AsyncStorage is a simple, unencrypted, asynchronous, persistent, key-value storage system that is global to the app. It should be used instead of LocalStorage.

It is recommended that you use an abstraction on top of AsyncStorage instead of AsyncStoragedirectly for anything more than light usage since it operates globally.

On iOS, AsyncStorage is backed by native code that stores small values in a serialized dictionary and larger values in separate files.

On Android, AsyncStorage will use either [RocksDB](http://rocksdb.org/) or SQLite based on what is available.

The AsyncStorage JavaScript code is a simple facade that provides a clear JavaScript API, real Errorobjects, and simple non-multi functions. Each method in the API returns a Promise object.

Persisting data:

try {

await AsyncStorage.setItem('@MySuperStore:key', 'I like to save it.');

} catch (error) {

// Error saving data

}

Fetching data:

try {

const value = await AsyncStorage.getItem('@MySuperStore:key');

if (value !== null){

// We have data!!

console.log(value);

}

} catch (error) {

// Error retrieving data

}

Methods

* [getItem](https://facebook.github.io/react-native/docs/asyncstorage.html#getitem)
* [setItem](https://facebook.github.io/react-native/docs/asyncstorage.html#setitem)
* [removeItem](https://facebook.github.io/react-native/docs/asyncstorage.html#removeitem)
* [mergeItem](https://facebook.github.io/react-native/docs/asyncstorage.html#mergeitem)
* [clear](https://facebook.github.io/react-native/docs/asyncstorage.html#clear)
* [getAllKeys](https://facebook.github.io/react-native/docs/asyncstorage.html#getallkeys)
* [flushGetRequests](https://facebook.github.io/react-native/docs/asyncstorage.html#flushgetrequests)
* [multiGet](https://facebook.github.io/react-native/docs/asyncstorage.html#multiget)
* [multiSet](https://facebook.github.io/react-native/docs/asyncstorage.html#multiset)
* [multiRemove](https://facebook.github.io/react-native/docs/asyncstorage.html#multiremove)
* [multiMerge](https://facebook.github.io/react-native/docs/asyncstorage.html#multimerge)