

# Tutorial 1: Data Collection

本資料は KapowKatalyst8.3.1 をベースに作成されているため、所々画面の表示項目が最新版と一致していませんが、内容には問題ございません。

## 概要

練習用の Web サイトへアクセスし、Kapow の「Kapow Katalyst tip of the week」リストを収集した後にデータベースへ格納するという、単純なロボットを作成します。

まずは収集するデータを格納するための型（type）を作成しましょう。この型を Tip と名付け、tips の内容を記述するための属性（attribute）を持たせましょう。そうしたら続けてロボットを作成します。ちなみに本演習で作成するロボットは練習用の Web サイトからほんの幾つかの情報を読み込むだけの簡単なロボットです。

ロボットの作成後は、実際にそのロボットを動かして見ましょう。まずは「Design Studio」に含まれているデバッガーでロボットを実行します。続いて「Management Console」からもロボットを動かしてみましょう。次に「RoboClient」から「RoboServer」を呼出してのロボットの実行。最後にプログラムから Java の API を通じて「RoboServer」を呼出してロボットを実行してみましょう。

本チュートリアルではロボットを作成してゆく過程を、一つ一つの作業単位に細かく分割し、それぞれ STEP として記述しています。各 STEP は冒頭に短い説明文と、その後【操作してみましょう】と【確認しましょう】という内容で構成されています。

## 注)

本チュートリアルを進めていく過程で、何度か「インストールフォルダ」という言葉が登場しますが、これは Kapow Katalyst をインストールした先のフォルダを意味しています。

Windows ではインストール時にデフォルト設定を選択した場合、C : ¥Program Files¥Kapow Katalyst 8.3.1 がインストールフォルダになります。ただ実際には正確なフォルダはインストールしたユーザにより異なるため、本チュートリアルにおいては一律「インストールフォルダ」という言葉でそれを示します。

## Tip 型の定義

Tip 型を定義するために最初にとすることとして、まずは練習用の Web サイトへアクセスし、ロボットに何をさせたいのかを考えて見ましょう。

ブラウザで次の URL にアクセスして下さい。

<http://www.kapowsoftware.com/tutorial/quickstart> このサイトは本チュートリアル専用の練習用 Web サイトなので、非常にシンプルな作りになっています。トップページには Number と Tip のタイトルだけが表示されており、title にはられているリンクの先には、それぞれの tip に関する記述が表示されます。まずは数分間かけて、このサイトの構造をよく理解しましょう。

Tip 情報を収集してデータベースへ書き込むロボットを作成するために、個々の Tip を識別できるような型を作成しましょう。型名は“Tip”とします。

以下、“Tip”型に含まれる属性を追加してゆきましょう。

title :

何について書かれた tip なのかを記述する短い文字列を挿入します。

number:

tip の No を通番で割り振ります。No は 1 から始まり 1 ずつ加算してゆきます。

url:

tip の記述ページ URL を挿入します。

description:

1 つの短い段落で記述された実際の tip の記述を挿入します。

## Step 1 — Design Studio を開いてみましょう

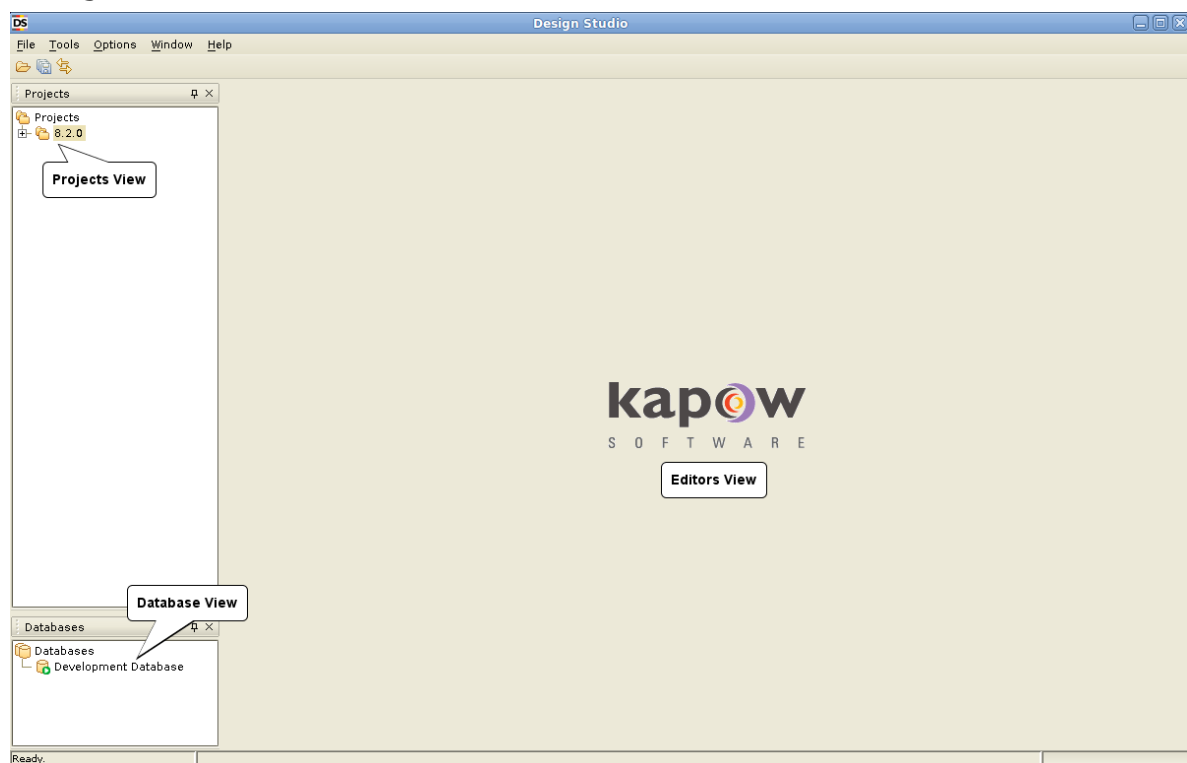
型の定義は完了したので、早速 Design Studio を使って作ってみましょう。

### 【操作してみましょう】

Windows のメニューから「Design Studio」を選択して起動しましょう。

### 【確認しましょう】

「Design Studio」が起動して、次のような画面が表示されます。



Design Studio Welcome Screen

このチュートリアルでは、Design Studio のさまざまなコンポーネントに関して、それを使用する度に短い説明を提供します。Design Studio に関するより完全なチュートリアルについては、[Design Studio User's Guide](#) をご確認ください。

## Step 2 — Type の新規作成

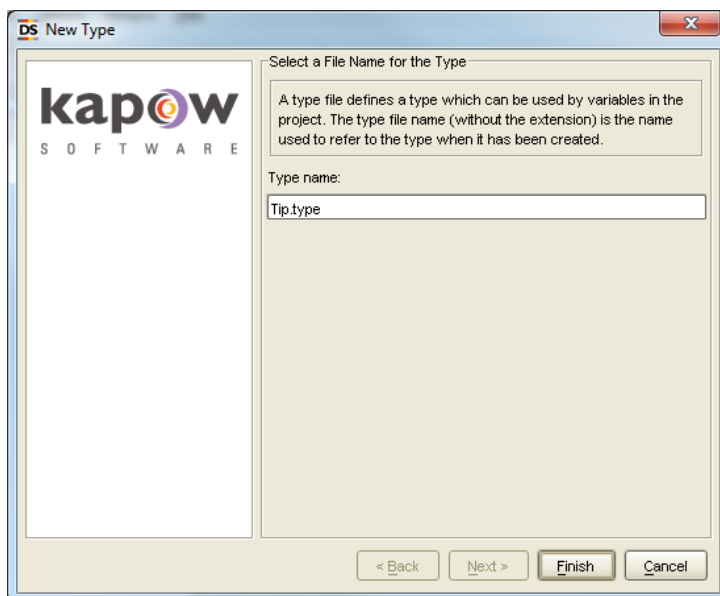
型（Type）ファイルを新規作成します。

### 【操作してみましょう】

プロジェクトビュー（Project View）内の "8.3.1" を右クリックし、"New..." | "Type" を選択します

### 【確認しましょう】

「New Type」ダイアログが表示されますので、作成する型のファイル名を入力しましょう。



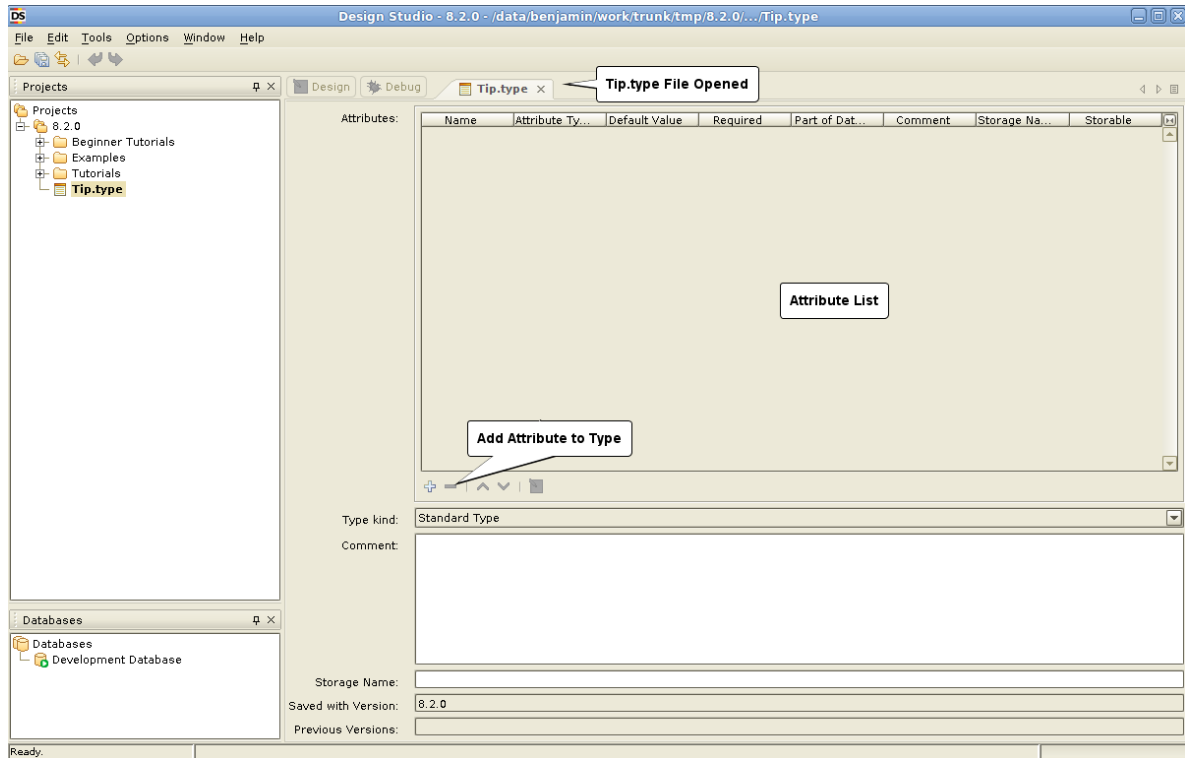
New Type Window

### 【操作してみましょう】

ファイル名として「Tip.type」を入力し、「Finish」ボタンをクリックしましょう。


## 【確認しましょう】

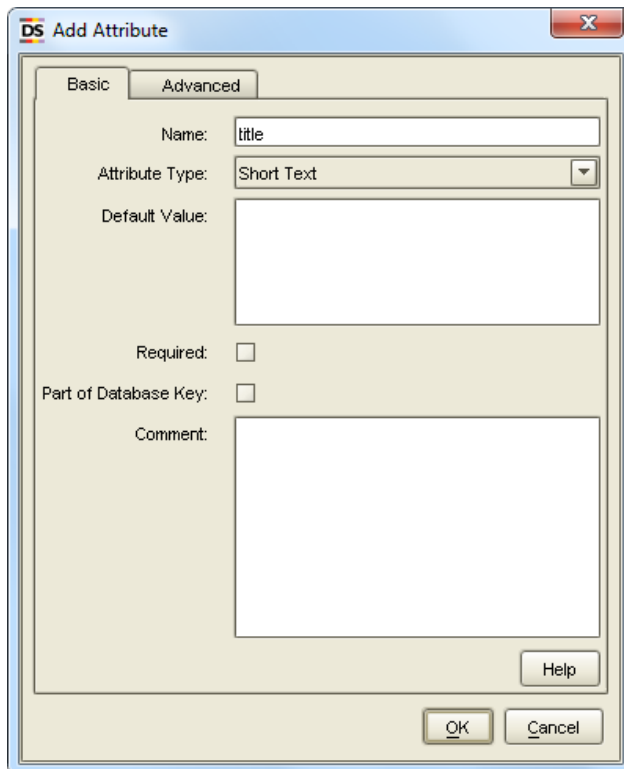
ファイル名を入力後、「Tip.type」ファイルが生成されると、下記のようなタイプエディターの画面が表示されます。



## Type Editor

## Step 3 — Tip 型の属性定義

最初は属性リストの中身が空なので、tips に関する情報を保持するための 4 つの属性を作成します。  
アトリビュートリスト (Attribute) 欄の下に見える  ボタンをクリックしましょう。



### Adding an Attribute

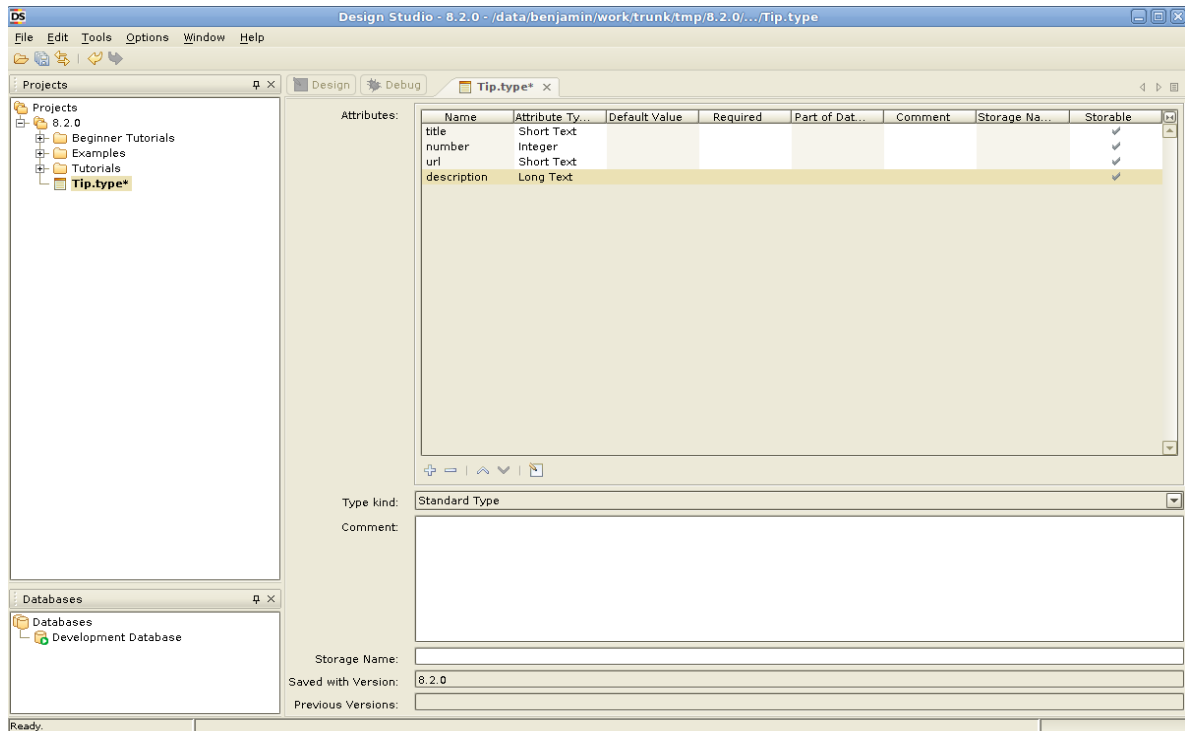
名称 (Name) 欄に「title」と記入し、型 (Type) から「Short Text」を選択してください。上記イメージの通り入力ができたら「OK」ボタンをクリックしてダイアログを閉じてください。  
リストに新しい属性が定義されたことが確認できます。

以下、上記と同様の手順で 3 つの属性を追加し、テーブルに記述された 4 つの型を宣言してください。

Attribute Name	Type
title	Short Text
number	Integer
url	Short Text
description	Long Text

### Attributes of the Tip Type

すべての属性追加後、画面の表示が以下のようにになっていることをご確認ください。



Type Editor after Addition of Attributes

## Step 4 — 型の保存

作成した型をディスクに保存しましょう。

File メニューから Save を選択してファイルを保存してください。

## ロボットの作成

Design Studio を使って、ロボットを作成しましょう。

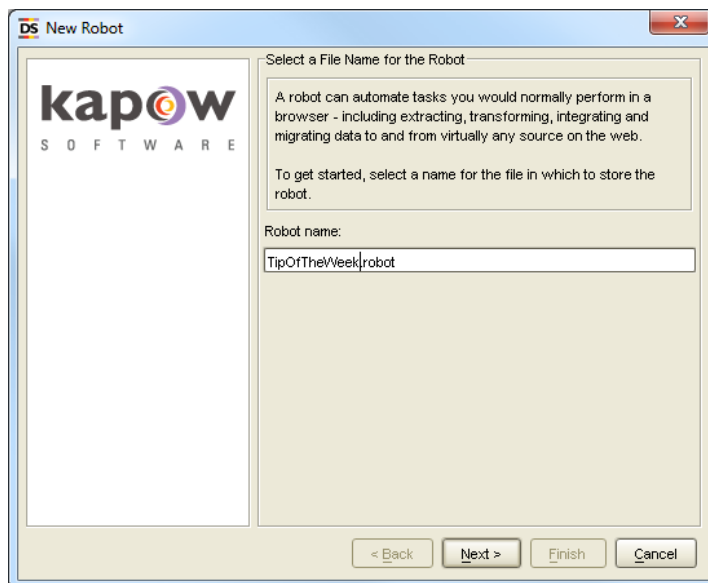
### Step 5 — Robot の新規作成

#### 【操作してみましょう】

プロジェクトビュー（Project View）内の "8.3.1" を右クリックし、"New..." | "Robot" を選択します

#### 【確認しましょう】

「New Robot」ダイアログが表示されますので、作成するロボットのファイル名として「TipOfTheWeek.robot」と入力し、「Next」ボタンをクリックしましょう。

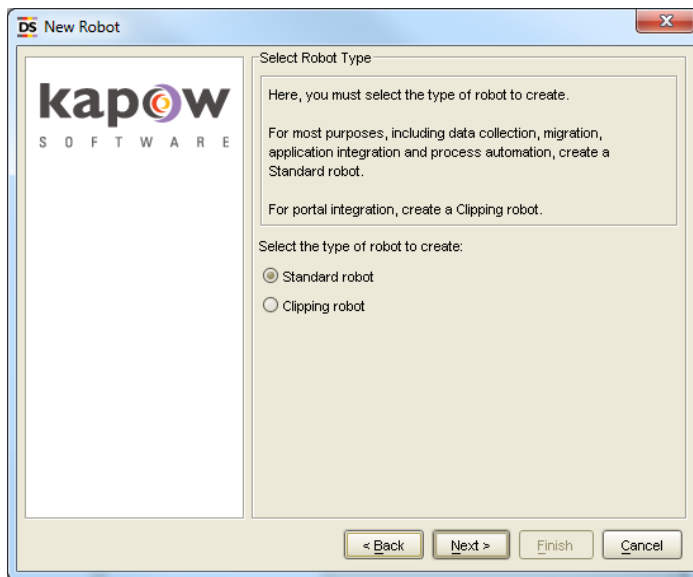


Naming a new Robot

### Step 6 — Robot Type の選択

次にロボットのタイプを選びます。ご利用のライセンスや設定によってこの画面は表示されないかもしれませんが、表示された場合には「Standard robot」を選択し、「Next」ボタンをクリックしましょう。



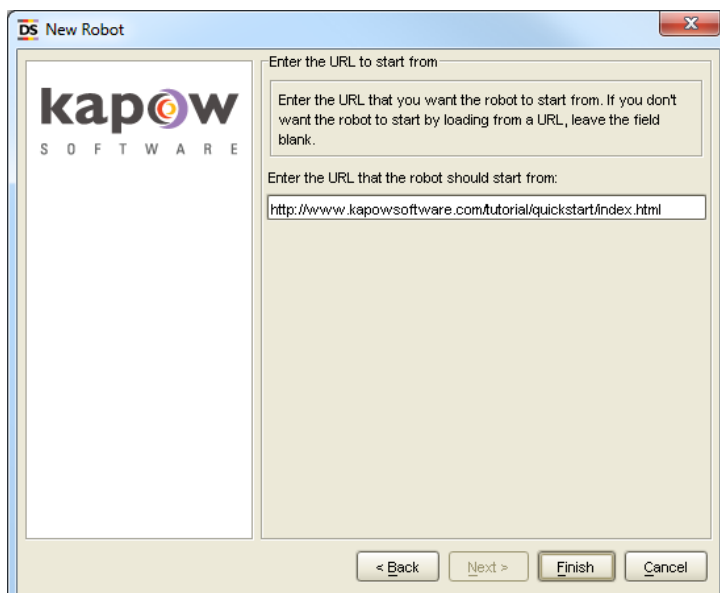


Choosing the Robot Type

## Step 7 — 開始 URL の入力

ロボット作成開始時点で読込むページの URL を入力します。下記の URL を入力し、「Finish」ボタンをクリックしましょう。

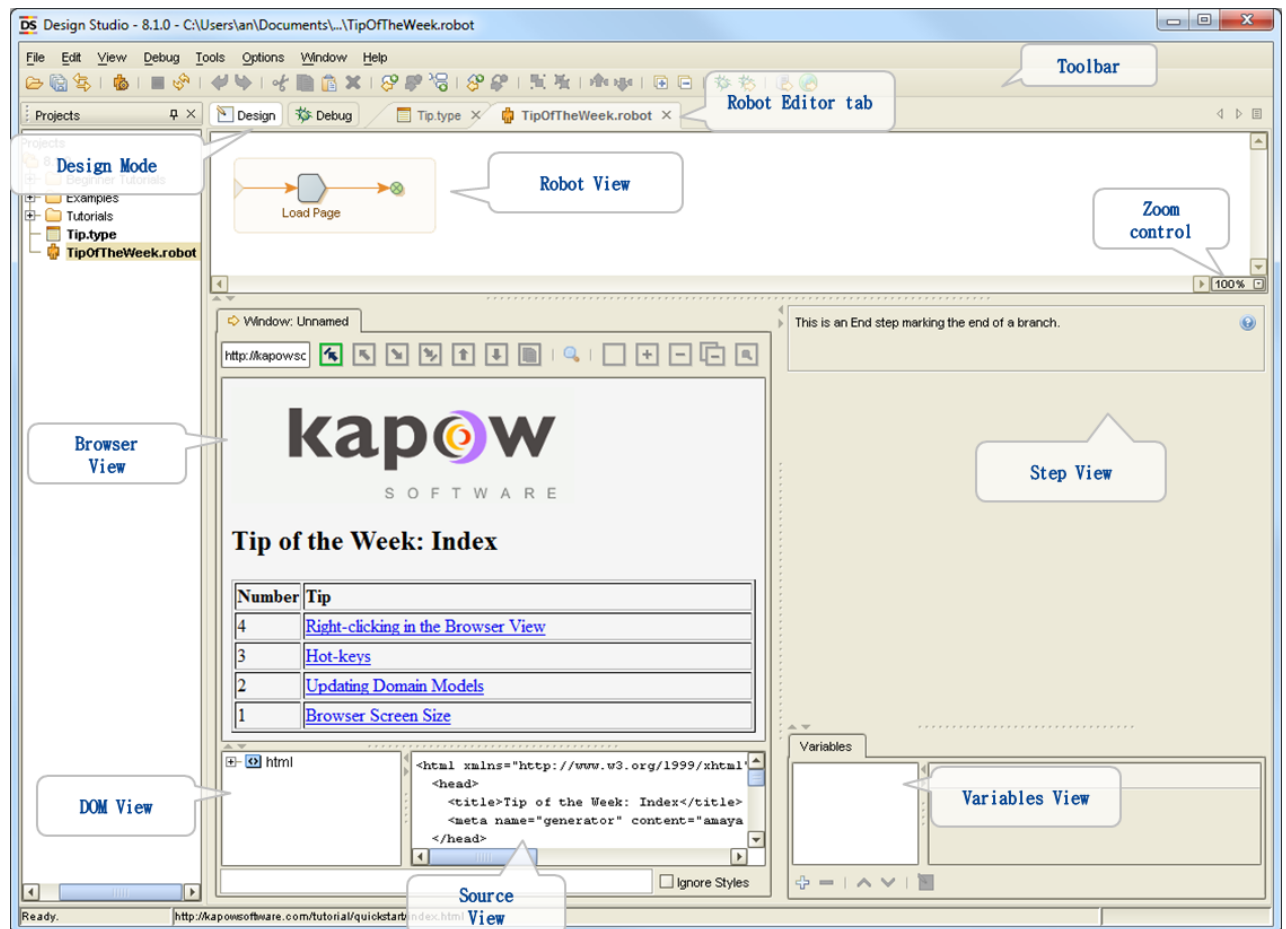
<http://www.kapowsoftware.com/tutorial/quickstart/index.html>



Entering the URL to Start From

## Step 8 — The Robot Editor

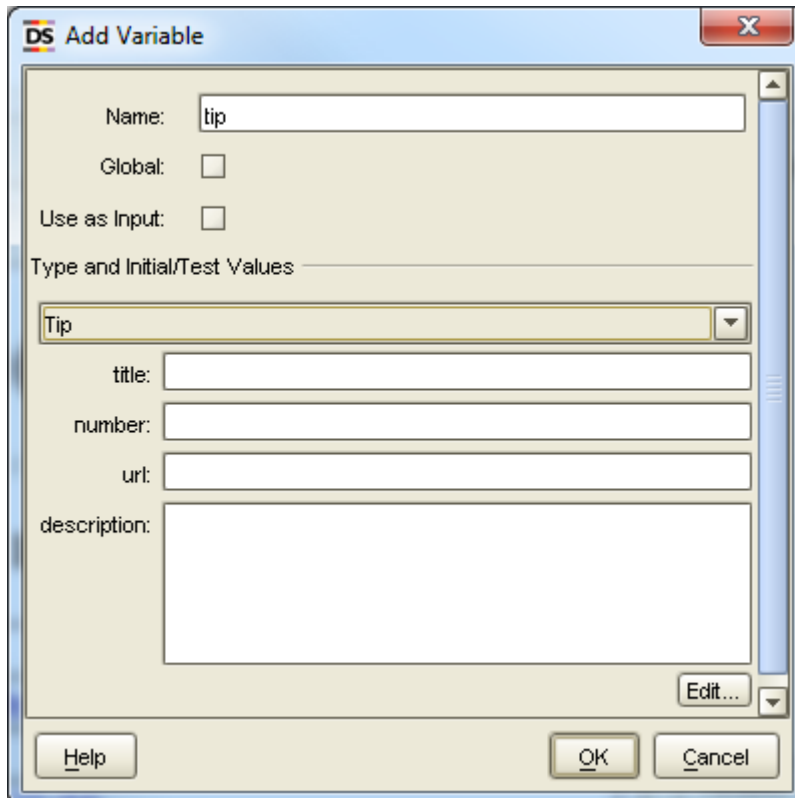
ウィザードの設定が完了すると Design Studio でロボットエディターが開かれます。ロボットエディター内のステップビュー（Step View）を見ると、既に 2 つのステップが作られています。最初のステップは Load Page ステップのアクションを含んでおり、ウィザードで指定した URL のページがブラウザビューに読み込まれた状態で表示されているのが確認できます。2 つ目のステップはエンドステップで、ロボットの処理の終端を示しています。



Robot Editor

## Step 9 — Tip 変数の追加

データを抽出する前に、タイプエディタで先ほど作成した Tip 型変数をロボットに追加しなければなりません。まずはバリアブルビューの下部にある + ボタンをクリックしてください。変数の名前を tip として、type のリストから Tip 型を選択してください。その後 OK ボタンをクリックしてダイアログを閉じます。



Creating a Variable of type Tip

## Step 10 — 表を行単位で繰り返す

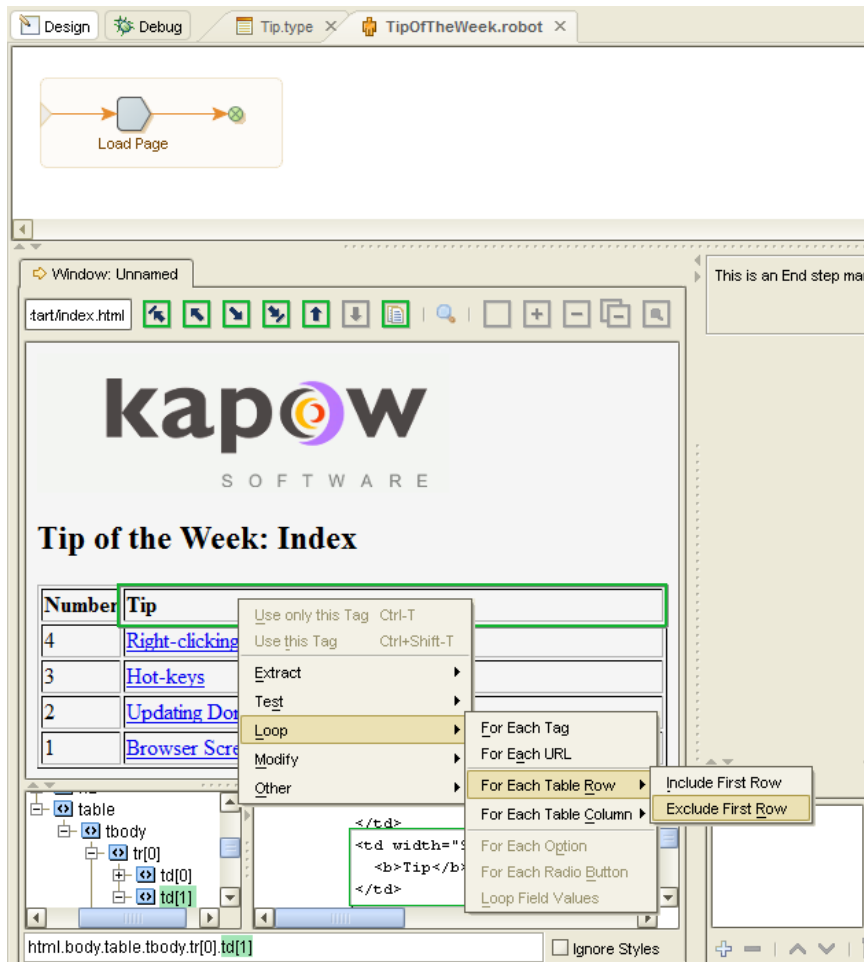
画面中央の表に対して、行数分の繰り返し処理をさせましょう。

### 【操作してみましょう】

ロードされたページには表が表示されます。表に一覧表示された tips の内容を各行一件ずつ順番に抽出したい場合、以下の手順で繰り返し処理を組み立てます。

まずはマウスカールソルを表（テーブル）上のどこかのセルに合わせ、右クリックします。次に表示されたコンテ

キストメニューの中から"Loop" | "For Each Table Row" | "Exclude First Row"を選択します。1 行目 (First Row) は見出し行のため、繰り返しの対象からは除外します。



## Looping Through Table Rows

コンテキストメニュー選択後、「For Each Tag Path」ステップが一つ前のステップに挿入されます。この挿入されたステップがループ (loop) アクションです。ループアクションに対しては、三角矢印ボタンや繰り返し番号表示欄を直接変更することにより、読み込む行をコントロールすることができます。

ループステップ上のどこかをクリックすると、画面右部にステップの設定情報が表示されます。ちなみに、「First Tag Number」ラベルの右側に表示されている\*に注目してください。この\*は設定値がデフォルト値から変更されていることを意味します。(この場合、デフォルト値は 0 : 「見出しも含む」です。) \*もしくはラベルを右クリックすると、ポップアップとして丸括弧で囲われたデフォルト値が表示され、ポップアップをクリックすることによりデフォルト値へ戻すことができます。ここで First Tag Number 項目がデフォルトと異なる理由は、ステップを作成する際にヘッダ行を繰り返し対象から排除するために " Exclude First Row "

を選択したためであり、デフォルトではヘッダ行から繰り返しの範囲となります。詳細については [Show Changes from Default](#) をご覧ください。



Looping Through Table Rows

## Step 11 — Tip 番号の抽出

次に、tip それぞれの“Number”を抽出するステップを挿入しましょう。

その前にエンドステップをクリックして、ステップの色がアクティブステップ（緑色）に変化するのを確認しましょう。

表内、一件目の Tip 行が下のような青い四角形で囲われているのを確認しましょう。この青い四角は、「Current Tag」のしるしになります。



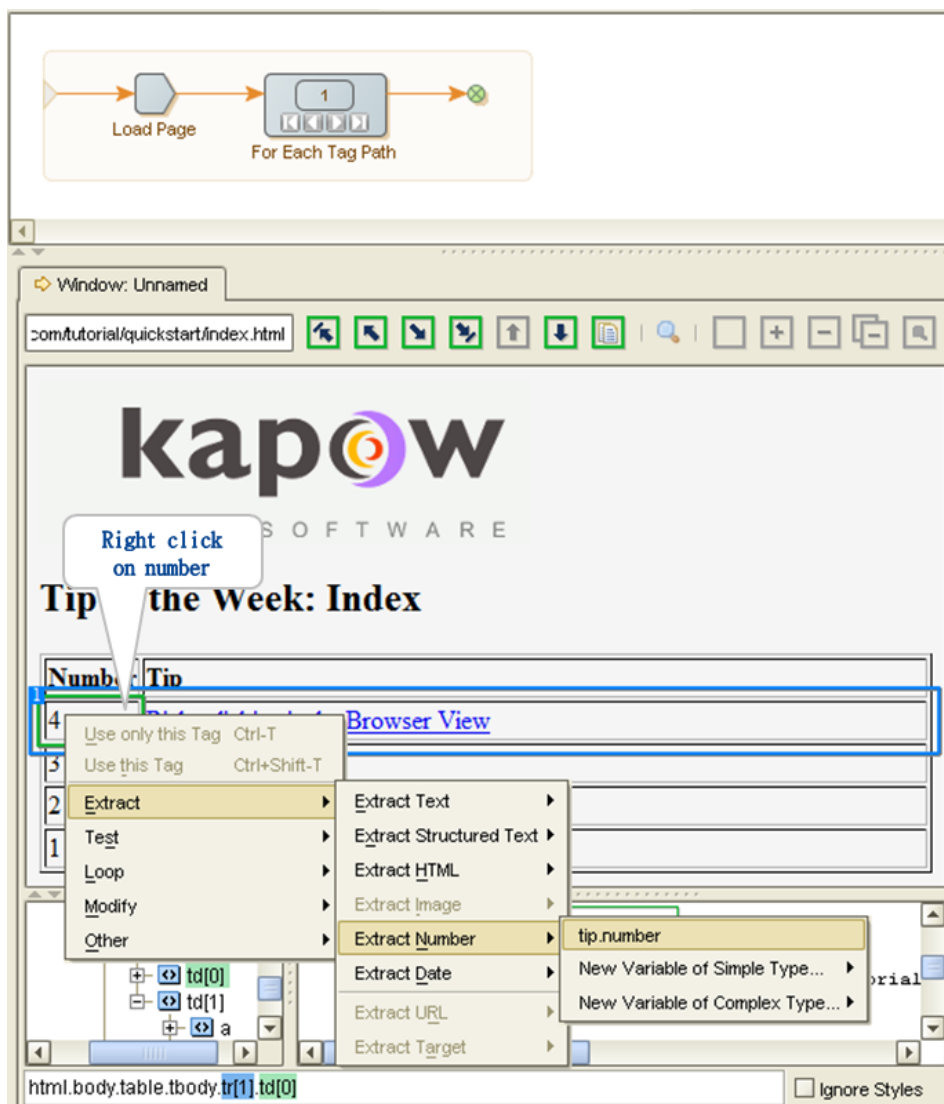
A current Tag

本例の場合、処理時点において表内のどの行を指し示しているのか表すためにループ内でカレントタグ（Current Tag）が作られています。

ループ中の各行で繰り返し実行されるアクションについては、常に青い四角の中をクリックしてタグとアクションを設定することが重要です。

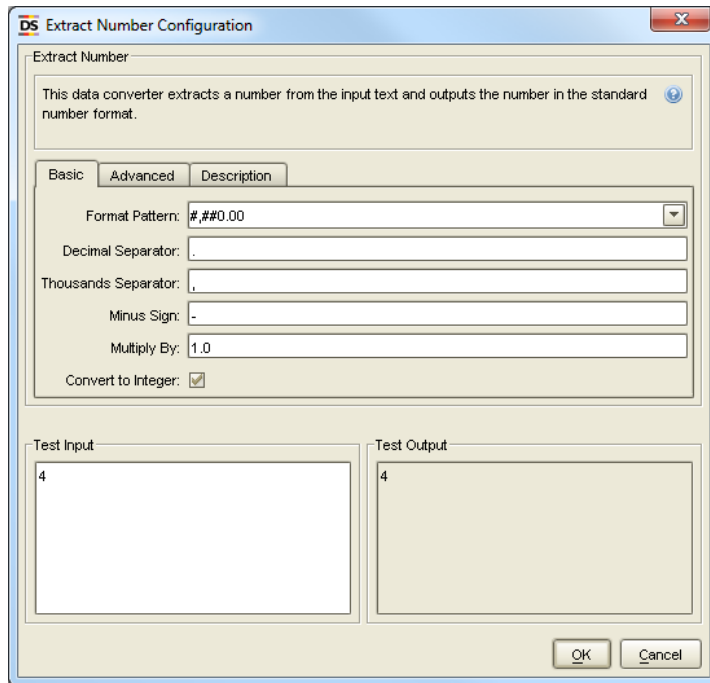
### 【操作してみましょう】

ブラウザビューで、青い四角の中の数字（Number 列）を右クリックします。コンテキストメニューが表示されたら、"Extract" | "Extract Number" | "Tip.number" を選択しましょう。



Extracting Tip Number

「Extract Number Configuration」ダイアログが表示されたら、「Convert to Integer」オプションをチェックします。



Current Tag

「OK」ボタンを押してダイアログを閉じます。画面右下部のバリアブルビューを見ていただくと、今抽出した数値(おそらく 4)が Tip 型の number 属性に設定されていることでしょう。

ループステップのナビゲーションボタンをクリックすると、カレントタグを切り替えることができます。数値の抽出処理を挿入した後でナビゲーションボタンを操作することによって、それぞれの行の数値が正しく収集できているかどうかを確認することもできるでしょう。



Loop navigation

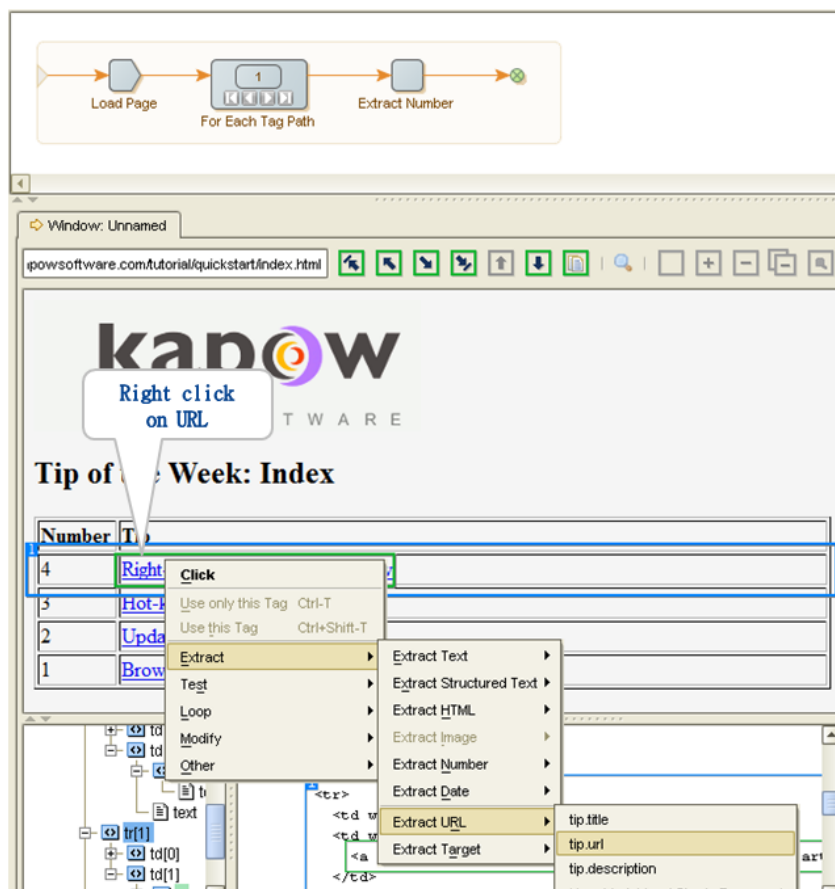
以降の処理に進む前に、必ずナビゲーションボタンを使用してカレントタグを最初の行に戻しておいてください。

## Step 12 — Extract Tip URL の抽出

同様に、Tip の URL も抽出してみましょう。

### 【操作してみましょう】

ブラウザビューで、青い四角の中の文字列（Tip 列）を右クリックします。コンテキストメニューが表示されたら、"Extract" | "Extract URL" | "Tip.url" を選択しましょう。



Extracting Tip URL

## Step 13 — Tip が記述されたページをロードする

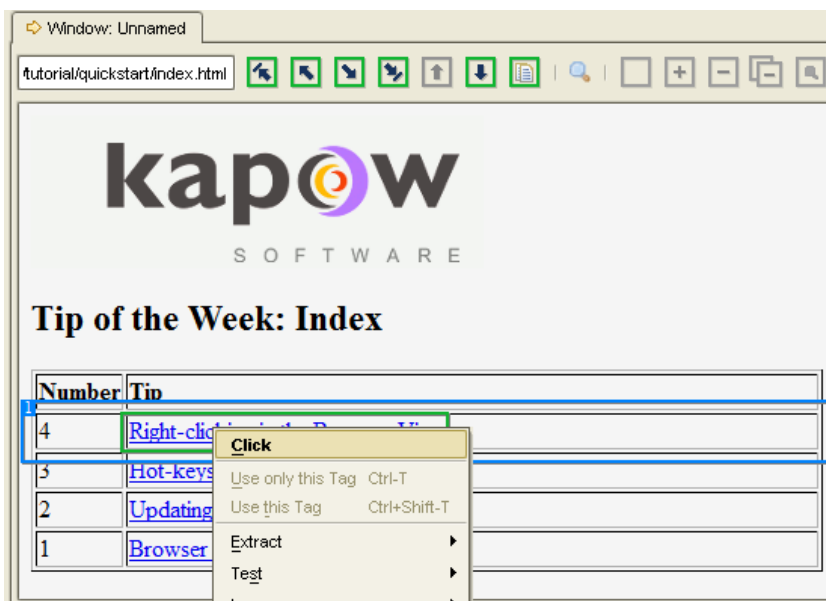
カレントタグが指し示している Tip のより詳しい情報を収集するためには Tip の中身が記述されたページをロードする必要があります。それではそのステップを挿入しましょう。



### 【操作してみましょう】

ブラウザビューで、青い四角の中のタイトルリンクを右クリックします。コンテキストメニューが表示されたら、"Click" を選択しましょう。ちなみにコンテキストメニュー内の **Click** という文字が太字になっていることに注意してください。これは選択したタグに対して Click というアクションがデフォルトで設定されていることを意味しており、ダブルクリックした際に実行される動作を示しています。デフォルトアクションはタグにより様々な動作が設定されており、たとえばテキストボックスにフォーカスしてダブルクリックすれば、入力フィールドが表示されます。

右クリックのステップが挿入され、tip のページがロードされます



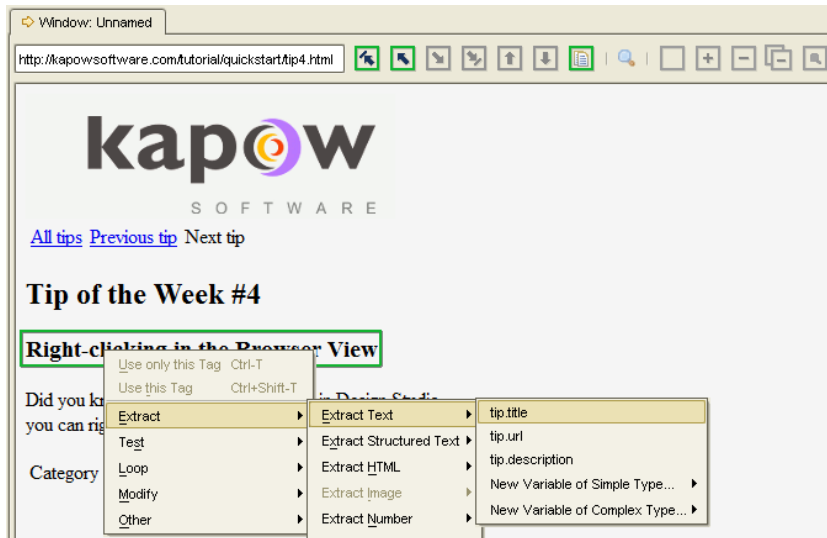
Loading the Page with the Tip

## Step 14 — Tip タイトルの抽出

Tip のタイトルと内容の記述を抽出します。まずはタイトルから抽出しましょう。

### 【操作してみましょう】

ブラウザビューで、Tip のタイトルを右クリックします。コンテキストメニューが表示されたら、"Extract" | "Extract Text" | "Tip.title" を選択しましょう。



Extracting Tip Title

## Step 15 — Tip 記述の抽出


最後に、Tip の記述を抽出します。

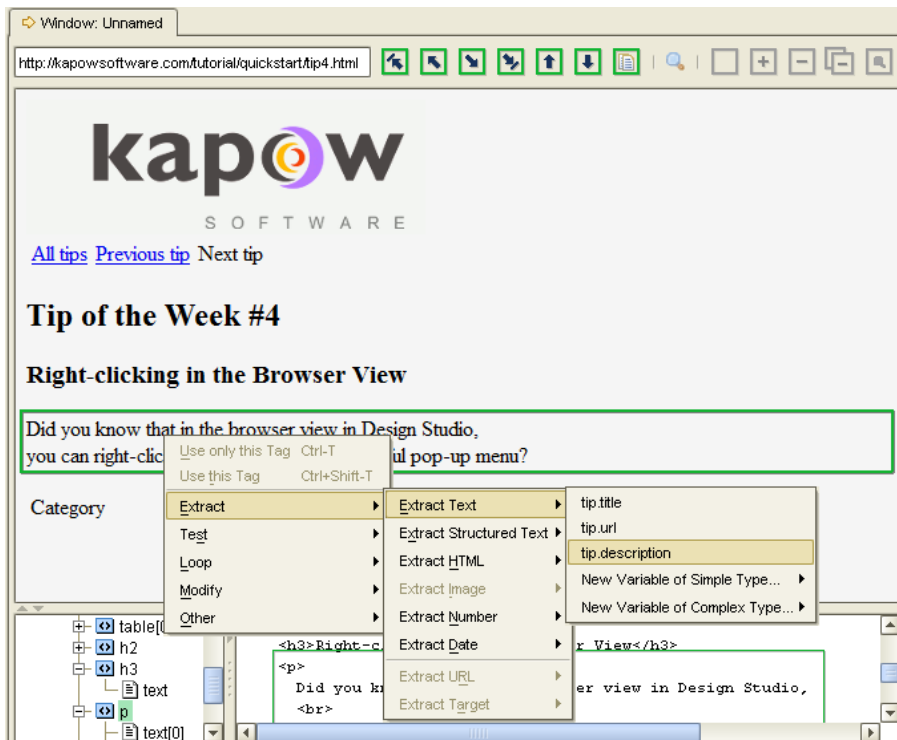
### 【操作してみましょう】

ブラウザビューで、Tip の記述をクリックします。今回の場合、緑色のボックスが tip の記述の全範囲を選択していないことに注意しましょう。



Extracting Tip Description

ツールバーの  ボタンをクリックして、緑色のボックスが tip の記述全体を選択するように範囲を広げましょう。その後、緑色のボックス内で右クリックをし、コンテキストメニューから "Extract" | "Extract Text" | "Tip. description " を選択しましょう。




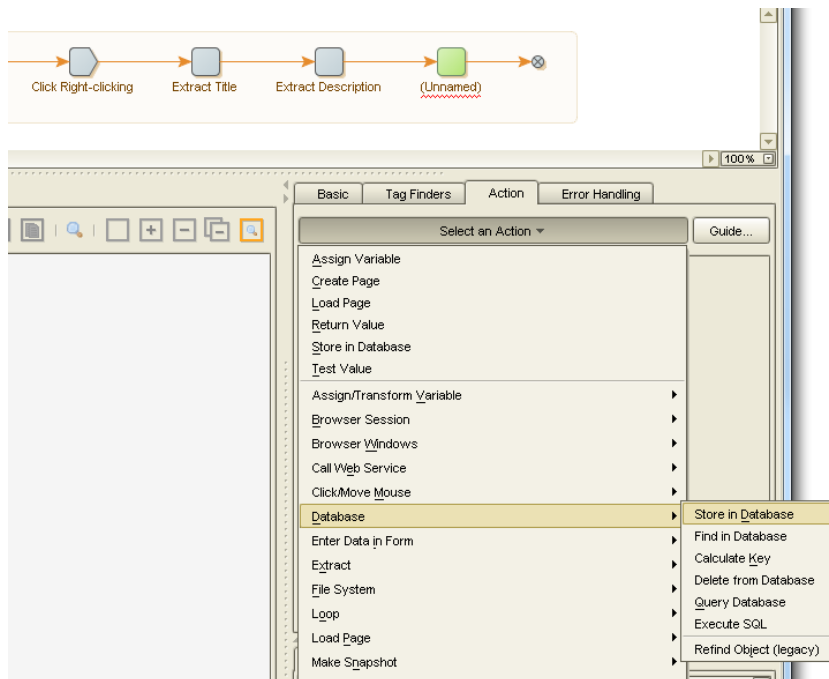
Extracting Tip Description

## Step 16 — データベースへ格納

Tip に関する 4 つすべての属性が抽出されたので、次にそれらのデータをデータベースへ格納します。エンドステップを選択し、手前にステップを追加してください。

### 【操作してみましょう】

エンドステップ（右端のステップ）をクリックしてアクティブステップ（緑色）にしてください。ツールバーの  ボタンをクリックし、エンドステップの手前に新しいステップ（Unnamed）を挿入してください。次に（Unnamed）ステップを選択し、アクションタブのドロップダウンリストから「Store in Database」アクションを選択します。



## Store In Database

それでは「Store in Database」アクションの設定をしましょう。

まず始めに、データベースを選択する必要があります。Kapow Katalyst にはデモ用のデータベースがインストールされており、デモロボットのデータ格納先として利用できます。デモ用のデータベースとして "objectdb" が用意されているので、ドロップダウンリストから選択してください。

次に、データベースへ格納する型を選ぶ必要があります。"Tip" 型を Variable ドロップダウンリストから選択してください。

最後に、型がデータベースへ格納される際のキー項目を選ぶ必要があります。キーはデータを格納する際の識別子であり、データベース内のデータに重複がないかチェックをするために利用されます。またキーはデータベースからデータを探し（「Find in Database」）たり、削除（「Delete from Database」）したりする際の、データ識別子としても利用されます。キー項目の初期設定値は、型で定義されているキーによって決定されます。つまりどういうことかという、型定義の中で "Part of Database Key" がチェックされている属性の値が自動的に計算されてキー項目値が算出されるということです。

「Store in Database」ステップに赤色のエラーアイコンが表示されている場合は、ステップの設定に不備があることを表しています。この場合、マウスのカーソルをステップの上に移動すると、発生している不備の内容がポップアップで表示されます。今回エラーは、データベースに格納しようとしている型の中に "Part of Database Key" としてチェックされているものがないため、キー項目が利用できないというものです。その

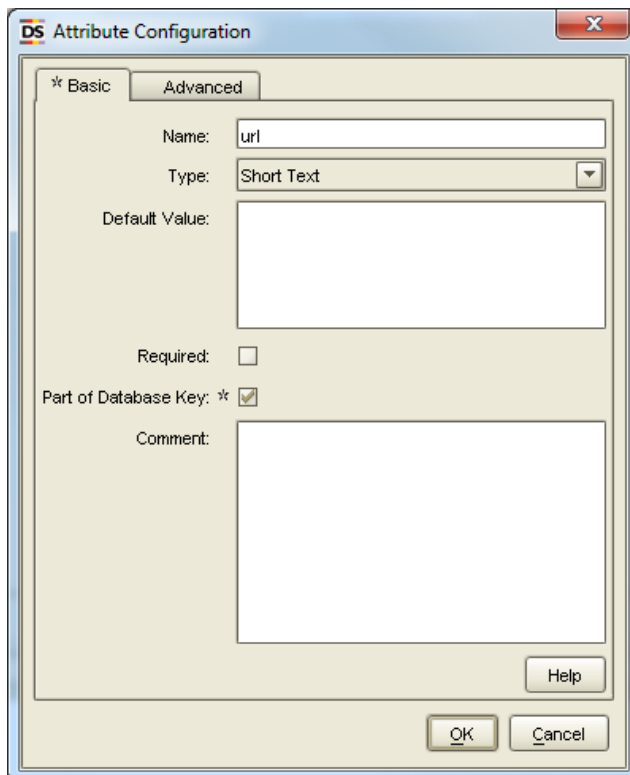
ためエラーを解決するためには、型定義を修正する必要があります。

## Step 17 — Tip 型の修正

### 【操作してみましょう】

“Tip.type”と表示されているタブをクリックし、タイプエディターへ表示を切り替えましょう。


“url” 属性をダブルクリックし “Part of Database Key” のチェックボックスを ON にした後、“OK” ボタンをクリックしてください。（ここで url 属性（url 属性のみ）をキー項目に利用する理由としては、tip 毎に URL がユニークな値だからです。）



Modify URL attribute

その後ロボットエディタへ再度切り替えると、赤色のエラーアイコンが消えていることと思います。これで “Store in Database”アクションは正しく設定されました。

## Step 18 — 保存します

ロボットはこれで完成です。また、モデルも修正していますので、ツールバー内の全て保存ボタン  をクリックしてください。

## ロボットを動かす

ロボットのデザインが完了し、やっとそれを動かす時がきました。まずは Design Studio 内のデバッガーでロボットを動かしてみましょう。動作上、特に問題がないようであれば、実際にデータベースにデータを登録するロボットを、スケジューラを使って動かします。なお、スケジューラについては、Tutorial2 で、次回行います。

## Step 19 — デモデータベースの開始

“Step 16 — データベースへ格納” の中でロボットが“objectdb”（Kapow Katalyst に最初からインストールされているデータベース）へデータを格納するように設定しましたが、データベースを利用したロボットを実行する際には必ずデモデータベースを起動しておいてください。データベースが動いていない場合、ロボットの実行は失敗してしまいます。

Windows OS の場合、スタートメニュー内の“Kapow Katalyst 8.3.1” | “Demo Database” の配下に “Start Demo Database” という名前のショートカットが用意されています。

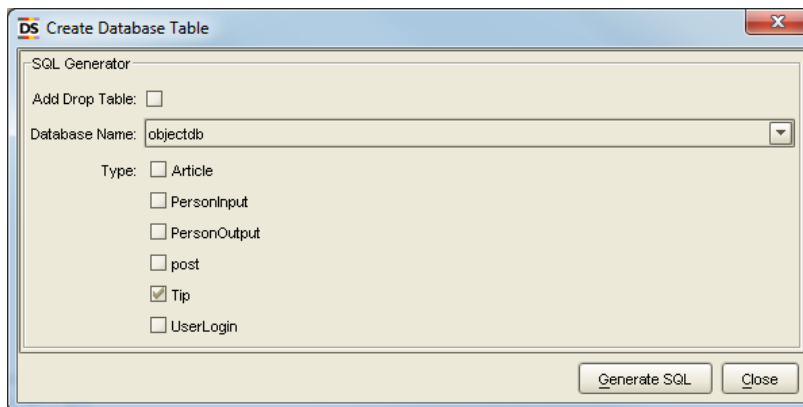
Linux OS の場合、インストールフォルダ内の bin フォルダにある “StartDemoDatabase” を実行する必要があります。

## Step 20 — Tip 型を格納するデータベースのテーブルの作成

型の内容をデータベースへ格納するためには、データを格納するためのテーブルをデータベース内に持っていない必要があります。これらのテーブルは Design Studio を使って作成することができます。

### 【操作してみましょう】

“Tip.type”と表示されているタブをクリックし、タイプエディターへ表示を切り替えましょう。“Tools”メニュー内の“Create Database Table” をクリックすると、テーブル作成画面（“Create Database Table”）が表示されます。Database Name から“objectdb”を選択し、Type から“Tip”を選択して下さい。この段階ではまだ既存の Tip テーブルは存在しないので、“Add Drop Table”にチェックを入れる必要はありません。



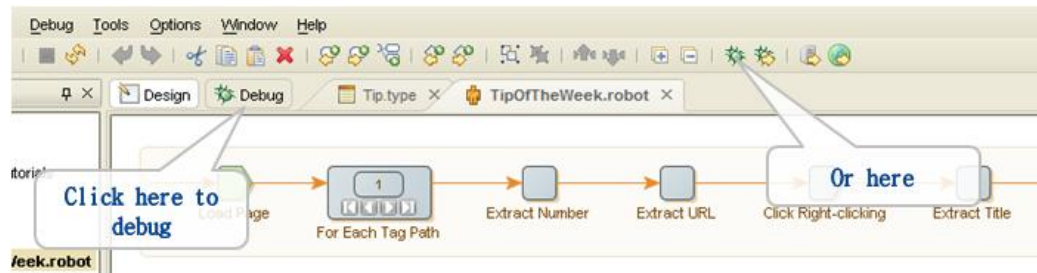
Create Database Table

「Generate SQL」ボタンをクリックすると、Tip 型を格納するテーブルを作成する SQL が表示されます。次に「Execute」ボタンをクリックするとテーブルが作成され、ポップアップにより問題なくテーブルが作成されたことを通知します。

※ 本画面については、使用しているバージョンまたはライセンスによっては表示されないこともあります。

## Step 21 —Design Studio デバッガーによるロボットの実行

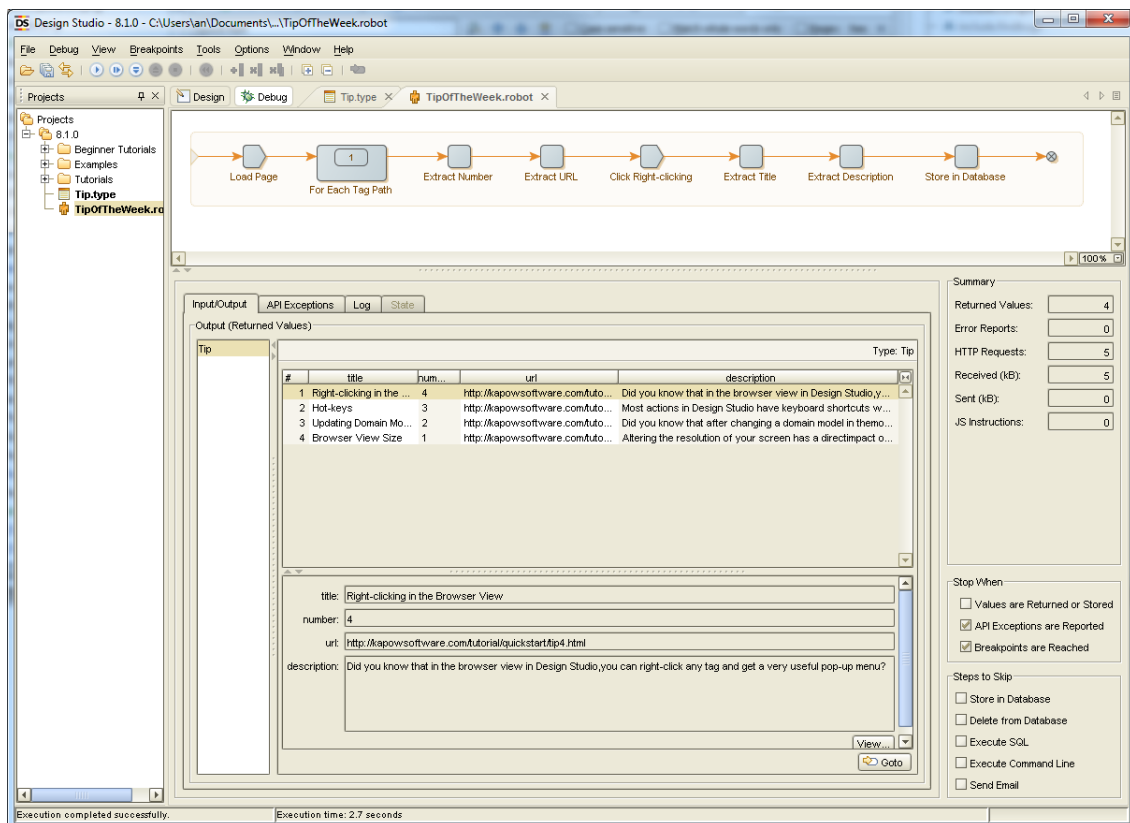
“TipOfTheWeek.robot”と表示されているタブをクリックし、ロボエディタへ表示を切り替えましょう。その後デバッグモードへ切り替えてください。



Switch to debug mode

タイプエディターからロボエディターへの切り替えや、ロボエディターのデバッグモードへの切り替えで画面の表示がどのように変わるか注意して見てみて下さい。

🔍 ボタンをクリックしてデバッガーからロボットを実行してください。抽出された型が都度デバッガーの Input/Output タブ内に表示されることでしょう。



Running in Debug Mode



## まとめ

本チュートリアルでは、シンプルなロボットの作り方、およびデバッガーでの動かし方について解説しました。

次の Tutorial では Management Console を使ってロボットを定期的に行う方法について解説します。

また、KapowKatalyst は API を利用してロボットを全異なるソフトウェアシステムの中に統合することも可能です。API に興味のある方は [Programming With Robots](#) をご覧ください。

以上  
2012 年 7 月