# Paragon  (/)

$\mathsf{Q}$

## paragon.storage.local

Use the paragon.storage.local API to store, retrieve, and track changes to user data.

## Summary

| Methods |
|---|
| get |
| set |
| remove |
| clear |
| getBytesInUse |

## Methods

### get

Returns an object that has the properties corresponding to the keys referenced from the local storage store.

```
paragon.storage.local.get(object keys, function callback)
```

| Parameters | | |
|---|---|---|
| object | keys | String or an array of strings pointing to names in the local storage store |
| function | callback | The *callback* parameter should be a function that looks like this: <br><br> `function(object obj) {...};` |

### set

Adds/Replaces and saves values to the local store.

```
paragon.storage.local.set(object value, function callback)
```

| Parameters | | |
|---|---|---|
| object | value | Value referenced in the store |
| function | callback | The *callback* parameter should be a function that looks like this: <br><br> `function() {...};` |

### remove

Removes the item(s) referenced from the storage.

```
paragon.storage.local.remove(object keys, function callback)
```

| Parameters | | |
| --- | --- | --- |
| object | keys | Item(s) in the storage |
| function | callback | The *callback* parameter should be a function that looks like this:<br><br>```function() {...};``` |

## clear

Deletes local storage file.

```
paragon.storage.local.clear(function callback)
```

| Parameters | | |
| --- | --- | --- |
| function | callback | The *callback* parameter should be a function that looks like this:<br><br>```function() {...};``` |

## getBytesInUse

Throws NotImplementedException

```
paragon.storage.local.getBytesInUse(array of string, function callback)
```

| Parameters | | |
| --- | --- | --- |
| array of strings | keys | |
| function | callback | The *callback* parameter should be a function that looks like this:<br><br>```function(double bytes) {...};``` |