

実際に要素を取得してみる

```
var casper = require('casper').create({
  pageSettings: {
    loadImages: false,    //画像読み込みをスキップ
    //Firefox
    userAgent: "Mozilla/5.0 (Windows NT 6.3; Win64; x64; rv:45.0) Gecko/20100101
Firefox/45.0"
  },
  viewportSize : { width: 1440, height: 900 }
});

//ヘルパーモジュール
var u = require('utils');

casper.start('http://www.for-it.co.jp/', function() {
  this.echo("HTML: "+this.getHTML('#HEAD'));
  //this.echo("HTML: "+this.getHTML('#HEAD', true));
  this.echo("URL: "+this.getCurrentUrl());
  this.echo("タイトル: "+this.getTitle());
  this.echo("キーワード: "+this.getElementAttribute('meta[name="keywords"]', 'content'));
  this.echo("ロゴ src: "+this.getElementAttribute('img[alt="株式会社フォーイトロゴマーク"]',
'src'));
  u.dump(this.getElementsAttribute('a', 'href'));
  u.dump(this.getElementsAttribute('img', 'src'));
  this.echo(this.fetchText('.title_Aria p'));
  u.dump(this.getElementInfo('.view.view-first'));
  //複数取得する場合はgetElementsInfo 後に使用
});

casper.run();
```

getHTML

第二引数にtrueで外側の指定したhtmlも取得

getElementAttribute

第一引数に指定する要素、第二引数に属性値を取りたい属性

getElementInfo

最初にマッチした最初の要素について情報を取得

utils.dump

引数をjson表現をダンプ

フォーム操作

```
var casper = require('casper').create({
  pageSettings: {
    loadImages: false,    //画像読み込みをスキップ
    //Firefox
    userAgent: "Mozilla/5.0 (Windows NT 6.3; Win64; x64; rv:45.0) Gecko/20100101
Firefox/45.0"
  },
  viewportSize : { width: 1440, height: 900 }
});

//ヘルパーモジュール
var u = require('utils');

casper.start('https://twitter.com/login?lang=ja', function(){

  this.capture('./img/twitter_login.png');

  this.fillSelectors('form.js-signin', {
    'input[name="session[username_or_email]"': 'test',
    'input[name="session[password]"': 'test',
  }); //trueにすることでsubmit

  this.capture('./img/form_fill.png');
});

// casper.then(function(){
//   this.wait(3000)
//   .then(function(){
//     this.capture('./img/submit_result.png');
//   });
// });

//実行する
casper.run();
```

クッキー取得

```

var casper = require('casper').create({
  pageSettings: {
    loadImages: false,    //画像読み込みをスキップ
    //Firefox
    userAgent: "Mozilla/5.0 (Windows NT 6.3; Win64; x64; rv:45.0) Gecko/20100101
Firefox/45.0"
  },
  viewportSize : { width: 1440, height: 900 },
  onWaitTimeout: function(timeout, selector) {
    this.echo('timeout Error');
    this.capture('./img/login_error.png');
    this.exit();
  },
});

//ヘルパーモジュール
var u = require('utils');
//ファイルシステム
var fs = require('fs');

casper.start('https://twitter.com/login?lang=ja', function(){

  this.fillSelectors('form.js-signin', {
    'input[name="session[username_or_email]"']: 'test',
    'input[name="session[password]"']: 'test',
  }, true); //trueにすることでsubmit

});

casper.then(function(){

  this.waitForSelector('.DashboardProfileCard-content', function(){
    this.echo('login success');
    this.capture('./img/login_after.png');
    //クッキー保存
    var cookies = JSON.stringify(phantom.cookies);
    fs.write("./config/t_cookie.txt", cookies, 644);
  });

});

//実行する
casper.run();

```

```

var fs = require('fs');
CommonJSファイルシステムの提案をモデルにした
ファイルシステムモジュール
ファイルの書き込みや読み込みが可能

```

then()

直前のステップがある場合それを待つ。

また、リクエストしたときのurlの読み込みをまつ

waitForSelector

指定したセレクトアが出現するまで

処理の実行を待つ

phantom.cookies

ブラウザに与えられたクッキーを取得

casperはphantomjsを動かしているため

参照可能

onWaitTimeout: function(timeout, selector) {}

タイムアウト時に呼ばれるコールバック

exit()

phantomjsを終了させる

//configファイルの読み込み

```
var config = require('./config/config');
```

```
casper.start('https://twitter.com/login?lang=ja', function(){
```

```
    this.fillSelectors('form.js-signin', {
        'input[name="session[username_or_email]"]': config.user,
        'input[name="session[password]"]': config.passwd,
    }, true); //trueにすることでsubmit
```

```
});
```