

storageinfo

ON THIS PAGE

- [Object Creation](#)
- [StorageInfo](#)
 - [getDeviceInfo\(\)](#)
- [StorageDeviceInfo](#)
- [Example](#)

The *storageinfo* object provides information about a storage device connected to the player. To retrieve information about the filesystem on the device, use the *filesystem* object.

storageinfo IDL

```
[
  GarbageCollected,
  constructor(DOMString path)
] interface StorageInfo {
  Promise <number> getDeviceSize();
  Promise <StorageDeviceInfo> getDeviceInfo();
}

interface StorageDeviceInfo {
  attribute number size;

  attribute DOMString productName; //@Optional
  attribute DOMString oemID; //@Optional
  attribute number auSize; //@Optional
  attribute DOMString productRev; //@Optional
  attribute DOMString mfrDate; //@Optional
  attribute number mfrID; //@Optional
  attribute DOMString serial; //@Optional
  attribute DOMString signalVoltage; //@Optional
  attribute DOMString uhsMode; //@Optional
  attribute number specVersion; //@Optional
  attribute DOMString speedClass; //@Optional

  attribute DOMString sataModel; //@Optional
  attribute DOMString sataVendor; //@Optional
}
```

Object Creation

To create an *storageinfo* object, first load the `brightsign/storageinfo` module using the `Require()` method. Then create an instance of the *storageinfo* class with a string value specifying the storage path.

```
var StorageInfoClass = require("@brightsign/storageinfo");
var storageInfo = new StorageInfoClass("/storage/sd");
```

Use the following string values to specify different storage drives:

- `"/storage/usb1"` – The drive for USB storage devices connected to the player
- `"/storage/sd"` – The primary SD or microSD drive on the player.
- `"/storage/sd2"` – The internal microSD drive on the player (4Kx42, XDx32 models only)
- `"/storage/ssd"` – The internal SSD on the player (XTx44, XTx43, XDx34, and XDx33 models only)

StorageInfo

Use this interface to retrieve information about the storage device.

`getDeviceSize()`

```
Promise <Number> getDeviceSize
```

Returns the total size of the storage device (in bytes).

The size returned by this method uses a different tool from the `size` parameter in the `StorageDeviceInfo` interface. The values returned using the two processes will be approximately, but not exactly, the same.

`getDeviceInfo()`

```
Promise <StorageDeviceInfo> getDeviceInfo()
```

Returns a `StorageDeviceInfo` interface containing manufacturer information about the storage device.

StorageDeviceInfo

This interface contains manufacturer information about the storage device. Any of the below parameters may be absent from the interface depending on the device type and the information returned by the device:

- `[Number] size`: The total size of the storage device (in bytes)
- `[DOMString] productName`: The product name, as assigned by the device manufacturer (5 bytes for SD, 6 bytes for MMC)
- `[DOMString] oemID`: The two-character card OEM identifier as assigned by the SD Card Association
- `[number] auSize`: The size of the SD AU in bytes
- `[DOMString] productRev`: The product revision assigned by the device manufacturer
- `[DOMString] mfrDate`: The manufacture date reported by the storage device
- `[Number] mfrID`: The card manufacturer ID as assigned by the SD Card Association
- `[DOMString] serial`: The serial number of the storage device
- `[DOMString] signalVoltage`: The signal voltage reported by the SD card
- `[DOMString] uhsMode`: The UHS mode (i.e. BUS speed) reported by the SD card
- `[Number] specVersion`: The version of SD spec to which the card conforms
- `[DOMString] speedClass`: The speed class (if any) declared by the SD card
- `[DOMString] sataModel`: The SATA model reported by the device
- `[DOMString] sataVendor`: The SATA vendor reported by the device

Example

```
var storageClass = require("@brightsign/storageinfo");
var storage = new storageClass("/storage/usbl");

storage.getDeviceSize().then(
  function(data) {
    console.log(JSON.stringify(data));
  })
.catch(
  function(data) {
    console.log(JSON.stringify(data));
  });

storage.getDeviceInfo().then(
  function(data) {
    console.log(JSON.stringify(data));
  })
.catch(
  function(data) {
    console.log(JSON.stringify(data));
  });
```