

```
BookmarkSet.saveBookmark(BookmarkSaveInfo: BookmarkSaveInfo, overwrite?: boolean):
BookmarkInfo
```

Note: This method replaces the method `save()`, which is deprecated.

The `BookmarkSaveInfo` class, which is passed to `saveBookmark()` and contains properties as key-value pairs:

```
class BookmarkSaveInfo {
  name: string,
  isGlobal?: boolean,
  isDefault?: boolean,
  properties?: { [key: string]: string | undefined }
}
```

The `Bookmark` class, which is returned by `saveBookmark()` and contains properties:

```
class BookmarkInfo {
  id: string,
  name: string,
  displayName: string,
  version: integer,
  isGlobal: boolean,
  isDefault?: boolean,
  properties?: { [key: string]: string | undefined }
}
```

The option `isDefault` indicates whether the bookmark is the default bookmark. The default bookmark is loaded when the analytic application's URL contains the URL parameter `bookmarkId=DEFAULT`.

Example:

In the following example, the model ID is saved along with the bookmark. This additional information is used to list all bookmarks which use the model ID `"modelId"`:

```
// Create bookmark properties (key-value pairs)
var oBookmarkProperty1 = { "modelId": "BestRunJuice" };
var oBookmarkProperty2 = { "modelId": "BusinessPlanning" };

// Save bookmark bk1 and bk2 with model ID in properties
BookmarkSet_1.saveBookmark({ name: "bk1", "isGlobal": true, "properties":
oBookmarkProperty1 }, true);
BookmarkSet_1.saveBookmark({ name: "bk2", "isGlobal": true, "properties":
oBookmarkProperty2 }, true);

// List all bookmarks using the "BestRunJuice" model
var aBookmarkInfo = BookmarkSet_1.getAll();
for (var i = 0; i < aBookmarkInfo.length; i++) {
  if (aBookmarkInfo[i].properties) {
    for (var j = 0; j < aBookmarkInfo[i].properties.length; j++) {
      if (aBookmarkInfo[i].properties[j] &&
aBookmarkInfo[i].properties[j]["modelId"] === "BestRunJuice") {
        // print the name of bookmark which uses the "BestRunJuice" model
        console.log(aBookmarkInfo[i].name);
      }
    }
  }
}
```