

# Using Sembast

---



**Simone Alessandria**

AUTHOR, TRAINER AND PROUD DEVELOPER

[www.softwarehouse.it](http://www.softwarehouse.it)



# sembast

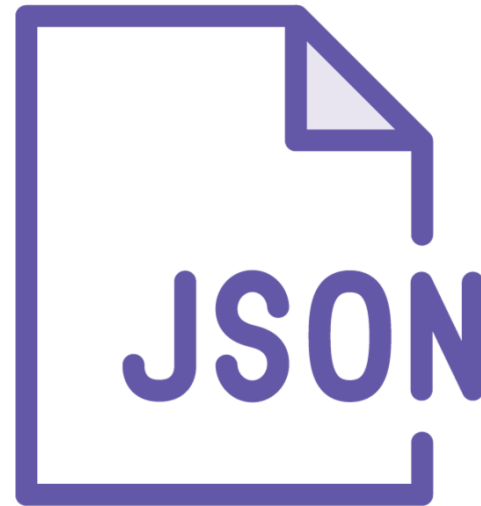
Simple Embedded Application Store



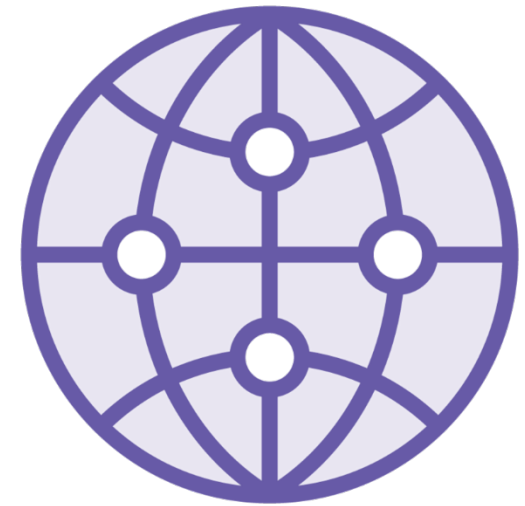
# sembast



**Single File**  
Loaded in memory



**Document Based**  
Data Stored in JSON



**Written in Dart**  
Android, iOS and Web



# Overview



**Adding sembast to the project**

**Using DatabaseFactory and StoreRef**

**Read and Write data with Sembast**

**Encrypting Data**



```
DatabaseFactory dbFactory =  
databaseFactoryIo;
```

```
final db = await  
dbFactory.openDatabase(dbPath);
```

```
final store =  
intMapStoreFactory.store('mystore');
```

```
await store.add(db, myObject.toMap());
```

◀ Use databaseFactoryIo to open Sembast databases

◀ The openDatabase method takes the db Path

◀ A Store is where you save data inside a Database

◀ Add a new document into a store



```
final finder = Finder(  
    sortOrders: [SortOrder('name'),]  
);
```

```
final snapshot = await  
    store.find(_db, finder: finder);
```

```
return snapshot.map((item) {  
    return Password.fromMap(  
        item.value);  
}).toList();
```

◀ Order documents with a Finder instance

◀ Retrieve existing data

◀ Returning a List of Objects



```
final finder = Finder(  
    filter: Filter.byKey(myObject.id));
```

```
await store.update(  
    db,  
    myObject.toMap(),  
    finder: finder);
```

```
await store.delete(db, finder: finder);
```

◀ Filter documents with a Finder instance

◀ Update existing data

◀ Delete data



# Summary



## Sembast

- Create
- Read
- Update
- Delete

## Encrypting Data

