

MDbmAPI

MaachangComet には、MaachangDBM を操作する API が存在します。

説明する API の内容は、以下の通りです。

- ・ **Image マネージャ関連**

Image マネージャは、MDbm に画像情報を管理に特化した API 群です。画像情報を出力する、簡単な API として、「C.TagAPI」の ImageOpTag を利用します。

- ・ **MDbm 関連**

MDbm は、DBM(データベースマネージャ)です。DBM とは、key,value を高速に、管理、永続化し、トランザクション管理するためのものです。

これらの内容を次ページより、説明していきます。

1. Image マネージャ関連.

Image **タイプ**.JPEG.

Image.TYPE_JPEG :JPEG タイプを表すイメージタイプ.

Image **タイプ**.GIF.

Image.TYPE_GIF :GIF タイプを表すイメージタイプ.

Image **タイプ**.PNG.

Image.TYPE_PNG :PNG タイプを表すイメージタイプ.

Image **タイプ**.BMP.

Image.TYPE_BMP :BMP タイプを表すイメージタイプ.

Image **タイプ**.FLASH.

Image.TYPE_FLASH :FLASH タイプを表すイメージタイプ.

指定名の拡張子から、Image タイプを取得

```
function Image.getType(name)
```

name : 対象のファイル名を設定します.

戻り値 : Image タイプが返されます.

説明 : ファイル名から、イメージタイプを取得するためのメソッド。これらは[Image.xxxImage]関連メソッドを利用すれば、利用する必要はありません.

対象イメージを追加.

function Image.addImage(name, image)

name : 対象のファイル名を設定します.

image : 追加したいイメージバイナリ情報を設定します.

戻り値 : 追加された ID 情報が返されます.

この情報は、たとえば、DB と連携する場合、このメソッドで追加されたときに戻される ID を利用して DB に登録したりします。

<例>

```
var id = Image.addImage( " test.jpg " , image ) ;
```

```
var bean = TestTable.bean() ;
```

```
bean.imageId = id ;
```

```
bean.name = " test.jpg " ;
```

```
TestTable.save( bean ) ;
```

説明 : 対象イメージをファイル名指定で追加します。画像タイプが既にわかっている場合は、このメソッドを利用せず、[Image.add]を利用します。

対象イメージを追加

function Image.add(type, image)

type : 画像タイプを設定します.

image : 追加したいイメージバイナリ情報を設定します.

戻り値 : 追加された ID 情報が返されます.この内容は、前に説明した[Image.addImage]と同様の条件です.

説明 : 対象イメージを画像タイプで選別して追加します。ファイル名を元に指定する場合は、[Image.addImage]メソッドを利用します.

対象イメージを上書き

function Image.setImage(id, name, image)

id : 上書き対象の ID を設定します.

name : 対象のファイル名を設定します.

image : 上書きしたいイメージバイナリを設定します.

説明 : 既に存在する画像イメージを追加します。画像タイプが既にわかっている場合は、このメソッドを利用せず、[Image.set]を利用します。

対象イメージを上書き

```
function Image.set(id,type,image)
```

id : 上書き対象の ID を設定します.

type : 画像タイプを設定します.

image : 上書きしたいイメージバイナリを設定します.

説明 : 既に存在する画像イメージを追加します。ファイル名を元に指定する場合は、[Image.setImage]メソッドを利用します.

指定 ID イメージを削除

```
function Image.remove(id)
```

id : 削除対象の ID を設定します.

説明 : 対象 ID を指定して、Image マネージャから画像情報を削除します.

指定 ID イメージを取得

```
function Image.get(id)
```

id : 取得対象の ID を設定します.

戻り値 : [null]以外の場合、画像情報が取得できています.またこの情報は、java オブジェクトです。利用方法は、下記の通り.

```
var img = Image.get( 1 );  
img.getType() ;// Image タイプ.  
img.getMime() ;// Mime タイプ.  
img.getLastTime() ;// 最終設定時間.  
img.getLength() ;// イメージ長.  
img.getImage() ;// イメージバイナリ.
```

指定イメージ ID が存在するかチェック.

```
function Image.isImage(id)
```

id : チェック対象の ID を設定します.

戻り値 : [true]の場合、指定 ID のイメージ情報は存在します.

2.MDbm 関連.

情報を追加

function MDbm.put(key,value)

key : 設定対象の Key 情報を数値 or 文字列で設定します.

Value : 設定対象の value 情報を設定します.この情報は、JSON 形式で格納されます.

説明 : 情報を追加します。追加された情報は、JSON 形式で格納されます.

情報を削除

function MDbm.remove(key)

key : 削除対象の Key 情報を数値 or 文字列で設定します.

説明 : 情報を削除します.

情報を取得

function MDbm.get(key)

key : 取得対象の Key 情報を数値 or 文字列で設定します.

戻り値 : 格納内容が返されます.[null]の場合、情報は存在しません.

説明 : 情報を取得します.

指定 Key が存在するかチェック.

function MDbm.containsKey(key)

Key : チェック対象の Key 情報を数値 or 文字列で設定します.

戻り値 : [true]の場合、指定 Key の内容は存在します.

説明 : 指定 Key が存在するかチェックする場合に利用します.

シーケンス ID を取得.

`function MDbm.sequenceId(no)`

no : シーケンス発行項番を設定します.指定可能な範囲は[1-63]までです。

戻り値 : 対象シーケンス発行項番に対する、シーケンス ID が返されます.

説明 : この情報は、シーケンス ID を発行するためのメソッドです.また、この情報も永続化されます。
しかし、このメソッドは、トランザクションは利用できません.

格納サイズを取得.

`function MDbm.size()`

戻り値 : 現在 MDbm に格納されている個数が返されます.

説明 : この情報は、Image マネージャ及び、MDbm 全体で格納されている情報数が返されます.