TSNET スクリプト通信

TSC 編集委員会

ISSN 1884-2798

目次

表紙写真: 葉サボテンのスケッチ
制作: jscripter
日時: 2011 年 2 月 6 日
場所: 自宅にて
メモ: 気まぐれに水彩で描いて、スキャナで読み込んだ。裏表紙は左右反転して、Picasa3 で色を反転してヒートマップに変換。

巻頭言 - TSNET あるいはテキスト主義の再発明

jscripter

さて、さらに世界は混迷を深めている。アルジェリアの邦人10名の命が失われた。エジプトの情勢も不安定だ。アフリカや中東はどうなるんだろうと思う。TSNET スクリプト通信の季刊5サイクル2号目の第18号の意義はどこにあるのだろうとさらに考える。5サイクル目のはじまりが5カ月遅れとなり、少し早めにサイクルを回したいと考えたが、自らが息切れ。年末に機械さんからは原稿をいただいていたが、手間取ってしまった。今号の執筆メンバーは、機械伯爵氏と自分だけになった。

今号は、機械伯爵氏の「Python の文法」は第9回、コレクションの「集合」と「組み込み関数」である。おそらく次回で一区切りになるのだろう。

僕自身は「散歩世界プロジェクト」について紹介しようと思う。2005年に「更新日記」で開始したプロジェクトだが、Google Maps の組み合わせを試したところで、中断したというか、さらに発展させるだけの余裕がなかった。データを簡単に登録できる仕組みを作るのが今後の課題だが、今回は Google Maps バージョン3への移行を題材に書いて、紹介に代えたい。これをきっかけに機能を拡張していく予定だ。

前号の巻頭言の最後に「まだまだ、プログラミングの世界は活発に動いている。どこに行き着くのか、見届けたいし、TSNET スクリプト通信を通じて関わっていきたいと考えている。」と書いた。18号の原稿募集時に少し議論した TSNET あるいはテキスト主義を再発明するということはどういうことか考えてみたい。

[日記] 今年の IT、注目すべきは・・・

今朝5時は-1℃、現在9時半が-2℃。昨日も風が冷たい日だった。Apple 決算、iPhone と iPad の販売台数が過去最高で売上高・純利益も過去最高に - ITmedia ニュース、Microsoft 決算、Windows 8 とエンタープライズ製品が好調で増収だが減益 - ITmedia ニュースなどネタ。ITmedia ニュースはタイトルと内容のバランスが取れている。

Windows 8 が発売1ヶ月で市場シェアを3倍に伸ばす - GIGAZINE が使っている Operating system market share では様々な観点から Desktop、Mobile/Tablet の OS やブラウザのシェアを見ることができる。インターネットにアクセスしているデバイスの OS・ブラウザのシェアを調べている。

これからわかることは、Mobile+Tablet を Desktop に合わせても Windows のシェアはまだ 80%はある。アップルは Mac と iOS を合わせて 10%のシェアを獲得したという程度なのだ。Android は 3%に満たない。このシェアの測り方というのも、アクセス回数なのか、滞留 時間なのか、通信量なのか、むずかしい面もあると思うのだが、いずれにせよ、その程度なのだ。

Windows のシェア低下はまだ始まったばかり、Desktop でその存在感はまだ圧倒的だが、現在大きく伸長している Mobile+Tablet ではほぼ0ということが問題なわけで、Surface タブレットの売れ行きが注目される。

- マイクロソフト、Windows RT 版 Surface は「まずまず」の売上げ（バルマー CEO） - Engadget Japanese
- マイクロソフト Surface Windows 8 Pro は北米2月9日発売。10型フル HD に Core i5、デジタイザペン付属 - Engadget Japanese
- マイクロソフト、iOS / Android 版 Office Mobile を来年春にもリリース？ - Engadget

Japanese

おそらく、Mobile+Tablet でシェアを伸ばすためには機能的にはほぼ成熟してきているために価格が最も重要である。したがって、 Windows がこの領域で支配的になる可能性は極めて低い。何もコンピューティングをモバイルでする必然性はほとんどない。Web ブラウジングやマッ プ、メールの送受、SNS、デジカメ、書類の閲覧ぐらい。マイクロソフトのなすべきことは実際 iOS / Android 版 Office Mobile を出すことかもしれない。最早 OS よりはアプリケーションの時代なのだ。

今や、我がデスクトップの Windows 8 Pro は十分なパフォーマンスを示している。なにしろ、クァッドコアに 16GB の RAM、2TB のハードディスクを搭載しているのだから、モバイルデバイ スの不自由な環境でわざわざ何かをしたいとは思わない。起動終動もあっという間で XP 時代の終わりのイライラする不自由さは雲散霧消してしまった。

Google は Web ブラウザですべてが処理できるような環境を生み出す方向で動いている。Chrome/Chromium だが、まだ目に見えて大きな動きにはなっていない。HTML5 のアプリケーションの登場が今後の動向を決めるだろう。

Apple は iOS で成功したが、次の一手が難しい。iOS の延長線上に未来が見えているだろうか。テレビなどを iOS/Mac OS X の新たなデバイスとして登場させることができるかどうかが今後の展開のキーになるように見える。無論、新興国で iPhone/iPad を普及させること ができるかどうかも課題なのだろう。ここには微妙な問題がある。現在が高収益であるだけに利幅の薄い低価格路線にいかに移ることができるかが課題になる。

スマートフォンの低価格路線に載る OS として、Firefox OS や Ubuntu for Phones が名乗りを上げているが、HTML5 の潮流にマッチしているのが、Firefox OS だろう。既存の iOS/Android の強敵になる可能性を秘めている。

Amazon が、売れるとみれば Firefox OS 搭載機を出してくる可能性も極めて高いのではないか。デバイスの OS に拘る必要はまったくない立場にある。

- Announcing the Firefox OS Developer Preview Phone! ? Mozilla Hacks -- the Web developer blog

混迷が深まるデスクトップ OS、モバイル OS、ブラウザ OS の戦い。優位に立ったと思われた iOS/Android が HTML5 ブラウザに追われる展開が見えてきた。我々スクリプタは HTML5 に向かう。

AMI さんも言われたように、Web はテキストから成り立っている。HTML/XML にしても、CSS にしても、JavaScript にしても。「MS-DOS テキストデータ料理学 sed、awk のある UNIX 流パソコン環境」(翔泳社、1992 年)が出た当時よりも、テキスト主義は現在に似つかわしい。それどころかは今はテキストの時代である。データだけでなく、スクリプトもプログラムコードもテキストなのだから。わざわざテキスト主義を標榜する必要さえなくなった。スクリプタは当然のように新たなテキストを生成している。

我々が目指すべきは、HTML5+CSS3+JavaScript ということになる。そこで、少し不安なのは、Perl、Python、Ruby などのスクリプティング言語はどうなるのということだが、それは少しおいておいて、HTML5+CSS3+JavaScript で何ができるのだろうということがまずは重要である。いずれにせよ、テキストなのだから、必要であれば自在に取り扱えるはずである。

最近、HTML5 に関して注目したのは次の記事。

モジラ、「Firefox OS」でアプリの扱いを変更 (CNET Japan) - Yahoo!ニュース BUSINESS <http://newsbiz.yahoo.co.jp/detail?a=20130109-35026658-cnetj-nb>

「Mozilla は、新興企業 Everything.me への投資の一環で、マニフェストファイルを利用してウェブサイトをモバイルアプリに組み込む方法を開発した。」とある。

マニフェストは元々オフラインで Web を読むためにキャッシュに必要なコンテンツ・ファイルを指定しておくためのものだ。オライリーの「JavaScript 第 6 版」、「20.4 アプリケーションストレージとオフライン Web アプリケーション」(661 ページ) 、「20.4.1 アプリケーションキャッシュマニフェスト」に詳しい。様々なデバイスからオフライン Web アプリケーションを使う場合はいろいろとややこしい。オンラインになった場合にサーバーと通信してデバイス間の同期も取れるようにする必要があるだろう。サーバーは一種のクラウドである必要もあると考えられる。Firefox Sync のような仕組みが必要になる。

さて、HTML5 の大きな可能性の一端を感じ取れただろうか。クライアント-サーバーの仕組みは工夫次第で大きな可能性があると感じていたが、それが HTML5 で実体化しつつある。

TSNET を HTML5 を基盤として再発明できないかと考えている。その方法は・・・

(投稿日 2013 年 1 月 28 日)

Pythonの文法

TSNETスクリプト通信版　草稿　第９回

6-2-2-2-5 集合型(set type)

集合型オブジェクトは、写像型オブジェクトと同じく順序を持たないコレクションです。
しかし写像型オブジェクトと異なり、集合型オブジェクトには、オブジェクトにアクセスするインデックスすらありません。
集合型オブジェクトの目的は、文字通り集合的検索と集合演算です。
集合というオブジェクトのグループをつくり、あるオブジェクトがそのグループに所属しているかどうかを調べるメンバーテストの他、グループ同士の交叉（交わり、和集合）、結合（結び、積集合）などを調べる集合演算が、集合オブジェクトを用いれば簡単にできます。
集合型オブジェクトには、可変コレクションである'set'クラスのオブジェクト（以下「集合オブジェクト」）と、不変コレクションである'frozenset'クラスのオブジェクト（以下「凍結集合オブジェクト」）があります。
集合型は他クラスと異なり、要素の重複が認められません。
また、可変オブジェクトは、集合型の要素には出来ません。

※'set'クラスが登場する以前は、辞書クラスのインデックスを集合の代用としていました。

集合オブジェクトのリテラルは、ブレース括弧に要素を並べます。

```
{2, 4, 6}
```

ただし空集合に関しては、空の辞書で"{}"が使用されてしまっているため、"set()"となります。
凍結集合オブジェクトは、さらに不恰好になります。

```
frozenset()
frozenset({2, 4, 6})
```

集合型は、他のコレクションのように内包表記が可能ですが、重複する要素は整理されます。

集合型オブジェクトは、"&","|","^"の演算子を用いて「交叉"∩"」「結合"∪"」「対称差"△"」の演算が可能です。
また、"in"演算子を用いてメンバーテストが可能です。
以下に、集合／凍結集合オブジェクトで使用可能な演算子と演算子対応メソッドを列挙します。

●演算子に対応した特殊メソッド（集合に特化したもののみ列挙）

集合演算	記号	コード	対応特殊メソッド
交わり、交叉、積(集合)	A ∩ B	A & B	__and__(A, B), __rand__(B, A)
結び、結合、和(集合)	A ∪ B	A | B	__or__(A, B), __ror__(B, A)
差(集合)、相対補	A ＼ B	A - B	__sub__(A, B), __rsub__(B, A)
対称差	A △ B	A ^ B	__xor__(A, B), __rxor__(B, A)

※集合オブジェクトにあって凍結集合オブジェクトに無い'__iand__','__ior__','__isub__','__ixor__'は、演算結果を代入する演算代入子'&=','|=','-=','^='に対応します

```
>>> s0={x for x in range(5)}
>>> s1={x for x in range(0,10,2)}
>>> s1
{0, 8, 2, 4, 6}
>>> s0
{0, 1, 2, 3, 4}
>>> s0 & s1
{0, 2, 4}
>>> s0 | s1
{0, 1, 2, 3, 4, 6, 8}
>>> s0 ^ s1
{1, 3, 6, 8}
>>> s0 - s1
{1, 3}
```

【補足】集合演算について
集合Ａを｛０，１，２，３，４｝、集合Ｂを｛０，２，４，６，８｝とするとき、
Ａ ∩ Ｂ ＝ ｛０，２，４｝
Ａ ∪ Ｂ ＝ ｛０，１，２，３，４，６，８｝
Ａ ＼ Ｂ ＝ ｛１，３｝
Ａ △ Ｂ ＝ ｛１，３，６，８｝

6-2-2-2-5-1 凍結集合(frozenset)

■クラス名：'frozenset'

凍結集合(frozenset)クラスは、集合(set)クラスの不変オブジェクト版です。
凍結集合オブジェクトのメソッドは、全て集合オブジェクトに含まれます（要するに凍結集合オブジェクトは、集合オブジェクトの要素変更メソッドを除いたサブセット(!?)版です）
ですから、ここで紹介するメソッド群は、全て集合メソッドにも実装されています。

●通常のメソッド

■凍結集合のコピー(copy items)

書式：FS.copy() ⇒ newFS
動作：凍結集合"FS"の内容をコピーした新しい凍結集合"newFS"を返す

```
>>> fs0
frozenset({0, 1, 2, 3, 4})
>>> fs0.copy()
frozenset({0, 1, 2, 3, 4})
```

●集合演算用メソッド

以下のメソッドは、集合型（集合・凍結集合）であれば、演算子を用いて行える演算を、他のコレクションとの間でも行えるようにしたものです。結果に得られる集合は、集合'set'なら集合、凍結集合'frozenset'なら凍結集合です。

■積集合の作成

書式：FS.intersection(C) ⇒ newFS
動作：凍結集合"FS"と、コレクション"C"との積集合"newFS"を作成する

■和集合の作成

書式：FS.union(C) ⇒ newFS

動作：凍結集合"FS"と、コレクション"C"との和集合"newFS"を作成する

■差集合の作成

書式：FS.difference(C) ⇒ newFS

動作：凍結集合"FS"と、コレクション"C"との差分"newFS"を作成する

■対称差の作成

書式：FS.symmetric_difference(C) ⇒ newFS

動作：凍結集合"FS"と、コレクション"C"との対称差"newFS"を作成する

```
>>> FS
frozenset({0, 1, 2, 3, 4})
>>> C
[0, 2, 4, 6, 8]
>>> FS.intersection(C)
frozenset({0, 2, 4})
>>> FS.union(C)
frozenset({0, 1, 2, 3, 4, 6, 8})
>>> FS.difference(C)
frozenset({1, 3})
>>> FS.symmetric_difference(C)
frozenset({1, 3, 6, 8})
```

●関係の判定

こちらも、相手のコレクションを問いません

■「互いに素」であるかどうかのテスト

書式：FS.isdisjoint(C) ⇒ bool

動作：凍結集合"FS"とコレクション"C"とが、互いに素（共通の要素を持たない状態）であるかどうかを判断する。互いに素ならば真("True")を返す

■部分集合であるかどうかのテスト

書式：FS.issubset(C) ⇒ bool

動作：凍結集合"FS"が、コレクション"C"の部分集合(サブセット)であるかどうかを判断する。部分集合であれば真("True")を返す

■上位集合であるかどうかのテスト

書式：FS.issuperset(C) ⇒ bool

動作：凍結集合"FS"が、コレクション"C"の上位集合(スーパーセット)であるかどうかを判断する。上位集合であれば真("True")を返す

```
>>> FS
frozenset({0, 1, 2, 3, 4})
>>> FS.isdisjoint([1,2])
False
>>> FS.isdisjoint([100,200])
True
>>> FS.issubset([0,1,2,3,4,5])
True
>>> FS.issubset([0,1,2,3])
False
>>> FS.issuperset([0,1,2,3])
True
>>> FS.issuperset([0,1,2,3,4,5])
```

```
False
```

6-2-2-2-5-2 集合(set)

■クラス名：'set'

集合型の可変オブジェクト版です。この項で説明するメソッドは、凍結集合(frozenset)と共通でないもののみです。凍結集合のメソッドは全て集合クラスにはあるので、その他のメソッドは凍結集合を参照してください

●要素(元)を操作する

■要素を加える

書式：S.add(I) ⇒ None
動作：集合"S"に要素"I"を加える。

■特定の要素を破棄（２種類）

書式１：S.remove(I) ⇒ None
書式２：S.discard(I) ⇒ None
動作：集合"S"から"I"を取り除く。書式１では、集合に無い要素を指定すると例外が発生するが、書式２では発生しない

■要素を取り出す

書式：S.pop() ⇒ I
動作：集合"S"の要素を端から順に取り出して返す。ランダムではないが、順序はハッシュされているので、順番に取り出したい場合は、リストにしてソートを掛けるような操作が必要となる

■要素を全て破棄する

書式：S.clear() ⇒ None
動作：集合"S"の要素を全て破棄し、空集合にする

```
>>> S = {x for x in range(10)}
>>> S
{0, 1, 2, 3, 4, 5, 6, 7, 8, 9}
>>> S.add(100)
>>> S
{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 100}
>>> S.remove(100)
>>> S
{0, 1, 2, 3, 4, 5, 6, 7, 8, 9}
>>> S.discard(9)
>>> S
{0, 1, 2, 3, 4, 5, 6, 7, 8}
>>> S.discard(100)  # 無い要素を指定してもエラーは出ない
>>> S.remove(100)   # 無い要素を指定するとエラーが出る
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
KeyError: 100
>>> S.pop()
0
>>> S
{1, 2, 3, 4, 5, 6, 7, 8}
>>> S.clear()
>>> S
set()
```

●更新する

■集合を更新する

書式：S.update(C) ⇒ None
動作：集合"S"に、コレクション"C"の要素を全て加えて更新する

■差集合に更新する

書式：S.defference_update(C) ⇒ None
動作：集合"S"とコレクション"C"の差集合に更新する

■積集合に更新する

書式：S.intersection_update(C) ⇒ None
動作：集合"S"とコレクション"C"の積集合に更新する

■対称差に更新する

書式：S.symmetric_difference_update(C) ⇒ None
動作：集合"S"とコレクション"C"の対称差に更新する

```
>>> S
{0, 1, 2, 3, 4}
>>> S.update((4,5,6,7,8,9))
>>> S
{0, 1, 2, 3, 4, 5, 6, 7, 8, 9}
>>> S.difference_update((0,1,2))
>>> S
{3, 4, 5, 6, 7, 8, 9}
>>> S.intersection_update((2,4,6,8,10))
>>> S
{8, 4, 6}
>>> S.symmetric_difference_update((2,4,10))
>>> S
{8, 2, 10, 6}
```

6-2-3 組み込み関数(built-in function or method)

6-2-3-1 数学的関数(mathematical function)

数学的な処理を行うための関数群です。
これ以外の数学的処理を行う関数は、'math'モジュールなどに収納されています。

■絶対値

書式：abs(C)
動作：数値"C"の絶対値を返す（"C"は実数と虚数を含む複素数が可能）

複素数の絶対値は、複素数平面での原点からの距離です。
直角三角形の斜辺の長さや、合成ベクトルの大きさを計算する際などに、手軽に便利に使えます。

例１）隣辺が長さ３と４長さの直角三角形の斜辺

```
>>> abs(3+4j)
5.0
```

例２）ベクトル(3,6)と(2,6)の合成ベクトルの大きさ

```
>>> abs(3+6j+2+6j)
13.0
```

■最大値(maximum number)、最小値(minimum number)

書式：max(ver1,var2..) ⇒ 最大値
書式：min(ver1,var2..) ⇒ 最小値
動作：与えられた引数の中で最大(あるいは最小)のものを返す。引数"ver1.."は、数値、順序列など。
文字：文字コードで比較
順序列：先頭の要素から要素順に比較する

※集合同士も比較できますが、順序が不規則（規則はあるのだけどわかりづらい）のであまり意味がありません

■除算／剰余(divide, module)

書式：divmod(A, B) ⇒ ([A ÷ B の商],[A ÷ B の剰余]
動作：除算(割り算)の、商と剰余(余り)をタプルで返す。数値"A","B"は実数

■累乗(冪乗)

書式：pow(A, B) ⇒ A^B
動作：数値"A"の"B"乗を返す。"A","B"は数値(複素数可能)。

■四捨五入(rounding)

書式：round(R, I) ⇒ [四捨五入された数値]
動作：数値"R"を、小数点第"I"桁で丸める。数値"R"は実数、数値"I"は整数

■合計(sum total)

書式：sum("C") ⇒ 合計値
動作：コレクション"C"の要素の合計値を返す。ただし、コレクションの要素は数値(複素数可能)

※'sum'関数は、１つの引数しか取らず、その内容はコレクション限定です。複数の引数を合計するわけではないので注意してください

```
>>> max(5,6,7)
7
>>> min(5,6,7)
5
>>> max([5,6,7])
7
>>> max([5,6,7],[3,4,8])
[5, 6, 7]
>>> divmod(10,3)
(3, 1)
>>> pow(2,10)
1024
>>> round(7/9,3)
0.778
>>> sum([1,2,3])
6
```

6-2-3-2 論理関数(logical function)

コレクションの各要素（辞書の場合はキー）に対して、そのオブジェクトが真であるか偽であるかを判定し、全体の結果を判定します。

■全真判定

書式：all(C) ⇒ bool
動作：コレクション"C"の要素が全て真ならば'True'を、一つでも偽ならば'False'を返す

■含真判定

書式：any(C) ⇒ bool

動作：コレクション"C"の要素が全て偽ならば'False'を、一つでも真ならば'True'を返す

```
>>> all([0,1,2,3])
False
>>> any([0,1,2,3])
True
>>> all([5,1,2,3])
True
>>> any([0,(),None,False])
False
>>>
```

6-2-3-3 変換関数(translating function)

変換関数は、文字⇔文字コードの変換や、記数法の変換を扱います。

なお、別クラスのオブジェクトに変換する「型キャスト」に相当するものは、Python では通常、クラス（コンストラクタ）の引数とします（文字列にするには"str(obj)"など）

【文字とコードの変換】

■文字コードを文字に変換(code to character)

書式：chr(I) ⇒ str

動作：文字コード"I"を、文字に変換する。文字コードは整数

■文字を文字コードに変換する(order of the character)

書式：ord(C) ⇒ int

動作：文字を文字コードに変換する。文字とは、長さ１の文字列のこと

※Python では、「文字」というクラスは特にありません。「文字」とは、「長さ１」の文字列のことです

```
>>> ord('𠮷')
12433
>>> chr(12433)
'𠮷'
```

【整数の記数法を変換する】

■二進法表示(to binary)

書式：bin(I) ⇒ str

動作：整数"I"を二進法表記の文字として返す

■八進法表示(to octal nation)

書式：oct(I) ⇒ str

動作：整数"I"を八進法表記の文字として返す

■十六進法表示(to hexadecimal nation)

書式：hex(I) ⇒ str

動作：整数"I"を十六進法表記の文字として返す

```
>>> bin(100)
'0b1100100'
```

```
>>> oct(100)
'0o144'
>>> hex(100)
'0x64'
```

※八進法は、現在ではあまり使われなくなりましたが、少し古いUNIXの本などにはASCIIコードが八進法で書かれていたりします。

6-2-3-4 オブジェクトに関する関数(function of Objects)

オブジェクトの性質を調べたり、属性について調べたりする関数群です。

【オブジェクトの性質】

■関数呼び出し可能かどうか(Is it callable?)

書式：callable(obj) ⇒ bool
動作：オブジェクト"obj"が関数として呼び出し可能（"obj(..)"の形で呼び出すことが出来る）ならば'True'を返す

※クラスオブジェクトはインスタンスを作るために、全て（コンストラクタとして）関数呼び出しが可能です

■特定のクラスのインスタンスかどうか(Is it a instance of the class?)

書式：isinstance(ins, cls) ⇒ bool
動作：オブジェクト"ins"が、クラス"cls"のインスタンスであれば'True'を返す

■特定のクラスのサブクラスかどうか(Is it a sub-class of the class?)

書式：issubclass("sub","sup") ⇒ bool
動作：オブジェクト"sub"が、クラス"sup"のサブクラスであれば'True'を返す

```
>>> class X:
...   def __call__(self):return self
...
>>> class C: pass
...
>>> class CC(C): pass
...
>>> x = X()
>>> c = C()
>>> cc = CC()
>>> callable(x)
True
>>> callable(c)
False
>>> callable(C)
True
>>> isinstance(c, C)
True
>>> isinstance(cc, CC)
True
>>> isinstance(cc, C)
True
>>> isinstance(c, CC)
False
>>> issubclass(CC, C)
True
>>> issubclass(C, CC)
False
```

```
>>> issubclass(X, C)
False
```

【属性に関する関数】

■属性を取得(get attribute)

書式：getattr(obj, attrname, defaultvalue) ⇒ value
動作：オブジェクト"obj"の"attrname"属性の値"value"を返す。"attrname"は文字列。"attrname"属性が無い場合、もし"defaultvalue"がセットされていればそれを返す。

■属性をセット(set attribute)

書式：setattr(obj, attrname, value) ⇒ None
動作：オブジェクト"obj"の"attrname"属性に値"value"をセットする。"attrname"は文字列。

■属性を消去(delete attribute)

書式：delattr(obj, attrname) ⇒ None
動作：オブジェクト"obj"の"attrname"属性を消去する。"attrname"は文字列。ただし削除できるのは、オブジェクト"obj"が直接保持している属性（インスタンスならインスタンス属性）のみ（クラス属性を消去する場合は、クラスに対して'delattr'を適用する）

■特定の属性を持っているかどうかの検査(dose this object has the attribute?)

書式：hasattr(obj, attrname) ⇒ bool
動作：オブジェクト"obj"が"attrname"属性を持っているならば'True'を返す。"attrname"は文字列。

```
>>> class C:
...   a = 100
...
>>> x = C()
>>> setattr(x,"b",200)
>>> getattr(x,"a")
100
>>> getattr(x,"b")
200
>>> getattr(x,"c")
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: 'C' object has no attribute 'c'
>>> getattr(x,"c",300)
300
>>> delattr(x,"a")
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: a
>>> delattr(x,"b")
>>> hasattr(x, "a")
True
>>> hasattr(x, "b")
False
>>> delattr(C,"a")
>>> hasattr(x, "a")
False
```

6-2-3-5 入出力関数(inuput/output function)

入出力に関する関数群です。直接、標準入出力(STDIO standard input/output)に関係する関数や、打ち出し形式の整形に関する関数などがあります。

■標準入力(input values)

書式：input(prompt) ⇒ str
動作：標準入力の値を文字列として返す。"prompt"は、入力の前に表示される文字列。

> ※Python2までの'raw_input'関数に相当します。なお、Python2までの'input'関数（入力された式を評価した値を返す）は、廃止されました。もし、同じ効果を期待するのであれば、"eval(input(prompt))"を使用してください（ただし、式を直接入力できる環境は、安全性があまり高くないので、使用場面に注意してください）

■標準出力(print it)

書式：print(value, ..., sep=' ', end='¥n', file=sys.stdout, flush=False)
動作：標準出力に文字列"value"(複数可)を渡す。'sep'オプションは、複数の文字列の区切り。'end'オプションは、文末に入れる文字（デフォルトは改行）、'file'オプションは、出力先(デフォルトは標準出力だが、ファイルストリームを指定できる)。'flush'オプションは、ストリームを強制的にフラッシュするかどうか

> ※Python2までは'print'はキーワードでしたが、Python3では関数になりました。一見、引数に括弧が必要になった分、わずらわしくなったように見えますが、関数になったことで、オブジェクトとして扱えるようになり、自由度は格段に上がったとも言えます。

■ファイルストリームを開く(open file stream)

書式：open(file, mode='rt', buffering=-1, encoding=None, errors=None, newline=None,
closefd=True, opener=None) ⇒ file
動作：ファイル"file"を開く。'mode'オプションは、以下の通り

'r'	読み取りモード
'w'	書き込みモード
'x'	新規ファイル（書き込み）
'a'	開いたファイルの末尾に追加するために開く
'b'	バイナリモードで開く
't'	テキストモードで開く
'+'	更新するために開く
'U'	改行コード互換モードで開く

'w'と'x'の違いは、'x'は既存のファイルがあった時例外を上げる。
その他のオプションについては省略

■オブジェクト表現を返す(representation)

書式：repr(obj) ⇒ str
動作：オブジェクト"obj"の文字表現を返す

■文字列のASCII表現を返す(ASCII string)

書式：ascii(S) ⇒ str
動作：文字列"S"を、アスキー文字表現で返す

■出力整形(formatting)

書式：format(obj) ⇒ str
動作：オブジェクト"obj"の出力を整形する

> ※'format'関数のオプションの詳細は「文字列フォーマット構文　詳細編」の「書式設定」を参照してください。

```
>>> k = input("<<< input:")
<<< input:input it
>>> k
'input it'
>>> print(k,end=":\n")
input it:
>>> f = open("test.txt","w")
>>> print(k,file=f)
>>> f.read()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
io.UnsupportedOperation: not readable
>>> f.close()
>>> ff = open("test.txt","r")
>>> ff.read()
'input it\n'
>>> ff.close()
>>> repr("text")
"'text'"
>>> ascii("abc")
"'abc'"
>>> ascii("あいうえお")
"'\\u3042\\u3044\\u3046\\u3048\\u304a'"
>>> repr("あいうえお")
"'あいうえお'"
>>> format(10.55,"*^10.1e")
'*1.1e+01**'
```

（投稿日 2012 年 12 月 30 日）

散歩世界プロジェクト

jscripter

はじめに

「散歩世界プロジェクト」という名前を実は気に入っている。英語では、Walking World Project である。知っている、あるいは知られている世界を散歩するというアイデアではない。散歩する未知の世界を把握しようというアイデアである。世界は未知に満ちている。

散歩世界プロジェクトへ参画を!!!

散歩世界プロジェクトは 2005 年に最初に手掛けたのだが、デジタルカメラと地図を活用すれば世界を把握できると考えた。最初は単に風景を撮って記録してマッピングしようと思った。緯度・経度のグリッドに撮影した写真を割り付けていけば風景世界が出来上がる。最初は撮影した方向を反映させたいと思った。記録が緯度・経度や方向をデジタルカメラで自動的に記録できればおもしろい。これは今や GPS や地磁気センサー内蔵のデジカメで実現できるようになった。それでプロジェクトを再稼働させたいと記事を書くことにした。なぜなら、撮影を自動化できたとしても世界は広すぎる。一人だけでは物理的に世界全体の把握は困難だからだ。みんなに参加してほしい。原始的なものだけど本当に作りたいのは次のような具合だ。一緒に世界グリッドを作ろう。

Walking World Project <http://text.world.coocan.jp/smap.html>

もう一つの要素は、Google Maps のページのサイト移転で Google Maps のバージョンを 2 から 3 へアップさせること。スクリプティングの観点からは本記事ではそこを取り上げる。移行前後のバージョンを次においている。Google Maps を利用するとすれば、散歩世界プロジェクトはこんな感じになるかなという例である。これをもうちょっと発展させたい。基本的にはコンテンツが重要なわけで、それをどのように表現するかは自由自在である。

Google Maps v2 から v3 への移行

実際、移行には試行錯誤でかなりの時間を要したが、API のプログラミングは理屈ではない。できてしまえば、それほど説明するほどのこともないので、次の Google のチュートリアルなどを参考にしてご確認いただきたい。JavaScript コードの部分を比較すれば容易にわかるだろう。

・Google Maps JavaScript API V3 チュートリアル - Google Maps API - Google Developers <https://developers.google.com/maps/documentation/javascript/tutorial?hl=ja>

・Google Maps JavaScript API V3 のイベント - Google Maps API - Google Developers <https://developers.google.com/maps/documentation/javascript/events?hl=ja>

・Google Maps Javascript API V3 Reference - Google Maps JavaScript API v3 - Google Developers <https://developers.google.com/maps/documentation/javascript/reference?hl=ja>

[バージョン 2]
Walking World Project <http://text.world.coocan.jp/googlemaps4_2.html>

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
```

```
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xmlns:v="urn:schemas-microsoft-com:vml">
<head>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8"/>
    <meta name="description" content="日記と連動するGoogle Maps APIを利用した「風景と記
憶」データベースの試みです。Walking World Projectと自称するパーソナル地理情報システムです。
">
    <meta name="keywords" content="Google Maps API,日記,風景,記憶,写真,地理情報システム">
    <title>Walking World Project</title>
    <style type="text/css">
        v:* {
                behavior:url(#default#VML);
        }
        body {background:#FFFFF0}
        p {text-indent: 2%; font-size:11pt}
    </style>
<!-- Google Maps API version 2 -->
    <script src="http://maps.google.com/maps?
file=api&v=2&key=ABQIAAAA_fIfpaL4LMc1rYGpOvH4fBSQ_z5dr1O_OVP7C-
xOUHXblOsQnRR2iL3Nz2o8kkVnQlVPuxWISm3_fQ" type="text/javascript"></script><!-- ←v=1 -->
    <script type="text/javascript">
    //<![CDATA[

    function createMarker(point, img, date, desc, link, title) {
        var marker = new GMarker(point);
        // Show this marker's index in the info window when it is clicked.
        var html = '<div style="width: 15em; style: font-size: small"><img src="' + img +'"
align=left hspace=10><b>' + date + '</b><br/>' + desc + ' <a href="' + link + '">' + title
+ '</a><br clear=all></div>';
        GEvent.addListener(marker, 'click', function() {
                marker.openInfoWindowHtml(html);
        });

        return marker;
    }

    function onLoad(lng, lat, data, zoom) {
        zoom = 17 - zoom;//← newZoom = 17 - oldZoom
        var map = new GMap2(document.getElementById("map"));//←Gmap2
        map.addControl(new GSmallMapControl());
        map.addControl(new GMapTypeControl());
        map.addControl(new GScaleControl());
        GEvent.addListener(map, 'moveend', function() {
            var center = map.getCenter();//← getCenterLatLng()
            var latLngStr = '(' + center.x + ', ' + center.y + ')';
            document.getElementById("message").innerHTML = latLngStr;
        });
        map.setCenter(new GLatLng(lat, lng), zoom);//← centerAndZoom(new GPoint(lng, lat),
zoom)
```

```
        var request = GXmlHttp.create();
        var baseurl = 'http://homepage1.nifty.com/kazuf/';
        request.open('GET', data, true);
        request.onreadystatechange = function() {
            if (request.readyState == 4) {
                var xmlDoc = request.responseXML;
                var markers = xmlDoc.documentElement.getElementsByTagName("marker");
                for (var i = 0; i < markers.length; i++) {
                    var point = new
GLatLng(parseFloat(markers[i].getAttribute("lat")),parseFloat(markers[i].getAttribute("lng"
)));//← GPoint...
                    var marker = createMarker(point, baseurl +
markers[i].getAttribute("img"),
                    markers[i].getAttribute("date"),markers[i].getAttribute("desc"),
                    baseurl +
markers[i].getAttribute("link"),markers[i].getAttribute("title"));
                    map.addOverlay(marker);
                }
            }
        }
        request.send(null);
    }

    //]]>
    </script>
</head>
<body onload="onLoad(132.472833, 34.385555, 'data.xml', 5)">
<h3>Walking World Project</h3>
<p>Google Maps API Version 2版 ver.0.8 (2006-05-05) since 2006-01-11</p>
<hr>
<table border=0>
<tr>
<td>
<div id="map" style="width: 500px; height: 600px"></div>
<div id="message"></div>
</td>
<td valign=top>
    <table>
    <tr><td>
    <a name="north"><b>[ 北 ]</b></a> <a href="#south"><b>↓南へ</b></a>
<hr>
    </td></tr>
    <tr><td>
    <input type="button" name="b6" value="県立美術館付近" onClick="onLoad(132.465141,
34.399793, 'data.xml', 1);">
<p>広島県立美術館は縮景園に隣接していて、広電白島線に面する交差点の対角線上にアーバンビューグランドタワーがある。</p>
<hr>
    </td></tr>
```

```
    <tr><td>
    <input type="button" name="b3" value="駅前大橋付近" onClick="onLoad(132.474132,
34.394888, 'data.xml', 1);">
<p>駅前大橋付近、福屋エールエールＡ館がある。</p>
<hr>
    </td></tr>
    <tr><td>
    <input type="button" name="b1" value="比治山" onClick="onLoad(132.472833, 34.385555,
'data.xml', 1);">
<p>比治山には広島市現代美術館がある。</p>
<hr>
    </td></tr>
    <tr><td>
    <input type="button" name="b2" value="国道２号線出汐町交差点付近"
onClick="onLoad(132.473458, 34.378711, 'data.xml', 1);">
<p>比治山の南、県立皆実高校、県立工業高校、進徳女子高校がある国道２号線出汐町交差点。</p>
<hr>
    </td></tr>
    <tr><td>
    <input type="button" name="b4" value="黄金山" onClick="onLoad(132.490228, 34.364817,
'data.xml', 1);">
<p>車でも登れる黄金山。春は桜がきれいだ。</p>
<hr>
    </td></tr>
    <tr><td>
    <input type="button" name="b5" value="宇品波止場" onClick="onLoad(132.469196,
34.353339, 'data.xml', 1);">
<p>広島高速３号線の宇品インターチェンジの海側、宇品波止場。10万トンバースで花火大会がある。
</p>
<hr>
    <tr><td>
    <input type="button" name="b7" value="カリブ(マリーナ広島)" onClick="onLoad(132.492928,
34.338864, 'data.xml', 1);">
<p>レストラン「カリブ」</p>
<hr>
    </td></tr>
    <tr><td>
    <a name="south"><b>[ 南 ]</b></a> <a href="#north"><b>↑北へ</b></a>
<hr>
    </td></tr>
    </table>
</td>
</tr>
</table>
<hr>
<I>(C)jscripter</I>
<script src="http://www.google-analytics.com/urchin.js" type="text/javascript">
</script>
<script type="text/javascript">
```

```
_uacct = "UA-155996-3";
urchinTracker();
</script>
</body>
</html>
```

[バージョン3]
Walking World Project <http://text.world.coocan.jp/googlemaps_v3_20130120.html>

```
<html>
<head>
    <meta name="viewport" content="initial-scale=1.0, user-scalable=no" />
    <meta http-equiv="content-type" content="text/html; charset=UTF-8"/>
    <meta name="description" content="日記と連動するGoogle Maps APIを利用した「風景と記憶」データベースの試みです。Walking World Projectと自称するパーソナル地理情報システムです。">
    <meta name="keywords" content="Google Maps API,日記,風景,記憶,写真,地理情報システム">
    <title>Walking World Project</title>
    <style type="text/css">
        v:* {
            behavior:url(#default#VML);
        }
        body {background:#FFFFF0}
        p {text-indent: 2%; font-size:11pt}
    </style>
<script type="text/javascript" src="http://maps.googleapis.com/maps/api/js?v=3.exp&key=AIzaSyDMutMOg-Hs8_r_ltNzA-ZD_i4_4Tg77qw&sensor=false"></script>
  <!-- jQuery -->
  <script type="text/javascript" src="https://ajax.googleapis.com/ajax/libs/jquery/1.6.1/jquery.min.js"></script>
<script type="text/javascript" src="util.js"></script>
<script type="text/javascript">
    function initialize(lat, lng ,zv) {
        var initPos = new google.maps.LatLng(lat, lng);
        var myOptions = {
            zoom : zv,
            center : initPos,
            mapTypeId : google.maps.MapTypeId.ROADMAP
        };
        var map = new google.maps.Map(document.getElementById("map_canvas"), myOptions);
        var infowindow = new google.maps.InfoWindow;
        var baseurl = 'http://homepage1.nifty.com/kazuf/';
        downloadUrl("data.xml", function(data) {
```

```
            var markers = data.documentElement.getElementsByTagName("marker");
            for (var i = 0; i < markers.length; i++) {
                var latlng = new
google.maps.LatLng(parseFloat(markers[i].getAttribute("lat")),
parseFloat(markers[i].getAttribute("lng")));
                var marker = new google.maps.Marker({
                    position: latlng,
                    map: map
                });
                var contentString = "<dl><dt>" +
markers[i].getAttribute("title") + "</dt><dd><a href='" +
markers[i].getAttribute("link") + "'><img src='" + baseurl +
markers[i].getAttribute("img") + "' align='left'/></a>" +
markers[i].getAttribute("desc") + " (" + markers[i].getAttribute("date") +
")</dd></dl>";
                bindInfoWindow(marker, map, infowindow, contentString);
            }
        });
    }

    function bindInfoWindow(marker, map, infoWindow, html) {
        google.maps.event.addListener(marker, 'click', function() {
            infoWindow.setContent(html);
            infoWindow.open(map, marker);
        });
    }
</script>
</head>
<body onload="initialize(34.385555, 132.472833, 12)">
<h3>Walking World Project</h3> <h4>since 2006-01-11
<p>Google Maps API Version 3版 ver.0.3 (2013-01-20) </p>
<hr>
<table border=0>
<tr>
<td>
<div id="map_canvas" style="width:500px; height:600px"></div>
</td>
<td valign=top>
    <table>
    <tr><td>
    <a name="north"><b>[ 北 ]</b></a> <a href="#south"><b>↓南へ</b></a>
<hr>
    </td></tr>
    <tr><td>
    <input type="button" name="b6" value="県立美術館付近"
onClick="initialize(34.399793, 132.465141, 16)">
<p>広島県立美術館は縮景園に隣接していて、広電白島線に面する交差点の対角線上にアーバンビューグランドタワーがある。</p>
<hr>
    </td></tr>
```

```
    <tr><td>
    <input type="button" name="b3" value="駅前大橋付近"
onClick="initialize(34.394888, 132.474132, 16)">
<p>駅前大橋付近、福屋エールエール A 館がある。</p>
<hr>
    </td></tr>
    <tr><td>
    <input type="button" name="b1" value="比治山" onClick="initialize(34.385555,
132.472833, 16)">
<p>比治山には広島市現代美術館がある。</p>
<hr>
    </td></tr>
    <tr><td>
    <input type="button" name="b2" value="国道 2 号線出汐町交差点付近"
onClick="initialize(34.378711, 132.473458, 16)">
<p>比治山の南、県立皆実高校、県立工業高校、進徳女子高校がある国道 2 号線出汐町交差点。</p>
<hr>
    </td></tr>
    <tr><td>
    <input type="button" name="b4" value="黄金山" onClick="initialize(34.364817,
132.490228, 16)">
<p>車でも登れる黄金山。春は桜がきれいだ。</p>
<hr>
    </td></tr>
    <tr><td>
    <input type="button" name="b5" value="宇品波止場"
onClick="initialize(34.353339, 132.469196, 16)">
<p>広島高速 3 号線の宇品インターチェンジの海側、宇品波止場。10 万トンバースで花火大会がある。</p>
<hr>
    <tr><td>
    <input type="button" name="b7" value="カリブ(マリーナ広島)"
onClick="initialize(34.338864, 132.492928, 16)">
<p>レストラン「カリブ」</p>
<hr>
    </td></tr>
    <tr><td>
    <a name="south"><b>[ 南 ]</b></a> <a href="#north"><b>↑北へ</b></a>
<hr>
    </td></tr>
    </table>
</td>
</tr>
</table>
<hr>
<I><copyright>(C)jscripter</copyright></I>
<script type="text/javascript">

 var _gaq = _gaq || [];
 _gaq.push(['_setAccount', 'UA-155996-5']);
 _gaq.push(['_trackPageview']);
```

```
 (function() {
  var ga = document.createElement('script'); ga.type = 'text/javascript';
ga.async = true;
  ga.src = ('https:' == document.location.protocol ? 'https://ssl' :
'http://www') + '.google-analytics.com/ga.js';
  var s = document.getElementsByTagName('script')[0];
s.parentNode.insertBefore(ga, s);
 })();

</script>
</body>
</html>
```

おわりに

さて、いかがだろうか。散歩世界プロジェクトに参画されたい方はご連絡をいただきたい。散歩世界プロジェクトのバージョン3はHTML5アプリケーションだ。

単なる地図と風景写真というだけではおもしろくないと思っている。地理的な情報と重ね合わせるとおもしろい情報・知識は様々に考えられる。情報処理システムとしても奥が深い。クラウド・アプリケーションであることも可能だし、オフラインのプライベートなアプリケーションにもなりえるだろう。

更新日記では、viewpointと名付けたMicroformatの導入も検討したことがある。僕が構想しているのはviewpointデータを持つ連邦型Webを結んで、一種のプラネット(地球)を作ることである。現在あるプラネットは、RSSなどのFeedから関連サイトのニュースを表示するのだが、地理的情報に関連する記事を世界グリッドに表示するのが散歩世界プロジェクトである。世界グリッドはGoogle Mapsであってもよい。様々な表現が出てきてよいのである。

(投稿日 2013年1月28日;改訂 2013年1月31日)

編集後記

jscripter

いつものことながら、編集後記を書く。実はまだメインの記事を書いていないが、後記をまずまとめて気楽に記事に取り組みたい。もう表紙・裏表紙は作ったしね。

HTML5 が今後のスクリプティングのキーワードになる。HTML5 は Web でデスクトップと同様にアプリケーションを動作させることを目的として考案されたのだ。スクリプタにとってはこれ以上のものはないが、それだけ中味が複雑になってきたことも間違いない。おそらく、僕はこの続きを更新日記の「JavaScript 追っ掛け再入門」で続けることになるだろう。

一方、TSNET はどこに向かうだろう。消失するのか。辛うじて細々と継続するのか。もっと何か他のものに変身するのか。それを決めるのは、これを読んでいるあなたである。

（投稿日 2013 年 1 月 28 日）

ＴＳＮＥＴスクリプト通信

ISSN 1884-2798　出版地: 広島市

2013 年 1 月　28 日　　5.2.000 版
2013 年 1 月　31 日刊行　　5.2.001 版

投稿規程

TSNETWiki ： 「投稿規程」のページを参照のこと

編集委員会(投稿順)

機械伯爵　kikwai[at]livedoor[dot]com
jscripter jscripter9[at]gmail[dot]com

著作権

使用許諾・配布条件

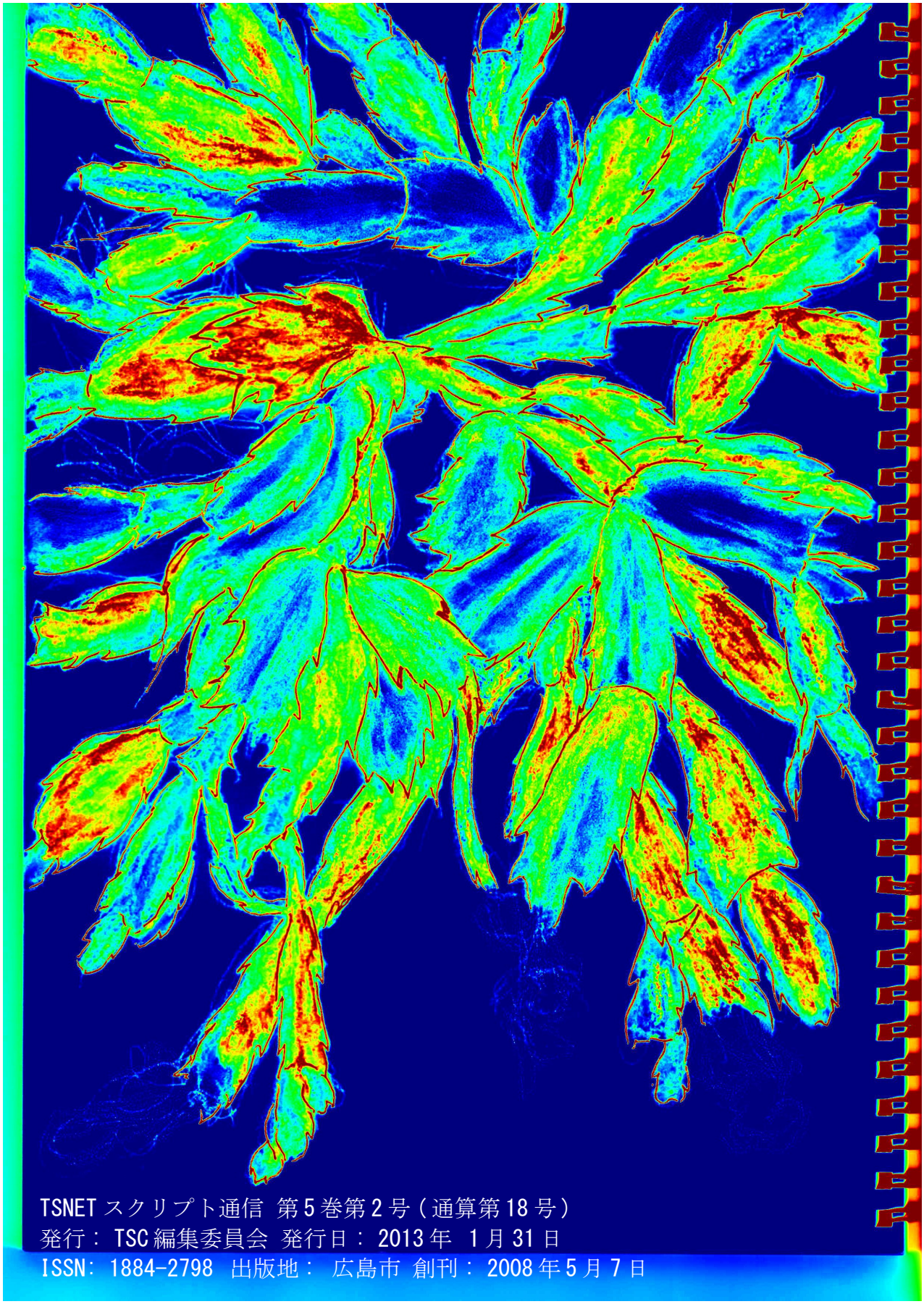
免責事項

編集ソフトウェア

LibreOffice 3.6.4.3 Writer (The Document Foundation)

発行所

一次配布所: TSNET スクリプト通信刊行リスト
http://text.world.coocan.jp/TSNET/?TSNET%E3%82%B9%E3%82%AF%E3%83%AA%E3%83%97%E3%83%88%E9%80%9A%E4%BF%A1%E5%88%8A%E8%A1%8C%E3%83%AA%E3%82%B9%E3%83%88



TSNET スクリプト通信 第5巻第2号（通算第18号）
発行： TSC 編集委員会 発行日： 2013年 1月31日
ISSN: 1884-2798 出版地： 広島市 創刊： 2008年5月7日