The Elements of an Info Management System

By Bob Boiko

Senior Lecturer, University of Washington iSchool

[www.iSchool.washington.edu](https://exchange.washington.edu/owa/redir.aspx?URL=http%3a%2f%2fwww.iSchool.washington.edu) \* [bboiko@u.washington.edu](mailto:bboiko@u.washington.edu)

President, Metatorial Services Inc.

[www.metatorial.com](https://exchange.washington.edu/owa/redir.aspx?URL=http%3a%2f%2fwww.metatorial.com) \* [bob@metatorial.com](mailto:bob@metatorial.com)  \* 206 706 3078

Founder LearnTogether \* <http://learnTogether.io>

Contents

[Step 1: Understand what we will do 2](#_Toc440346957)

[Step 2: Download and unzip BTS 2](#_Toc440346958)

[Step 3: Run webHomePage.html 4](#_Toc440346959)

[Step 4: Run webAnimalFullViewPage.html 5](#_Toc440346960)

[Step 5: Understand the infoBase 6](#_Toc440346961)

[Step 6: Use the infoBase 8](#_Toc440346962)

[Step 7: Analyze the webAnimalFullView template 8](#_Toc440346963)

[Step 8: Analyze the webHomePage template 9](#_Toc440346964)

[Step 9: Analyze and Run printFactSheet.html 10](#_Toc440346965)

[Step 10: Conclusion 12](#_Toc440346966)

## Step 1: Understand what we will do

In this exercise you will run through all the basic parts of an Info Management System.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_s create Web pages and other presentations from a presentation neutral repository that we call an InfoBase.

\_\_\_\_\_\_\_\_\_\_\_\_\_ is a *logical* concept. It is a term we use for a variety of *physical* storage places such as relational databases and XML files. In this course we focus mainly on infoBases that are \_\_\_\_\_\_\_\_\_\_\_\_\_\_.

In this exercise, however, we will use a different InfoBase called a JSON file. \_\_\_\_\_\_\_\_\_\_\_\_\_\_s are today’s most popular way to send information to a Web page. We will use a \_\_\_\_\_\_\_\_\_\_\_\_\_\_ as our \_\_\_\_\_\_\_\_\_\_\_\_\_\_ in this exercise because they are cool (I want you to be exposed to them) and because they fit very easily into the Basic Template System (BTS).

The \_\_\_\_\_\_\_\_\_\_\_\_ demonstrates all the basic parts of an \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_. In this exercise you will learn how the presentation neutral info in an \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ is transformed into a user-ready presentation such as a Web or printed page.

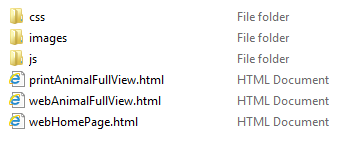
## Step 2: Download and unzip BTS

BTS is an ultra-simple templating system that demonstrates all the basics of multi-channel publishing. We will use this software throughout this exercise.

Templating is at the core of any multi-channel system. Any templating system, ours included, has these parts:

* **Templates** that hold the look and feel and layout of publications
* **A Template Processor** that is the program that combines the templates and the InfoBase.
* **Styling** which works in conjunction with the templates to create look and feel
* **An InfoBase** that holds all the information the system needs to fill any channel
* **Media assets** that are images and other files that are part of the information but outside the InfoBase.

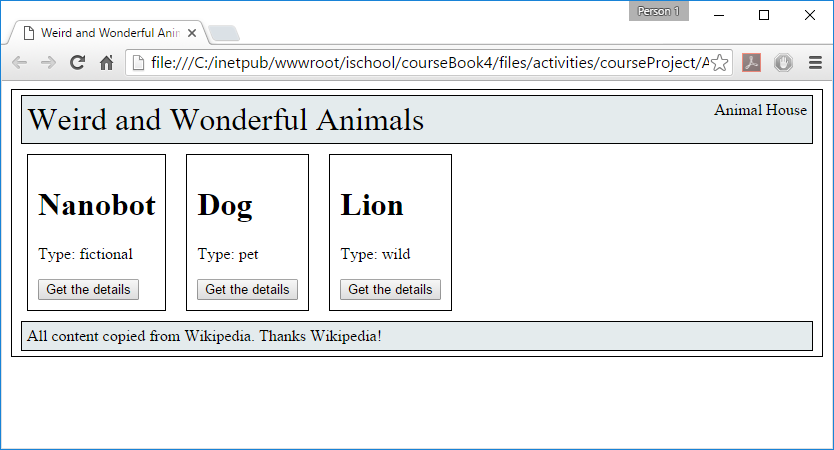
To Do

1. Download the BTS from the courseBook
2. Unzip it into a set of folder that should look like this.  
     
   
3. Look over all the files and complete the table below

All the files in the BTS system are listed in the table below. For each file, choose from the list above to state the purpose of the file.

|  |  |
| --- | --- |
| File | What kind of file? (template, processor, InfoBase, styling, media) |
| webHomePage.html |  |
| tplProcessor.js |  |
| animalHouse.css |  |
| dog1.jpg |  |
| infoBase.js |  |

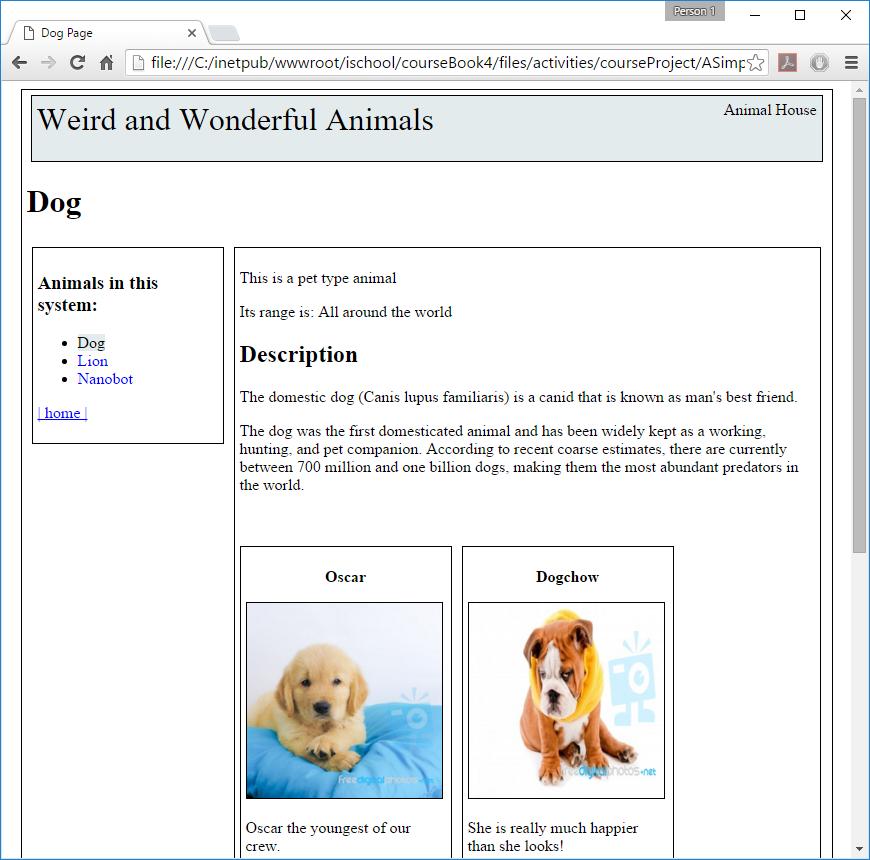
## Step 3: Run webHomePage.html

1. webHomePage.html is a (template, processor, InfoBase, styling, media) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ file
2. Open webHomePage.html in any browser
3. If it does not look something like the image below, ask for help  
     
   
4. On the image above, identify and label the box that contains these elements

* The page
* The header
* The content area
* A partial view of an animal
* An attribute of an animal
* The footer

1. Click around to see what happens.
   * Which links work? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
   * Which links do not seem to work? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Step 4: Run webAnimalFullViewPage.html

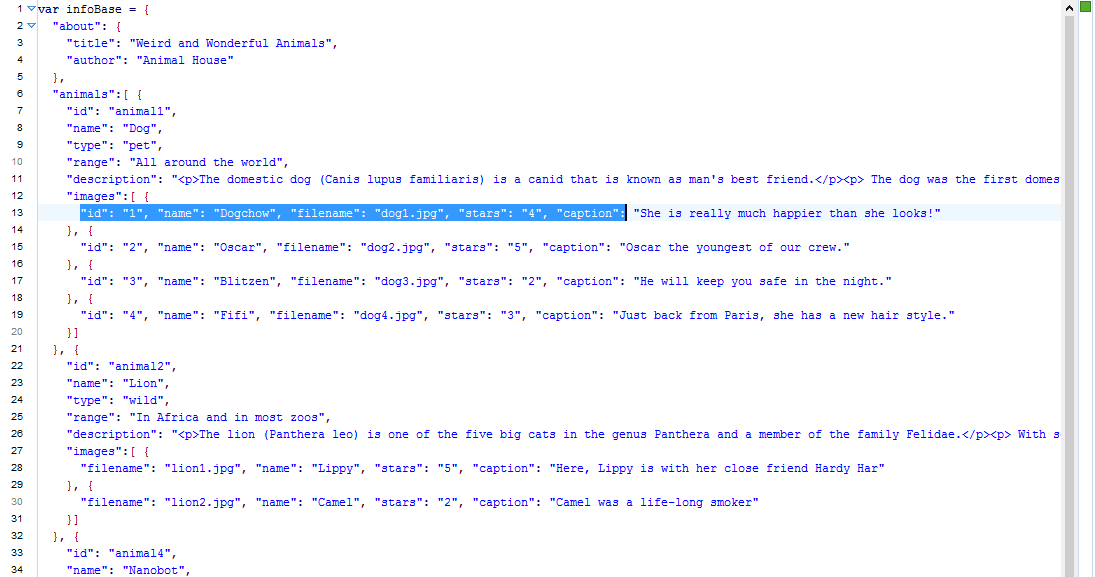
1. webAnimalFullViewPage.html is a (template, processor, InfoBase, styling, media)   
     
   \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ file.
2. On webHomePage.html, click the button that says “Get the Details” for Dog   
   The system will display webAnimalFullViewPage.html with the Dog item showing
3. If it does not look something like the image below, ask for help  
     
   
4. On the image above, identify the box that contains these elements
   * The page, The header, The navigation area
   * The animal item area
   * A full view of an animal item
   * The same attribute of an animal that you identified in webHomePage.html
   * The footer
5. Fill out the table below for each part of the URL that is displayed in the URL area of the browser. On my computer the URL looks like this:

file:///C:/Users/bboiko/Documents/\_active/\_Projects/hartman2015/animalHouse/fullViewAnimal.html#/animals/id=animal1

|  |  |
| --- | --- |
| Part of URL | What is its purpose |
|  | This is the path to the HTML file you are displaying on your hard drive |
|  | This is the name of the HTML file that is displayed |
|  | This is the “hash” or “pound” symbol |
|  | This tells the page to display the animal whose id is “animal1” |

## Step 5: Understand the infoBase

1. infoBase.js is a (template, processor, InfoBase, styling, media) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ file.
2. Open infoBase.js in oxygen.  
   The file should look something like this:



1. In the image above, identify these parts

* The system information (about)
* An info type
* An info item
* An attribute of an item
* The value of an element of an info item

## Step 6: Use the infoBase

1. Using the other animal items in infoBase,js as a guide, fill out the Nanobot item. Notice that there are already Nanobot images ready for you to use.
2. Display webHomePage.html again and click “Get the Details” for the Nanobot item. Do you now have a working Nanobot fullView page? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
3. Go back to infoBase.js and using the other animal items as a guide, create a new item of type animal using the animal of your choice.
   * Get at least 3 images from the Web
   * Get the description from Wikipedia or another source of your choice
4. Display webHomePage.html again.
   * Did your new animal display? \_\_\_\_\_\_\_\_\_\_\_\_\_\_
   * Can you click on the new animal’s “Get the Details” button? \_\_\_\_\_\_\_\_\_\_\_\_\_\_
   * Does the fullView page display for the new animal? \_\_\_\_\_\_\_\_\_\_\_\_\_

## Step 7: Analyze the webAnimalFullView template

**All templates retrieve information from an InfoBase and place it in the page.**

The BTS system uses the following kinds of commands to retrieve and place information:

**{%%}** commands to include attribute values in the HTML. For example, the command {%pageItem:?name%} puts an animal’s name on the page.

1. Open webAnimalFullView.html in oxygen (not in your web browser).
2. Fill out the table below to identify {%%}commands and what they do.

|  |  |  |
| --- | --- | --- |
| **Command** | **Line (s) it is found on in the file** | **What does it do?** |
| {%pageItem:?name%} | 4, 15 | puts an animal’s name on the page |
| {%/about/?title%} | 11 | puts the title under “about” |
| {%/about/?author%} | 12 | Puts the author under “about” |
| {%?name%} | 20, 40 | Puts name in navigation area |
| {%pageItem:?range%} | 30 | Puts range on the page |
| {%pageItem:?description%} | 33 | Puts the description on the page |
| {%?caption%} | 42, 44 | Puts caption on the page |
| {%?stars%} | 45 | Puts the stars on the page |

1. Change {%?caption%} to {%?stars%}. Run the page  
     
   What happens? ---The caption changed to stars

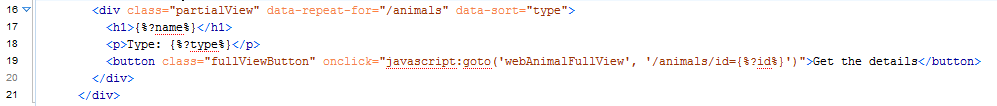
## Step 8: Analyze the webHomePage template

**Templates can repeat a part of the screen multiple times**

1. Display webHomePage.html in your browser.
2. What part of the page is repeated multiple times?

Partial View

1. Open webHomePage.html in oxygen (not the browser). Identify the following block of HTML



Note the following components in the block

* Line 16 has a div (a div is simply an area on a web page. This is the part of the page that will repeat
* Inside the div (in the area) you will see **data-repeat-for="/animals**. This is a command that says “repeat this div for each animal.”
* Inside the div you will see **data-sort="type"**. This is a command that says “sort the animals on the page alphabetically by the animals’ type
* All the lines inside the div (17-19) are the ones that will be repeated when the template runs in the browser.

1. In the design of the Web page, what purpose does the div serve?   
   Division of the page and allow us to create partial pages

Suppose the InfoBase was about plants instead of animals:

1. You would need to change animals to plants In order to make the div work
2. You would need to change data-sort=”type” to data-sort=”name”  
   In order to make the plants in alphabetical order by name
3. Change **data-sort="type" to data-sort="name".** Predict what will happen on the page when   
     
   you run it again? The plants are shown in alphabetical order by name

## Step 9: Analyze and Run printFactSheet.html

1. printFactSheet.html is a (template, processor, InfoBase, styling, media)   
     
   template file.
2. Suppose the URL to display this page is:   
     
   <file:///C:/hartman2015/animalHouse/printAnimalFullView.html#/animals/id=animal1>  
     
   Fill out the table below for each part of the URL

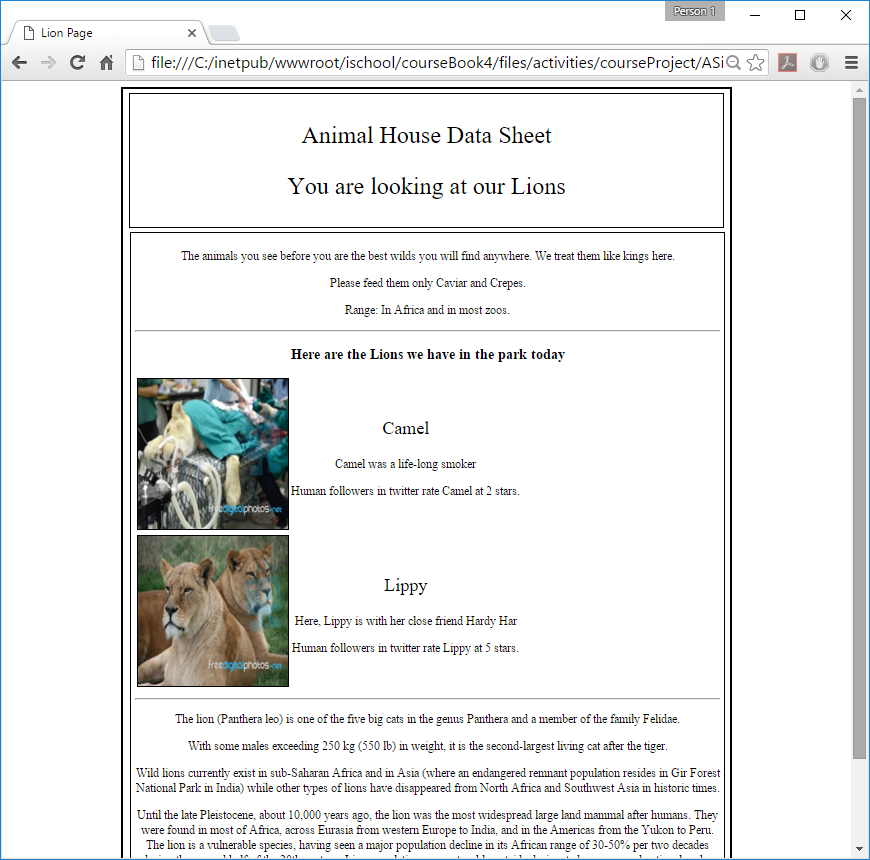
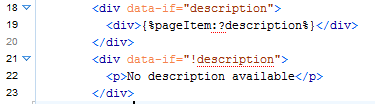
|  |  |
| --- | --- |
| File | What is its purpose |
| file:///Users/MEG/Documents/2rd%20quarter%20classes/542/animalHouse/printAnimalFullView.html#/animals/id=animal1 | This is the path to the HTML file you are displaying on your hard drive |
| printAnimalFullView.html | This is the name of the HTML file that is displayed |
| # | This is the “hash” or “pound” symbol |
| id=animal1 | This tells the page to display the animal whose id is listed |

1. Enter the correct URL in your browser to make the printFactSheet.html display the lion.
2. How is this template the same as webAnimalFullViewPage.html?

The information content presented is the same.

1. How is this template different from webAnimalFullViewPage.html?

The information is presented in a more loose way(different format)

1. Change printFactSheet.html so that the animal images are sorted by the name   
   Paste in the modified line below  
   <tr data-repeat-for="pageItem:images" data-sort="name">
2. Change printFactSheet.html so the display looks like this:  
     
   Paste the change below:  
   <p>Range: In the wild, {%pageItem:?name%}s are found {%pageItem:?range%}.</p>
3. Make any change to printFactSheet.html so that the layout looks like this:  
     
     
     
   Paste the change below:  
    </table>  
    </div>  
    <div data-if="description">  
    <div>{%pageItem:?description%}  
    </div>  
    <div data-if="!description">  
    <p>No description available</p>  
    </div>  
    <div class="footer">Content courtesy of Wikipedia</div>  
    </div>
4. What do these lines of the template do?  
     
     
     
   Show the description content of the page item. It means if there are descriptions in the InfoBase, the description will be pulled out from InfoBase. If not, there will be shown “No description available” in the description content area.

## Step 10: Conclusion

The BTS system is as simple as I could make it and still illustrate the main concepts of an information system.

To review, a template is processed by a template processor to display a built page in   
   
your browser. Content on the page is drawn in from a InfoBase and from media files.

\*\*\*\*\*

The info in the InfoBase is neutral. When it is drawn in by commands in the template  
  
and displayed using the formatting instructions in the CSS file, the neutral info is composed

into a useful screen for users.   
\*\*\*\*

While our InfoBase has only a few items, of one type. Generally an InfoBase will  
  
have a number of different types, each with many items.

\*\*\*\*  
Completed screens contain “views” into the InfoBase. A full view has most or all of the info elements  
  
from a single items. A partial view shows only a few elements of a type, but there are often   
  
many partial views on the same screen, each showing a different items.

\*\*\*\*  
By combining partial and full views on a page along with navigation, an entire web   
  
site can be created. But with a simple change of template the same information in the   
  
infoBase can be used to create any number of different presentations.