Document for Bright Edge Coding Assignment

1. Instruction for running the jar program:

Step 1: Extract the zip file and go to the directory of Assignment.jar

Step 2: With network connection, run the command to see the query result.

Query 1: (requires a single argument) java -jar Assignment.jar <keyword> (e.g. java -jar Assignment.jar "baby strollers")

Query 2: (requires two arguments) java -jar Assignment.jar <keyword> <page number> (e.g. java -jar Assignment.jar "baby strollers" 2)

Step 3: If the program executes well, you will see the screenshots as following:

```
jiashengqiu$ java -jar Assignment.jar "digital camera" 1
Query input: digital camera
Query URL: http://www.walmart.com/search/search-ng.do?search_query=digital+camera&ic=16_0
Number of total restults:953
----------------------------------------
Current page:1

Product Title: Canon Black EOS Rebel T3 12.2MP Digital SLR Camera Kit with Two Lenses, SD Card, Bag
Description: 12.2 megapixel resolution EF-S 18-55mm IS II and EF 75-300mm III lenses Bag & SD Card Included
Price: $449.00

Product Title: Nikon COOLPIX L830 Ultra Zoom Digital Camera with 16 Megapixels, 34x Optical Zoom with 4-136mm Lens (Available in mu
ltiple colors)
Description: 16 megapixel resolution NIKKOR 4-136mm zoom lens 3.0" TFT LCD display Full HD 1080p Movie Recording
Price: $196.99 - $299.00

Product Title: Nikon COOLPIX S3600 Digital Camera with 20.1 Megapixels and 8x Optical Zoom (Available in multiple colors)
Description: 20.1 megapixel resolution NIKKOR 4.5-36mm zoom lens 2.7" TFT LCD display 720p HD Movie Recording
Price: $139.95(List Price)

Product Title: Fujifilm AX655 16MP Digital Camera with Memory Card Value Bundle
Description: Comes with: FUJIFILM AX655 Digital Camera with 16 Megapixels and 5x Optical Zoom V7 8GB Class 4 SDHC Memory Card
Price: $62.88

Product Title: Olympus Black VG-180BLK Digital Camera with 16 Megapixels and 5x Optical Zoom
Description: 16 megapixel resolution Olympus 4.7-23.5mm zoom lens 2.7" TFT LCD display Advanced Face Detection Technology
Price: $54.99

Product Title: Sony Black NEX3N/BMBDL Compact System Digital Camera Bundle with 16.1 Megapixels, 8GB SD Card, Case, and 16-50mm Len
s Included
Description: 16.1 megapixel resolution Sony SELP1650 lens 3.0" TFT LCD display 1080p Full HD Movie Recording
Price: $339.98

Product Title: Vivitar Pink VF128-PNK Digital Camera with 14.1 Megapixels and 4x Digital Zoom
Description: 14.1 megapixel resolution 2.7" TFT LCD display
Price: $38.99
```

(The screenshot is for Query2 scenario)

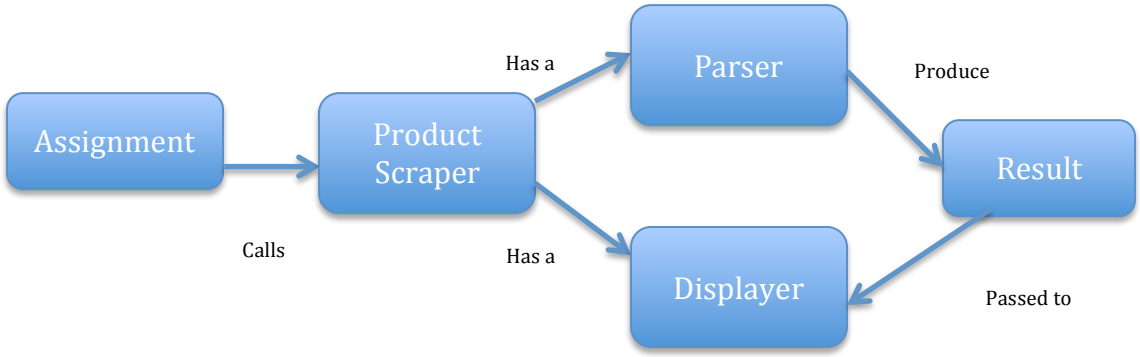
2. File Description:

- Assignment.jar: Containing executable jar file
- Src: Containing the source code for the project.
- Jsoup.1.7.3.jar: the Jsoup library used in parsing the HTML

* Others: generated Java doc

3. Key components of the project

The major functionality of the program includes taking the user query input, getting search result page from Walmart.com, parsing the content in the result page and displaying the results to console to users. Below is a class diagram to illustrate the program design:



- **Assignment.java**: The main program to run the project.
- **Product Scraper.java**: The scraper object, user can set the parser and displayer for the scraper.
- **Parser.java**: Object used to parse the html documentation.
- **WalmartParser.java**: Implementation of the parser for Walmart.com
- **Result.java**: Object used to store the parsing result, it contains a list of products, query, query url, page number for the query. (If there is no page number passed into the query, the page number would be 0)
- **Product.java**: Object used to store a single piece of product information, including product name, product description and price.
- **Displayer.java**: Used to display the result to console.
- **ParserTest**: Unit Test program for Parser

4. Parsing implementation

Below is a very brief summary of what WalmartParser do.

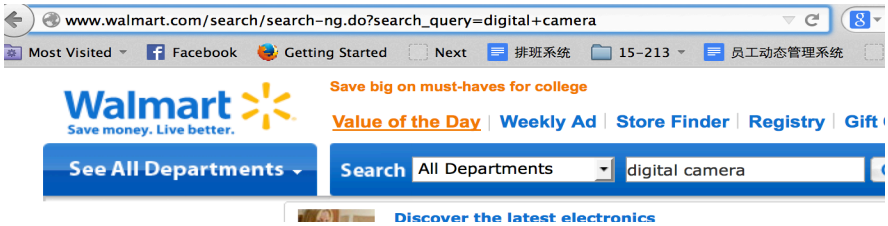
4.1 Setting the base url.

BaseURL: http://www.walmart.com/search/search-ng.do?search_query=

4.2 Reformat the query String.

When typing the query into search box in the web page, we can find that the query string is parsed into the URL in a different format.

For example, changing “digital camera” to “digital+camera”.



So the modified version should be “http://www.walmart.com/search/search-ng.do?search_query=digital+camera.

4.3 Include page query parameter in the URL

We can also easily get the following URL when we click page 2 in search result page.

It would not be difficult to find that “ic=16_16” indicates number per page and page number. In fact ic = Number per page_ (pagenumber – 1) * Number per page for any page number greater than 0. We also need to append this query parameter in the url.

4.4 Get the document object.

Use Jsoup to send HTTP GET Request to the modified URL and get response as jsoup.nodes.Document model so that we can traverse the nodes easily.

4.5 Navigate the product information

```
<div class="prodInfo">
  <div class="prodInfoBox">
    <a class="prodLink GridItemLink" title="GE X550 Power PRO Digital Camera with BONUS Memory
    Card Bundle" onclick="s_objectID="http://www.walmart.com/ip/GE-X550-Power-PRO-Digi…y-
    Card-Bundle/33356785_2";return this.s_oc?this.s_oc(e):true" href="/ip/GE-X550-Power-
    PRO-Digital-Camera-with-BONUS-Memory-Card-Bundle/33356785"></a>
    <a class="prodLink ListItemLink" onclick="s_objectID="http://www.walmart.com/ip/GE-X550-Power-
    PRO-Digi…y-Card-Bundle/33356785_3";return this.s_oc?this.s_oc(e):true" href="/ip/GE-X550-
    Power-PRO-Digital-Camera-with-BONUS-Memory-Card-Bundle/33356785">
      GE X550 Power PRO
      <span class="highlight">
        Digital
      </span>
      <span class="highlight"></span>
        with BONUS Memory Card Bundle
    </a>
    <div class="OnlinePriceAvail"></div>
    <div class="CRRLike clearfix"></div>
```

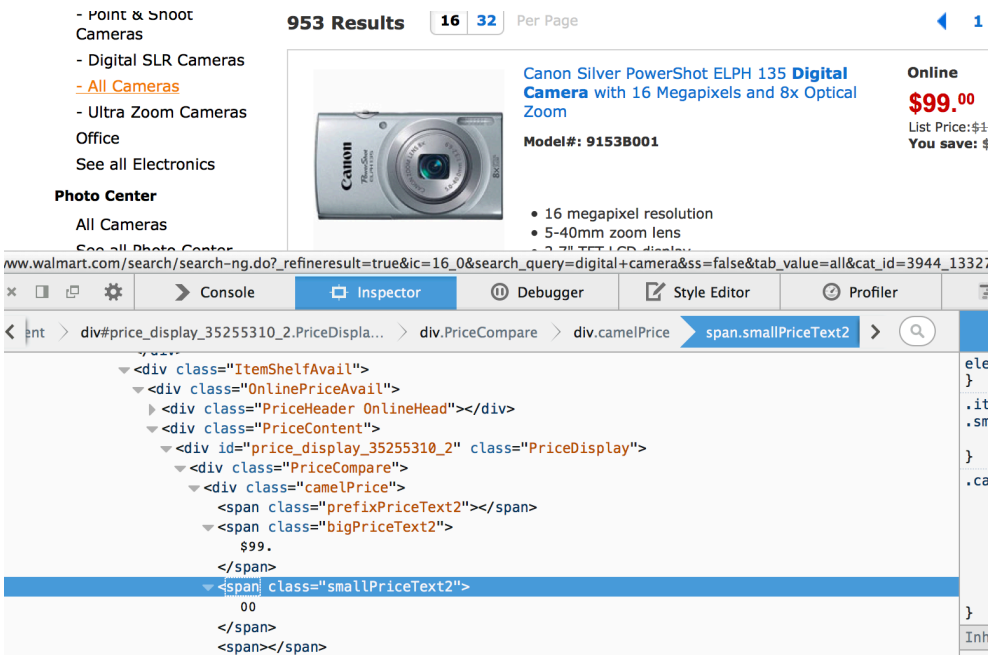
From the HTML source we can find that both product title and price are located within the tag. So we traverse through each prodInfo div and extract the product title, description and the price.

It is easier to get the title since it locates within the <a class = “prodLink ListItemLink>. We can just strip the text within this tag to get the title. There is no tag that can help us simplify the title. I have tried one simple way, which is to extract the text before the last highlight text. It works for query like “digital camera” or “baby stroller” but not for query like “cat”, “dog” since text ending with these queries cannot become a short title. So in the implementation I keep the original title.

For price, it has **THREE** scenarios we need to take into consideration

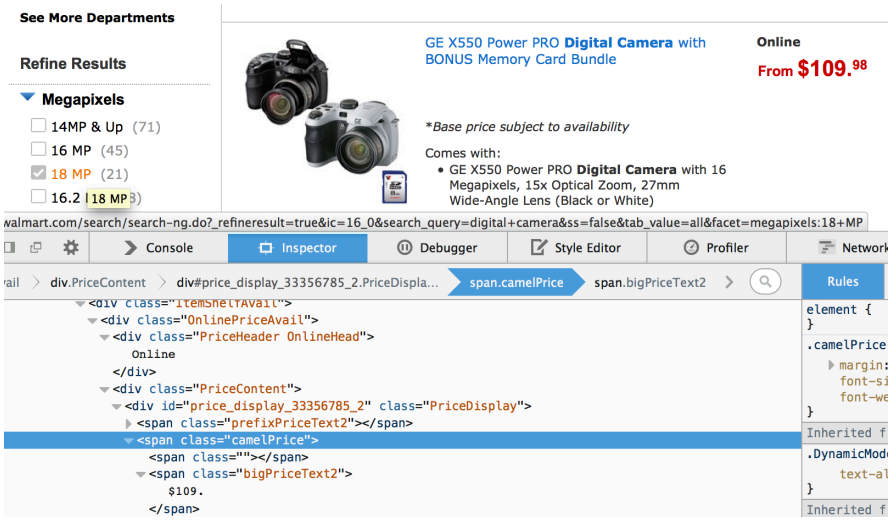
Case1: Regular online price

When online price is an exact price we can just get the text from the div with class name “camelPrice”.



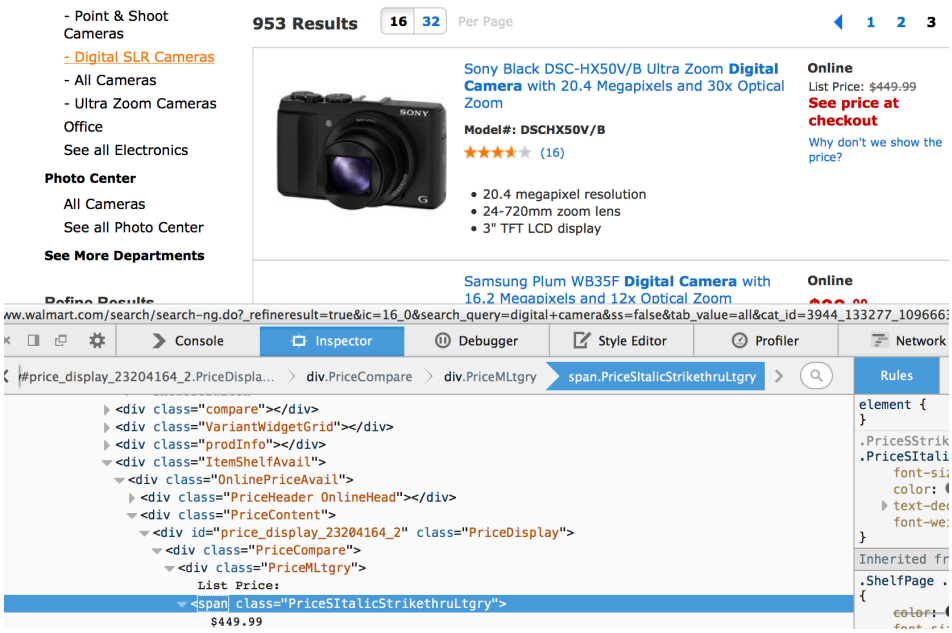
Case 2: Price is a range

When online price is a range like “From $109.98”, there is no div with class name “camel price”, we need to combine the text of span prefixPriceText and span camel price to get the price range.



Case 3: Price at checkout

When online price is not available, we need to get its list price.



In the implementation first we will check if we can find regular price, and then price range, if we cannot find previous two then we extract the list price.

5 Store the result

The last step is to create the product object and store each piece of the product information in the result set. After we collect all the product information we return the result object with product list, query term, query URL, page number, results in the current page and the total number of the query result. The result object will then be passed to the displayer.