoftAuthor



Delete Document Field -updateDoc()

1. getFirestore() Initialize Firestore Database
2. doc() Pass Database, Collection Name and Doc ID to it
3. updateDoc() docReference and field that you want to delete
4. deleteField() Value of a field that you want to delete

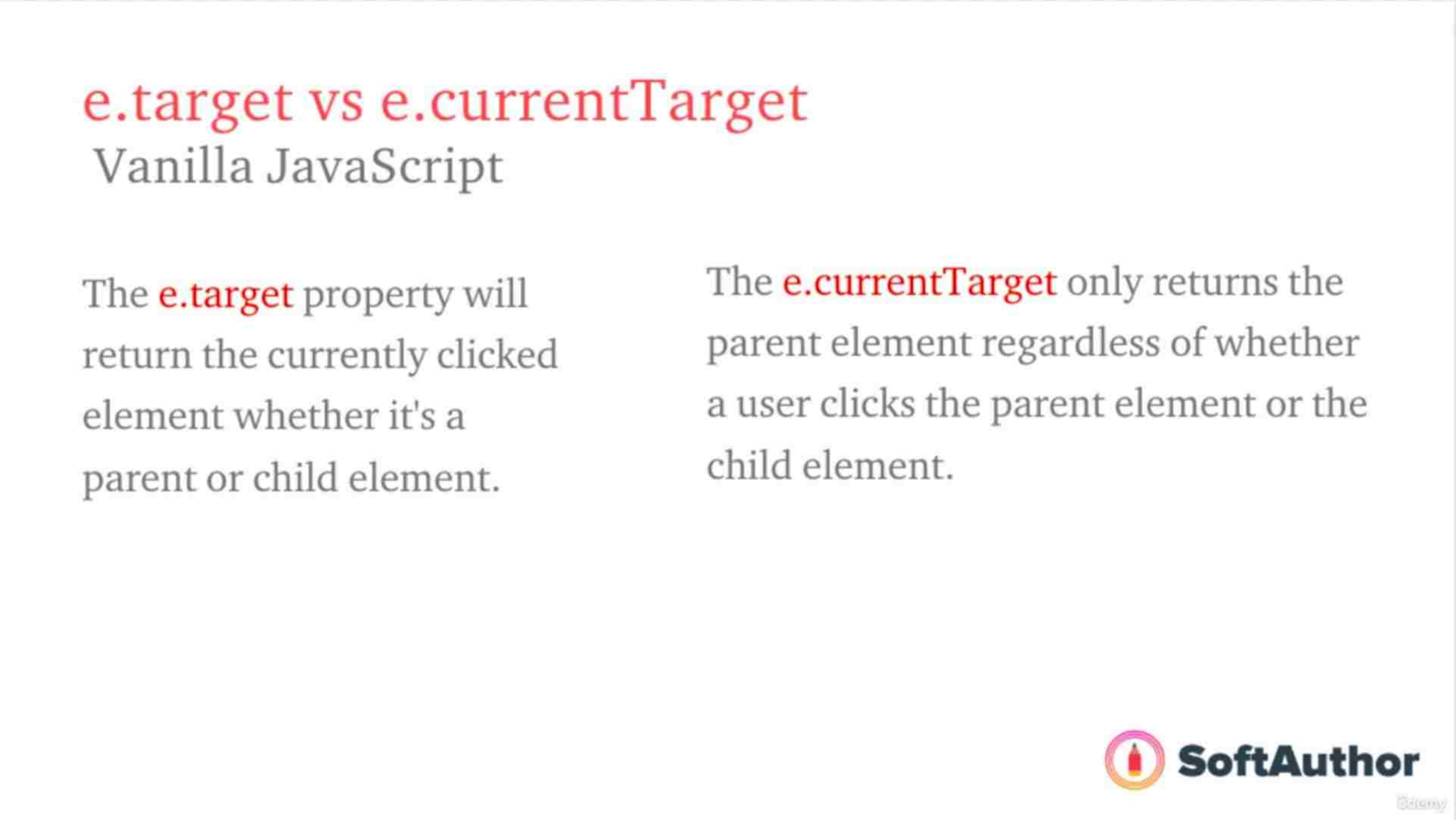
SoftAuthor



Get Document Data By ID - Methods

1. getFirestore() - Cloud Firestore Database Reference Object
2. doc() - Takes 3 args and returns docRef object
   1. Cloud Firestore Database Ref Object
   2. Collection Name: "cities"
   3. Document ID: "be1d4hGy3Kzdfb5jwkWi"
3. getDoc() - Takes 1 arg and returns snapshot object
   1. DocRef Object

SoftAuthor



e.target vs e.currentTarget

Vanilla JavaScript

The e.target property will return the currently clicked element whether it's a parent or child element.

The e.currentTarget only returns the parent element regardless of whether a user clicks the parent element or the child element.

SoftAuthor