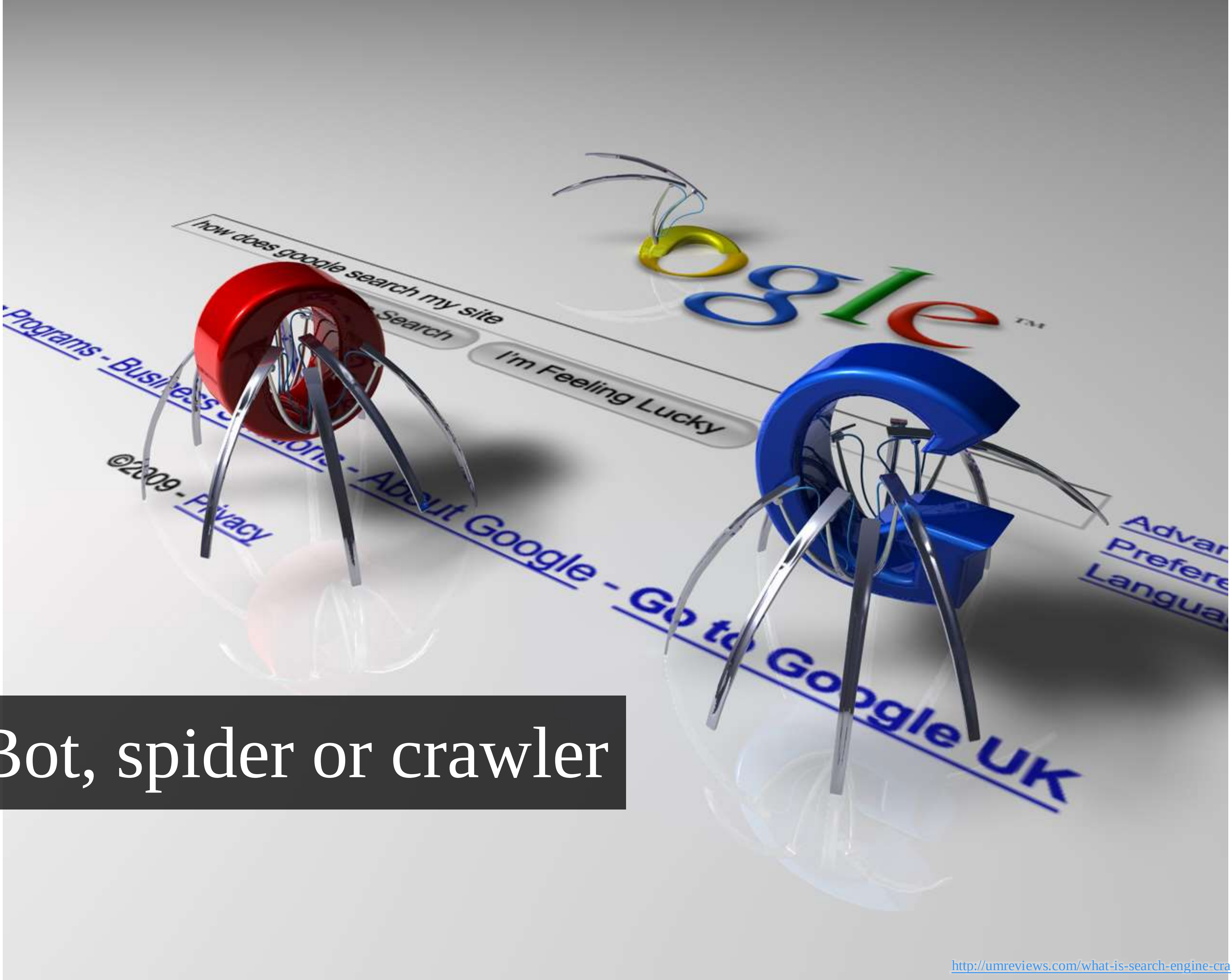
Molecular Biomedical Informatics
分 子 生 醫 資 訊 實 驗 室

W e b P r o g r a m m i n g
網 際 網 路 程 式 設 計

Bot
機器人


how does google search my site
Search
I'm Feeling Lucky
ogle
Programs - Business Solutions - About Google - Go to Google UK
©2009 - Privacy
Advar
Prefere
Langua

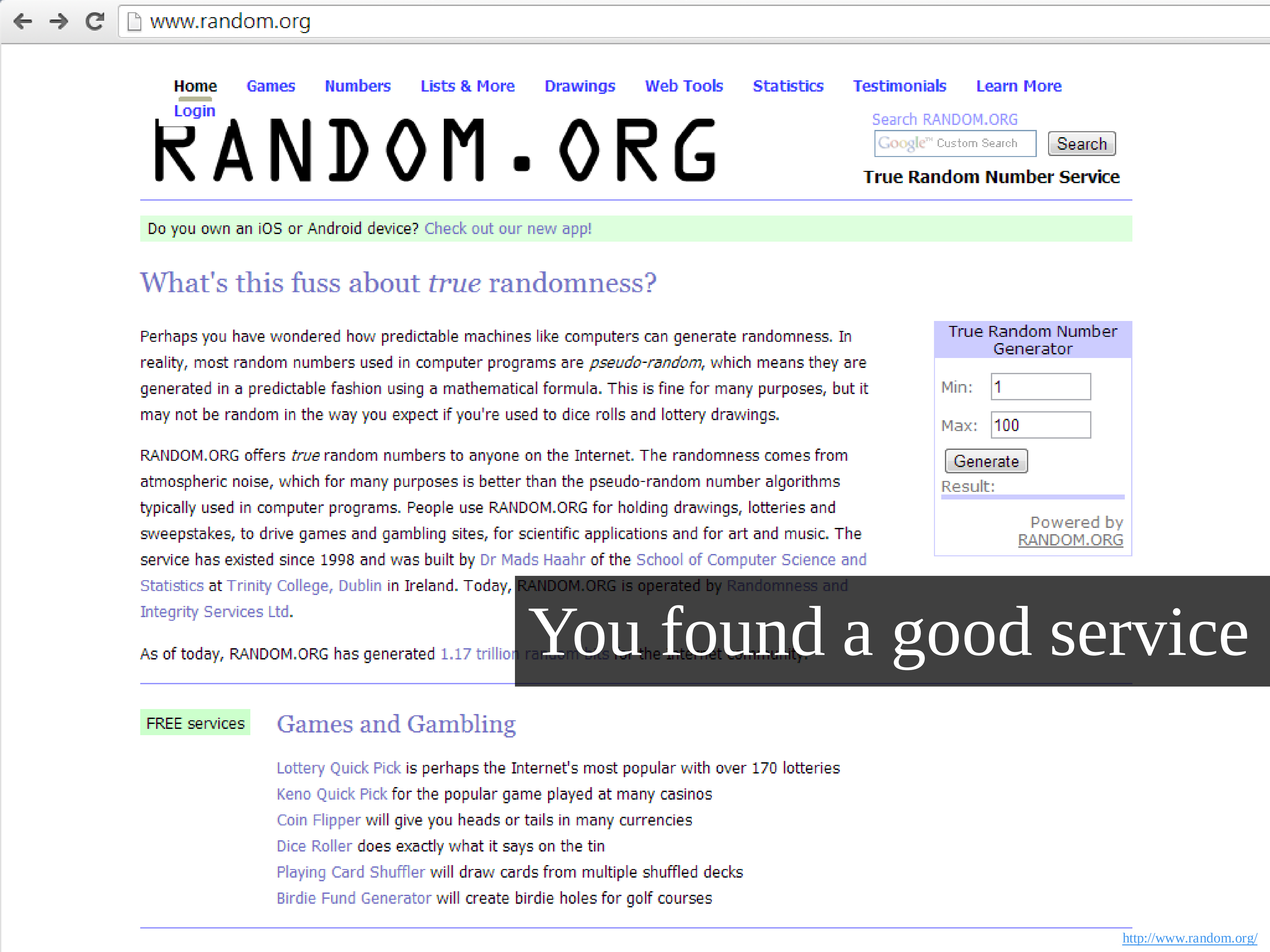
Bot, spider or crawler

http://umreviews.com/what-is-search-engine-crawler

Bot is useful

- Extract data from other sources
  - e.g. random numbers from random.org
  - e.g. word correction by google
- Commit data to other destinations
  - e.g. vote cheating or Search Engine Optimization (搜尋引擎最佳化, SEO)
- Automation
  - e.g. backup (daily archive user data and ftp it out)

- It could be performed by-request or periodically



Do you own an iOS or Android device? Check out our new app!

What's this fuss about *true* randomness?

Perhaps you have wondered how predictable machines like computers can generate randomness. In reality, most random numbers used in computer programs are *pseudo-random*, which means they are generated in a predictable fashion using a mathematical formula. This is fine for many purposes, but it may not be random in the way you expect if you're used to dice rolls and lottery drawings.

RANDOM.ORG offers *true* random numbers to anyone on the Internet. The randomness comes from atmospheric noise, which for many purposes is better than the pseudo-random number algorithms typically used in computer programs. People use RANDOM.ORG for holding drawings, lotteries and sweepstakes, to drive games and gambling sites, for scientific applications and for art and music. The service has existed since 1998 and was built by Dr Mads Haahr of the School of Computer Science and Statistics at Trinity College, Dublin in Ireland. Today, RANDOM.ORG is operated by Randomness and Integrity Services Ltd.

As of today, RANDOM.ORG has generated 1.17 trillion random bits for the Internet community.

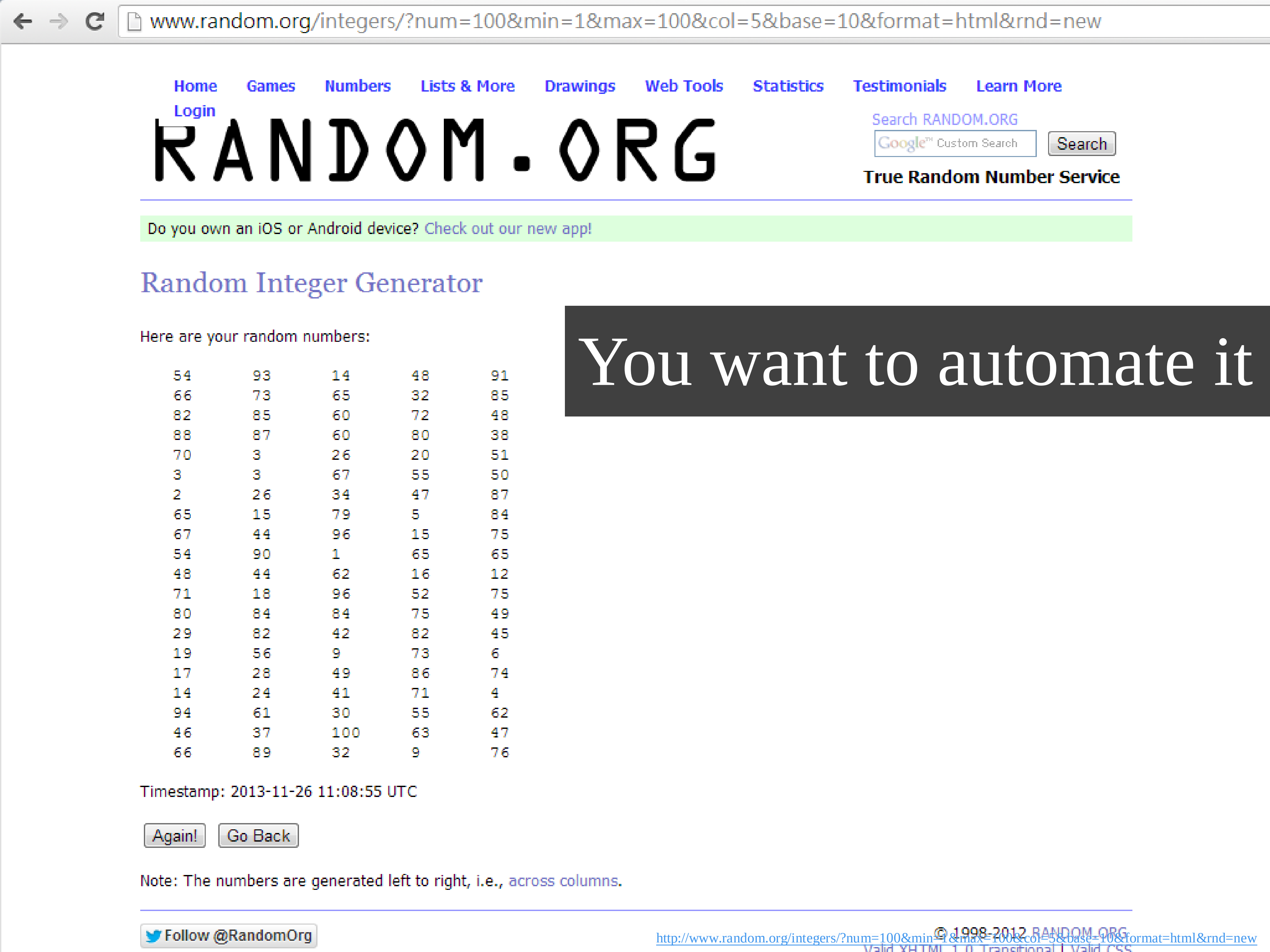
You found a good service

FREE services

Games and Gambling

Lottery Quick Pick is perhaps the Internet's most popular with over 170 lotteries
Keno Quick Pick for the popular game played at many casinos
Coin Flipper will give you heads or tails in many currencies
Dice Roller does exactly what it says on the tin
Playing Card Shuffler will draw cards from multiple shuffled decks
Birdie Fund Generator will create birdie holes for golf courses

http://www.random.org/



www.random.org/integers/?num=100&min=1&max=100&col=5&base=10&format=html&rnd=new

Home Games Numbers Lists & More Drawings Web Tools Statistics Testimonials Learn More Login

RANDOM.ORG

Search RANDOM.ORG

Search

True Random Number Service

Do you own an iOS or Android device? Check out our new app!

Random Integer Generator

Here are your random numbers:

54	93	14	48	91
66	73	65	32	85
82	85	60	72	48
88	87	60	80	38
70	3	26	20	51
3	3	67	55	50
2	26	34	47	87
65	15	79	5	84
67	44	96	15	75
54	90	1	65	65
48	44	62	16	12
71	18	96	52	75
80	84	84	75	49
29	82	42	82	45
19	56	9	73	6
17	28	49	86	74
14	24	41	71	4
94	61	30	55	62
46	37	100	63	47
66	89	32	9	76

Timestamp: 2013-11-26 11:08:55 UTC

Again! Go Back

Note: The numbers are generated left to right, i.e., across columns.

Follow @RandomOrg

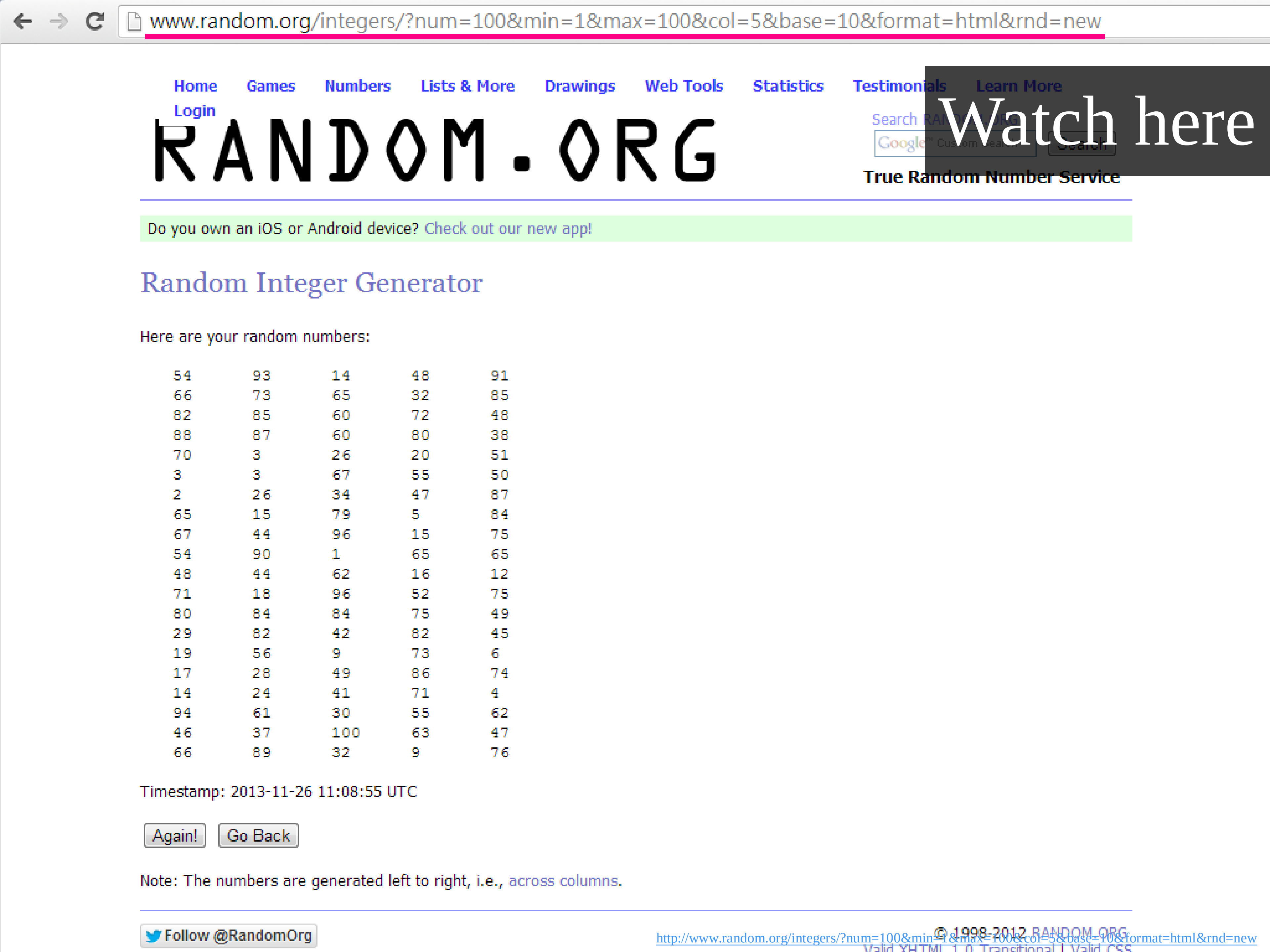
© 1998-2012 RANDOM.ORG

You want to automate it

http://www.random.org/integers/?num=100&min=1&max=100&col=5&base=10&format=html&rnd=new

What’s

the next step you do



www.random.org/integers/?num=100&min=1&max=100&col=5&base=10&format=html&rnd=new

Home Games Numbers Lists & More Drawings Web Tools Statistics Testimonials Learn More Login

RANDOM.ORG

Search RANDOM.ORG

True Random Number Service

Watch here

Do you own an iOS or Android device? Check out our new app!

Random Integer Generator

Here are your random numbers:

```
54      93      14      48      91
66      73      65      32      85
82      85      60      72      48
88      87      60      80      38
70      3       26      20      51
3       3       67      55      50
2       26      34      47      87
65      15      79      5       84
67      44      96      15      75
54      90      1       65      65
48      44      62      16      12
71      18      96      52      75
80      84      84      75      49
29      82      42      82      45
19      56      9       73      6
17      28      49      86      74
14      24      41      71      4
94      61      30      55      62
46      37      100     63      47
66      89      32      9       76
```

Timestamp: 2013-11-26 11:08:55 UTC

Again! Go Back

Note: The numbers are generated left to right, i.e., across columns.

Follow @RandomOrg

© 1998-2013 RANDOM.ORG

Valid XHTML 1.0 Transitional | Valid CSS

http://www.random.org/integers/?num=100&min=1&max=100&col=5&base=10&format=html&rnd=new

Just get the content

- In Perl

  -
    ```
    use LWP::Simple;
    my $content = get "http://www...";
    ```
- In PHP

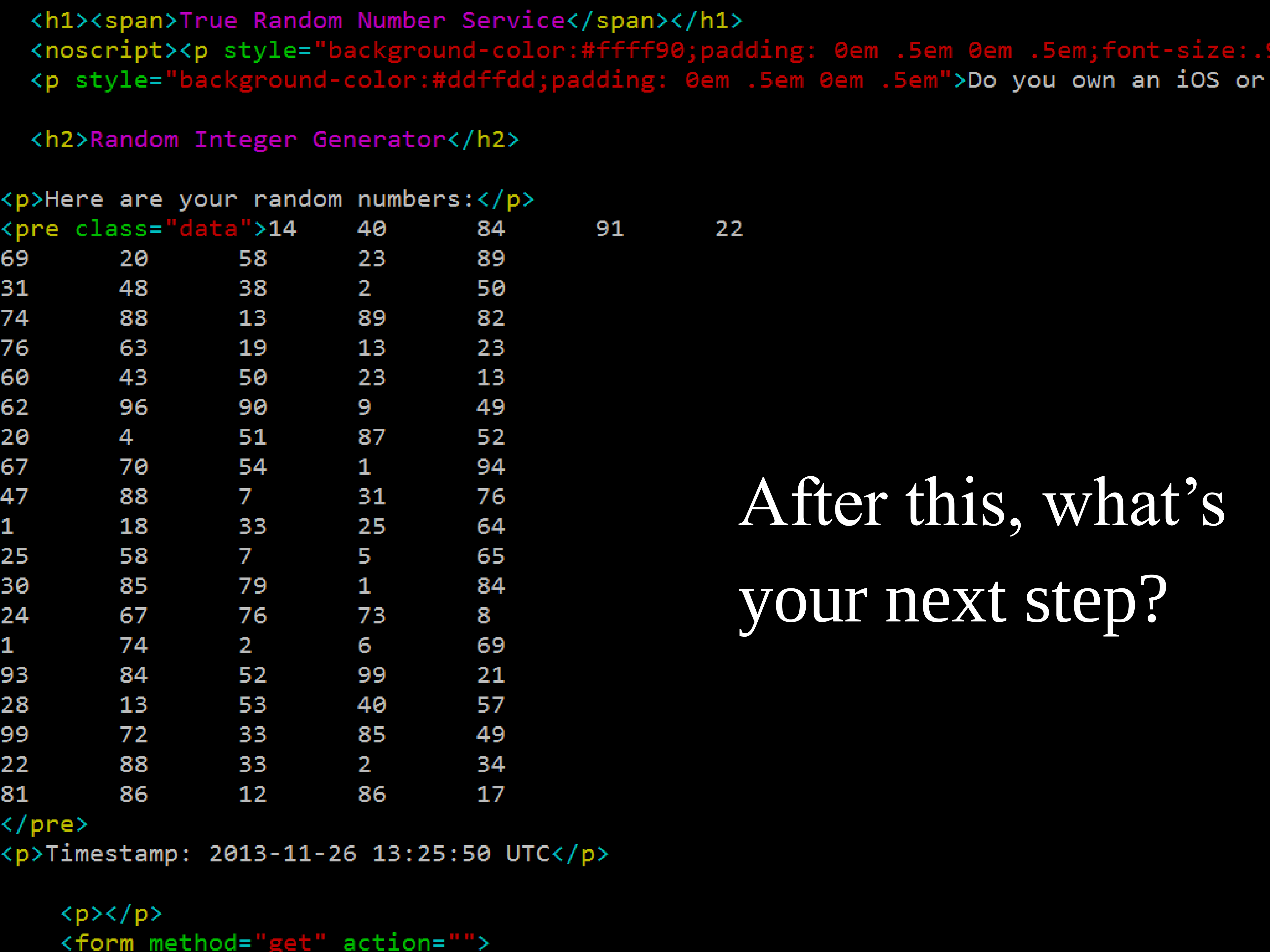
  -
    ```
    $content =
    file_get_contents("http://www...");
    ```
- In JavaScript (node.js)

  -
    ```
    http.request({
     host: 'www.random.org',
     path: '/integers/?num=100...'
    }, function(response){})
    ```

A more general way

- `curl -s 'http://www...'`
  - try the powerful Unix commands
  - `-s` indicates silent mode (up to you)
- In Perl, the backquote
  - `` `curl ...` ``
- In PHP
  - `exec('curl ...')`
- In JavaScript, everything is asynchronous

  -
    ```
    require('child_process')
    .exec('curl', function(){})
    ```



```
  <h1><span>True Random Number Service</span></h1>
  <noscript><p style="background-color:#ffff90;padding: 0em .5em 0em .5em;font-size:.
  <p style="background-color:#ddffdd;padding: 0em .5em 0em .5em">Do you own an iOS or

  <h2>Random Integer Generator</h2>

<p>Here are your random numbers:</p>
<pre class="data">14    40      84      91      22
69      20      58      23      89
31      48      38      2       50
74      88      13      89      82
76      63      19      13      23
60      43      50      23      13
62      96      90      9       49
20      4       51      87      52
67      70      54      1       94
47      88      7       31      76
1       18      33      25      64
25      58      7       5       65
30      85      79      1       84
24      67      76      73      8
1       74      2       6       69
93      84      52      99      21
28      13      53      40      57
99      72      33      85      49
22      88      33      2       34
81      86      12      86      17
</pre>
<p>Timestamp: 2013-11-26 13:25:50 UTC</p>

    <p></p>
    <form method="get" action="">
```

After this, what's your next step?

How many

code you need

One line

```
/"data">(.+?)</s and print $1;
```

Regular Expression
正規表示式

學會它，一輩子受用無窮

Regular expression

- A regular expression, or regex for short, is a pattern describing a set of strings that satisfy the pattern
  - e.g. `^A` means all strings that start with an ‘A’
  - in this slide, a string and its matched part is highlighted as `Adam`
  - in most regex engines (JavaScript, Perl, vi…), a regex is specified in between two slashes like `/^A/`

Literal character

- `a` matches `a`
- `a` matches `Jack is a boy`
  - match the first occurrence
  - not ~~`Jack is a boy`~~
- Characters with special meanings
  - `[ ] \ . ^ $ | ? * { } + ( )`
  - meta-characters
  - use backslash to escape them
  - matches `1+1=2` by `1\+1=2`

Character class/set

- Matches only one out of several characters
  - `[ae]` matches `a` or `e`
  - `gr[ae]y` matches `gray` or `grey`
  - `gr[ae]y` does not match ~~`graey`~~
- A hyphen inside a character class indicates range
  - `[0-9]` matches a single digit between 0 and 9
  - combine with single characters `[0-9a-fxA-FX]`
- A caret after the opening square bracket indicates not
  - `q[^x]` matches `question` but not ~~`Iraq`~~

Shorthand character classes

- `\d` matches a digit character
- `\w` matches a word character (alphanumeric characters and underscore)
- `\s` matches a whitespace character (space, tab and line break)
- `\D`, `\W` and `\S` are the complement sets

The dot

- `.` matches (almost) any char, except the line break
  - short for `[^\n]`
- Most regex engines have a single line mode that makes dot also match line break
  - `/regex/s`

Anchors

- Anchors do not match any characters but match a position
- `^` matches at the start of the string
- `$` matches at the end of the string
- `\b` matches at a word boundary
  - `\bis\b` matches `This island` <u>`is`</u> `nice`

Alternation

- The regular expression equivalent of “or”
  - `cat|dog` matches About cats and dogs
  - can be more than two `cat|dog|mouse|fish`

Repetition

- The question mark indicates optional (zero or one time)
  - `colou?r` matches `colour` or `color`
- The asterisk indicates zero or more times
  - `<[A-Za-z][A-Za-z0-9]*>` matches an HTML tag without any attributes
- The plus indicates one or more times
  - `<[A-Za-z0-9]+>` is easier to write but matches invalid tags such as `<2>`
- Curly brace indicates specific amount of repetition
  - `\b[1-9][0-9]{3}\b` matches 1000 to 9999
  - `\b[1-9][0-9]{2,4}\b` matches 100 to 99999

Greedy and lazy repetition

- The repetition is greedy by default
  - `<.+>` matches `This is <EM>good</EM>`
- A question mark make it lazy
  - `<.+?>` matches `This is <EM>good</EM>`

Grouping and back-references

- Round bracket indicates group
  - `Set(Value)?` matches `Set` or `SetValue`
- Round bracket creates back-references
  - how to access the back-references depends on the regex engine
  - `$0 = SetValue` and `$1 = Value` or `$0 = Set` and `$1` is null in Perl

Any Questions?

about regex

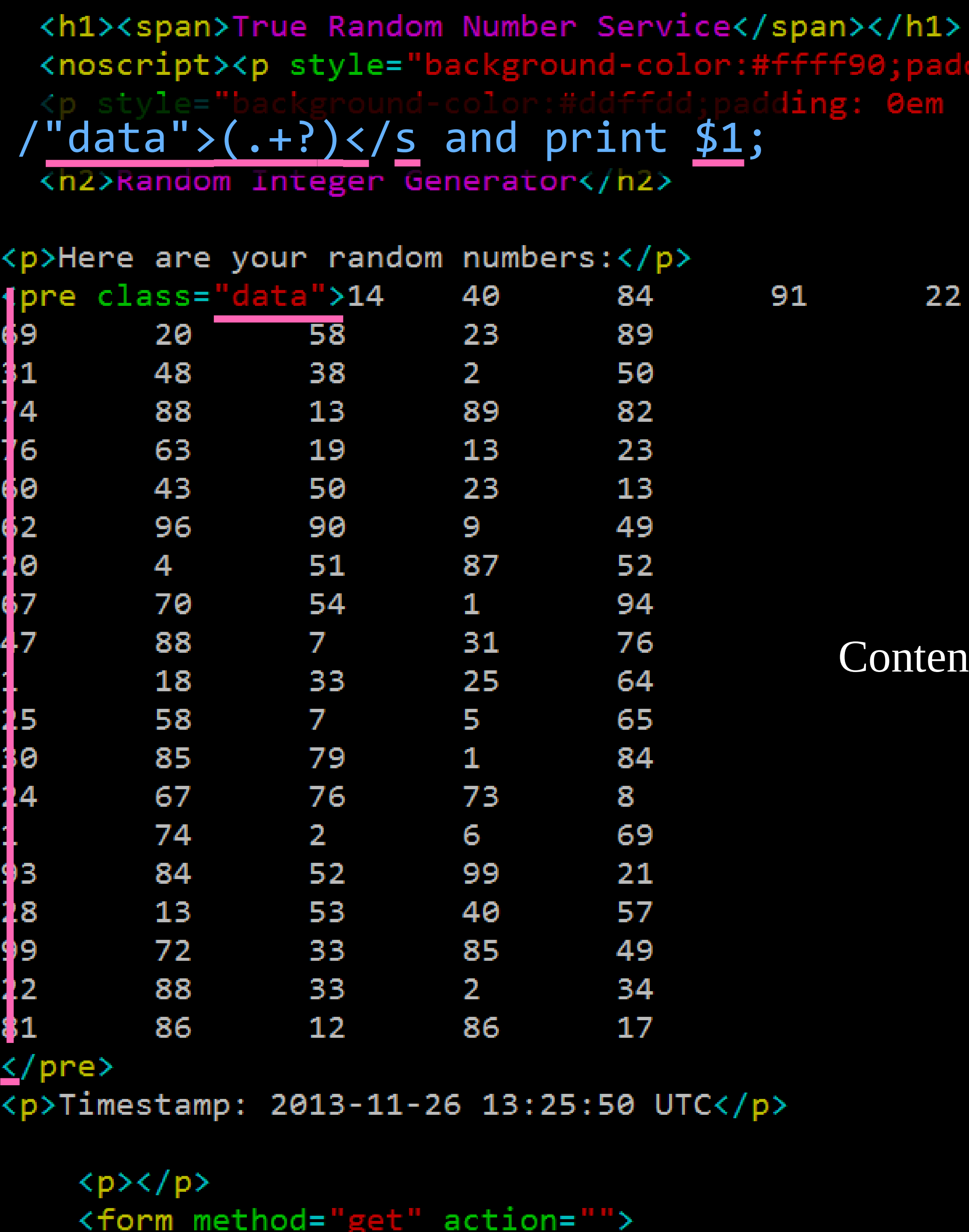
How many

meta-characters you remember

```
[ ] \ . ^ $ | ? * { } + ( )
```

One line

```
/"data">(.+?)</s and print $1;
```



```
  <h1><span>True Random Number Service</span></h1>
  <noscript><p style="background-color:#ffff90;padding: 0em .5em 0em .5em;font-size:.
  <p style="background-color:#ddffdd;padding: 0em .5em 0em .5em">Do you own an iOS or
 /"data">(.+?)</s and print $1;
  <h2>Random Integer Generator</h2>

<p>Here are your random numbers:</p>
<pre class="data">14     40      84      91      22
69      20      58      23      89
31      48      38      2       50
74      88      13      89      82
76      63      19      13      23
60      43      50      23      13
62      96      90      9       49
20      4       51      87      52
67      70      54      1       94
47      88      7       31      76
1       18      33      25      64
25      58      7       5       65
30      85      79      1       84
24      67      76      73      8
1       74      2       6       69
93      84      52      99      21
28      13      53      40      57
99      72      33      85      49
22      88      33      2       34
81      86      12      86      17
</pre>
<p>Timestamp: 2013-11-26 13:25:50 UTC</p>

    <p></p>
    <form method="get" action="">
```

Content Any ch Use brackets to create the

to prev back-reference and print it

Painful?

But, before parsing the content

Parse the request

- `http://www.random.org/integers/?num=100&min=1&max=100&col=5&base=10&format=html&rnd=new`

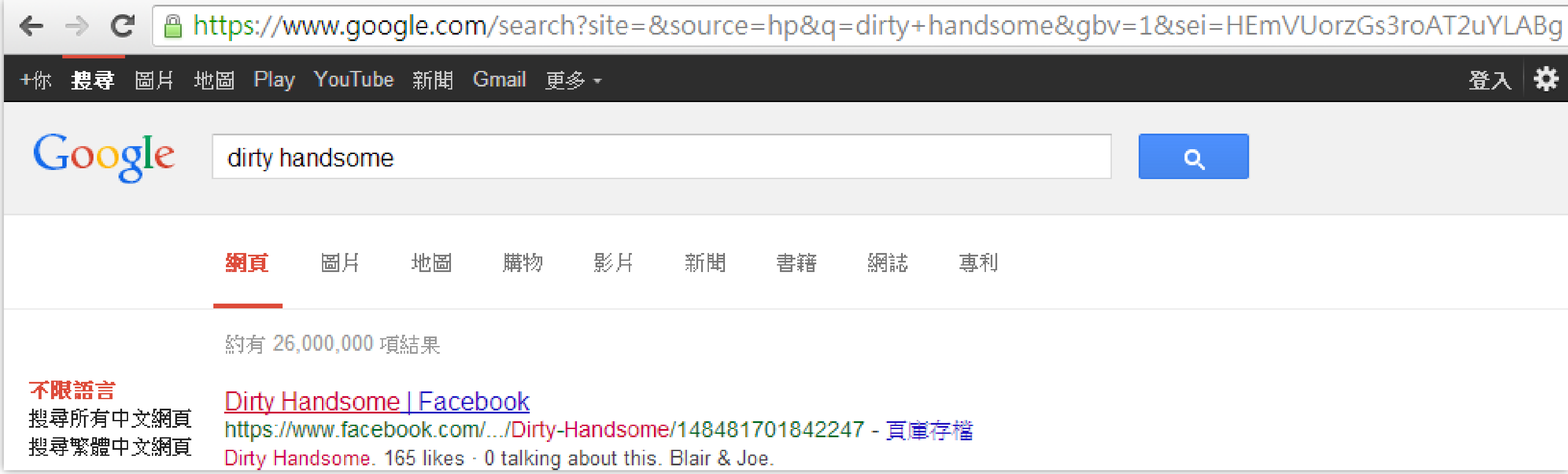
Parse the request

- `http://www.random.org/integers/?num=100&min=1&max=100&col=5&base=10&format=html&rnd=new`
- This is usually done manually

Occurrences in Google

The keyword should come from command line. Here we just keep the code simple.

```
my $kw = 'color';
my $ua = '-A "Mozilla/5.0"';
$kw =~ s/\s+/+/g;
my $url =
  "http://www.google.com/search?q=$kw";
$_ = `curl $ua -s '$url'`;
/"resultsStats".*? (\S+)/ and print $1;
```


https://www.google.com/search?site=&source=hp&q=dirty+handsome&gbv=1&sei=HEmVUorzGs3roAT2uYLABg

+你 搜尋 圖片 地圖 Play YouTube 新聞 Gmail 更多 ▾ 登入

dirty handsome

網頁 圖片 地圖 購物 影片 新聞 書籍 網誌 專利

約有 26,000,000 項結果

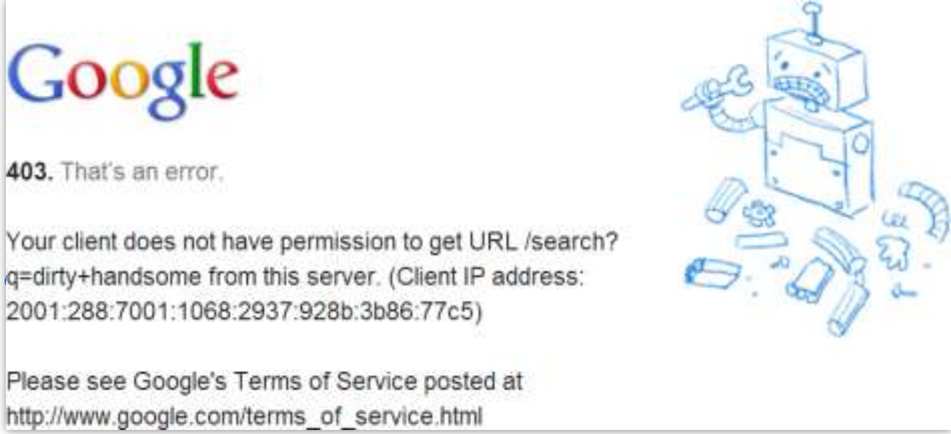
不限語言
搜尋所有中文網頁
搜尋繁體中文網頁

Dirty Handsome | Facebook
https://www.facebook.com/.../Dirty-Handsome/148481701842247 - 頁庫存檔
Dirty Handsome. 165 likes · 0 talking about this. Blair & Joe.

```
$kw =~ s/\s+/+/g;
my $url =
   "http://www.google.com/search?q=$kw";
$_ = `curl $ua -s '$url'`;
/"resultsStats".*? (\S+)/ and print $1;
```

Replace spaces to +, globally. Figure out this rule manually. Google's advanced web techniques make this harder. Watch the substitute operator with three slashes. The global mode indicates replace all occurrences rather than the first one.

Occurrences in Goo



Google

403. That's an error.

Your client does not have permission to get URL /search?q=dirty+handsome from this server. (Client IP address: 2001:288:7001:1068:2937:928b:3b86:77c5)

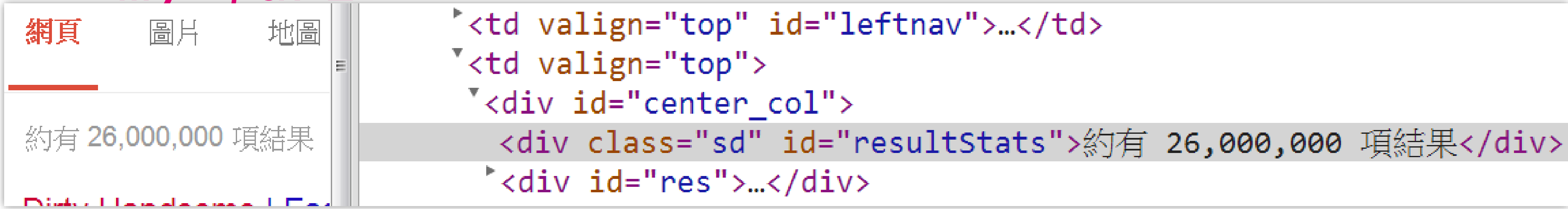
Please see Google's Terms of Service posted at http://www.google.com/terms_of_service.html

```
▪ my $kw = 'color';
  my $ua = '-A "Mozilla/5.0"';
  $kw =~ s/\s+/+/g;
  my $url =
    "http://www.google.com/search?q=$kw";
  $_ = `curl $ua -s '$url'`;
  /"resultsStats".*? (\S+)/ and print $1;
```

Get the results. The user agent string `$ua` is required to cheat Google.

Occurrences in Google

```
my $kw = 'color';
my $ua = '-A "Mozilla/5.0"';
$kw =~ s/\s+/+/g;
my $url =
```


網頁 圖片 地圖

約有 26,000,000 項結果

```
<td valign="top" id="leftnav">…</td>
<td valign="top">
 <div id="center_col">
  <div class="sd" id="resultStats">約有 26,000,000 項結果</div>
  <div id="res">…</div>
```

```
/"resultsStats".*? (\S+)/ and print $1;
```

Design the regex to parse the result.

Tip

- A demo, the backup of zoro
- crontab
  - if you want to perform some works periodically
  - 例行性工作排程的建立
- Net::Telnet
  - if you want to deal with BBS rather than web platforms
  - Re: 又出現一個PTT備份站了

Today’s assignment
今天的任務

Extract something from other sites

- Adopt any way mentioned today to enrich the content your site. If you have no ideas, try providing hot news from news sites or hot keywords from search sites.
- Use regular expression
- Reference
  - Regular-Expressions.info
- Your web site (http://merry.ee.ncku.edu.tw/~xxx/cur/, ex9) will be checked not before 23:59 6/18 (Mon). You may send a report (such as some important modifications) to me in case I did not notice your features. Or, even better, just explain your modifications in the homepage.