Visualization of Information

MSDS 6390

Live Session 6

World Changers Shaped Here

SMU

Contents

- Assignment 5
- Data types
  - CSV Data
  - JSON Data
  - XML Data
- Data processing
- Simple Data Structures:
  - ArrayList
  - HashMap
- More interaction:
  - Hovering elements

Assignment 5

Imaging functions

Data types

CSV data

JSON data

XML data

CSV data

- **C**omma **S**eparated **V**alues
- Tabular data in plain text
- Each line is a record of several fields
- Each field is separated by a separator character:
  - comma
  - semicolon
  - tab space
  - …

```
Region,Activity Type Code,Price Category Code,Terminal,Boarding Area
200507,ATA Airlines,TZ,ATA Airlines,TZ,Domestic,US,Deplaned,Low Fare
200507,ATA Airlines,TZ,ATA Airlines,TZ,Domestic,US,Enplaned,Low Fare
200507,ATA Airlines,TZ,ATA Airlines,TZ,Domestic,US,Thru / Transit,Lo
200507,Air Canada,AC,Air Canada,AC,International,Canada,Deplaned,Oth
200507,Air Canada,AC,Air Canada,AC,International,Canada,Enplaned,Oth
200507,Air China,CA,Air China,CA,International,Asia,Deplaned,Other,I
200507,Air China,CA,Air China,CA,International,Asia,Enplaned,Other,I
200507,Air France,AF,Air France,AF,International,Europe,Deplaned,Oth
200507,Air France,AF,Air France,AF,International,Europe,Enplaned,Oth
200507,Air New Zealand,NZ,Air New Zealand,NZ,International,Australia
200507,Air New Zealand,NZ,Air New Zealand,NZ,International,Australia
200507,AirTran Airways,FL,AirTran Airways,FL,Domestic,US,Deplaned,Lo
200507,AirTran Airways,FL,AirTran Airways,FL,Domestic,US,Enplaned,Lo
200507,Alaska Airlines,AS,Alaska Airlines,AS,Domestic,US,Deplaned,Ot
200507,Alaska Airlines,AS,Alaska Airlines,AS,Domestic,US,Enplaned,Ot
200507,Alaska Airlines,AS,Alaska Airlines,AS,Domestic,US,Thru / Tran
200507,Alaska Airlines,AS,Alaska Airlines,AS,International,Canada,De
200507,Alaska Airlines,AS,Alaska Airlines,AS,International,Canada,En
200507,Alaska Airlines,AS,Alaska Airlines,AS,International,Mexico,De
200507,Alaska Airlines,AS,Alaska Airlines,AS,International,Mexico,En
200507,Alaska Airlines,AS,Alaska Airlines,AS,International,Mexico,Th
200507,All Nippon Airways,NH,All Nippon Airways,NH,International,Asi
200507,All Nippon Airways,NH,All Nippon Airways,NH,International,Asi
200507,American Airlines,AA,American Airlines,AA,Domestic,US,Deplane
200507,American Airlines,AA,American Airlines,AA,Domestic,US,Enplane
200507,American Eagle Airlines,MQ,American Airlines,AA,Domestic,US,D
200507,American Eagle Airlines,MQ,American Airlines,AA,Domestic,US,E
200507,Asiana Airlines,OZ,Asiana Airlines,OZ,International,Asia,Depl
200507,Asiana Airlines,OZ,Asiana Airlines,OZ,International,Asia,Enpl
```

JSON data

- JavaScript Object Notation
- Standard in human-readable format to send data objects
- Data types
  - Number: signed decimal number
  - String: sequence of unicode characters delimited by quote marks “”
  - Boolean: true or false
  - Array: ordered list delimited by [ ]
  - Object: unordered list of attribute:value pairs delimited by {}


```
{
  "stage_3": {
    "Individuel Combativité Etape": [
      {
        "position": "1",
        "number": "179",
        "name": "Thomas VOECKLER",
        "time": "0 pts."
      }
    ],
    "Individuel Jeune Général": [
      {
        "position": "1",
        "number": "182",
        "name": "Julian ALAPHILIPPE",
        "time": "14h 34' 44\""
      },
      {
        "position": "2",
        "number": "111",
        "name": "WARREN BARGUIL",
        "time": "+ 00' 06\""
      },
      {
        "position": "3",
        "number": "051",
        "name": "Wilco KELDERMAN",
        "time": "+ 00' 06\""
      },
```

XML data

- Extensible Markup Language
- Standard in human-readable and machine-readable to send data
- Hierarchical elements delimited by tags:

```
<parent_tag>
    <child_tag>
    </child_tag>
</parent_tag>
```

- Each tag can have properties

```
<course>
    <reg_num>10531</reg_num>
    <subj>THEA</subj>
    <crse>205</crse>
    <sect>F02</sect>
    <title>Design for the Theatre: Sets</title>
    <units>0.5</units>
    <instructor>Muller</instructor>
    <days>M-W-F</days>
    <time>
        <start_time>01:10PM</start_time>
        <end_time>02:00</end_time>
    </time>
    <place>
        <building>THEATR</building>
        <room></room>
    </place>
</course>
```

Data processing

Functions to process data

- Strings
  - **`splitTokens(myString, delim)`**: Split a String depending on one or several delimiters or “tokens”. If no delim is provided, the delimiter is any white space
  - **`int(myString)`**: Change a String to an integer
  - **`float(myString)`**: Change a String to a float
- Interpolation
  - **`map(value, domMin, domMax, ranMin, ranMax)`**: Interpolates linearly a value from a domain to a range

Simple data structures

ArrayList

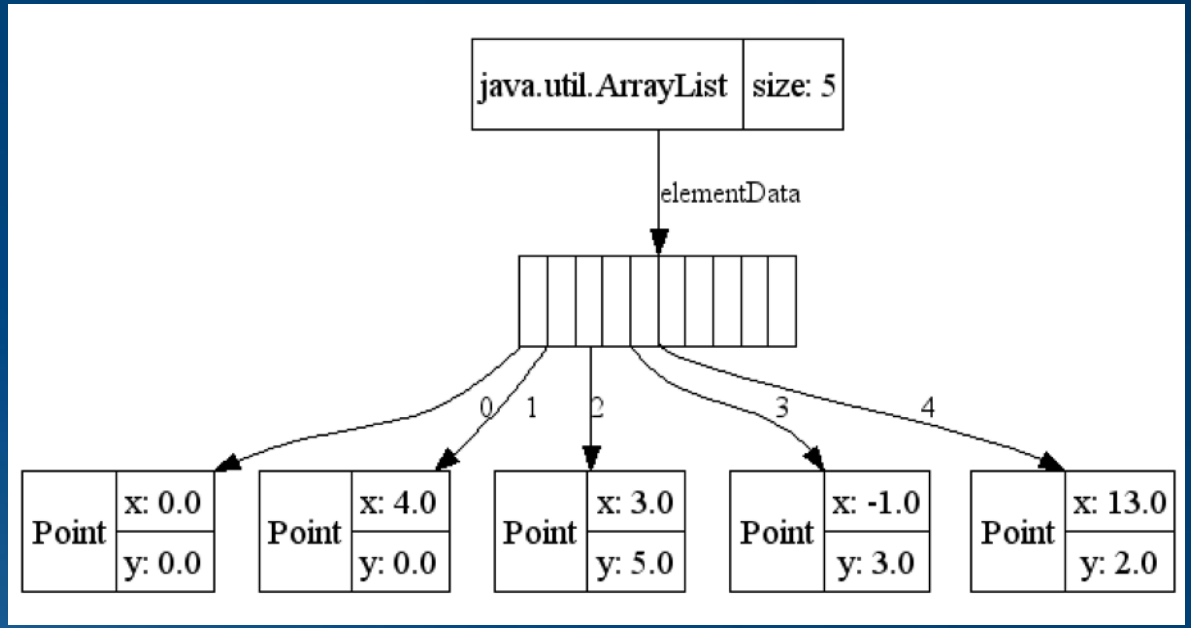
HashMap

World Changers Shaped Here

SMU

ArrayList

- Use a single variable to save a set of variables
- Dynamic size of ordered elements
- The first element is at position 0
- All elements have the same type of data and **cannot** be simple data (int, float, char, boolean)



http://4.bp.blogspot.com/…/ArrayList.png

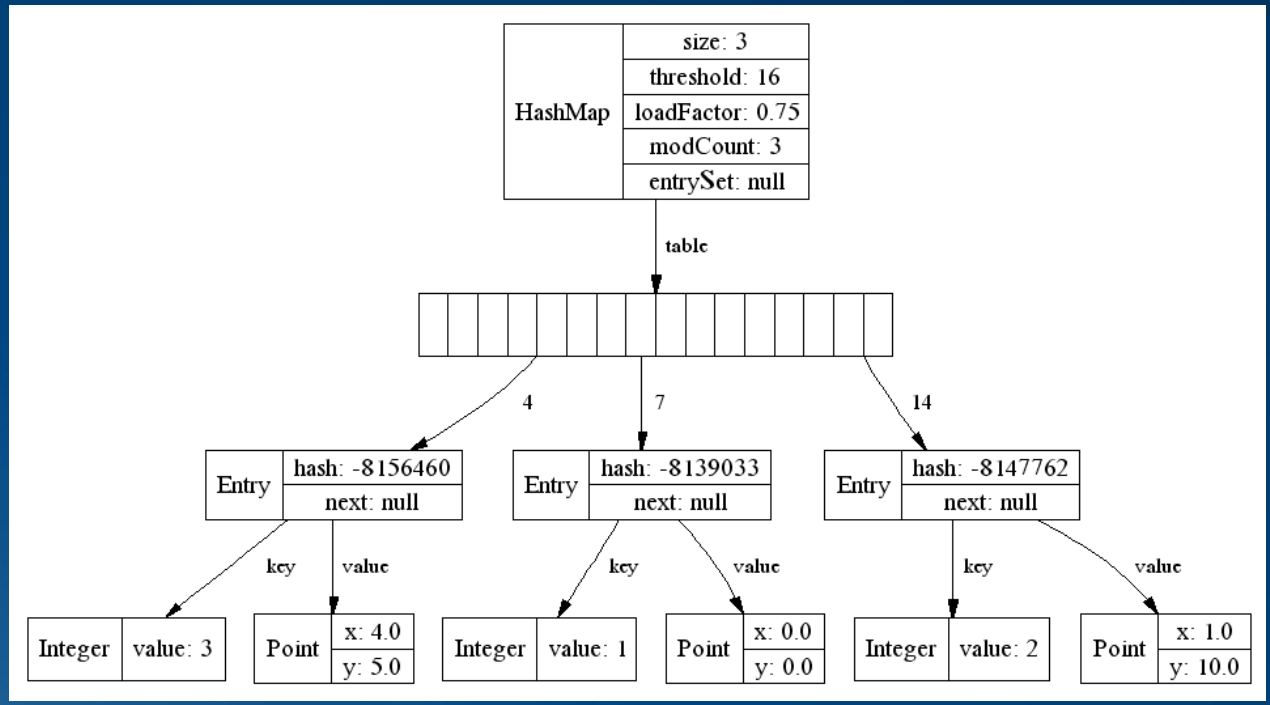
ArrayList

- As any variable, it must be:
  - Declared
  - Initialized
  - Used
    - Get a value from ArrayList
    - Put a value into ArrayList
    - Remove a value from ArrayList

```java
//Declaration
ArrayList<int[]> myPoints;
//Initialization
myPoints = new ArrayList<int[]>();
//Get a value
int index = 4;
myPoints.get(index);
//Insert a value at the end
myPoints.add(new int[]{5, 10});
//Insert a value in a position
//All values shift one position to the right
myPoints.add(index, new int[]{5, 10});
//Remove a value
myPoints.remove(index);
```

HashMap

- Use a single variable to save a set of variables
- Dynamic size of unordered elements
- Uses a key value pair
- Internally, it uses a hashtable to find elements



https://www.cs.auckland.ac.nz/~j-hamer/LJV/HashMapper1.png

HashMap

- As any variable, it must be:
  - Declared
  - Initialized
  - Used
    - Get a value from HashMap
    - Put a value into HashMap
    - Remove a value from HashMap

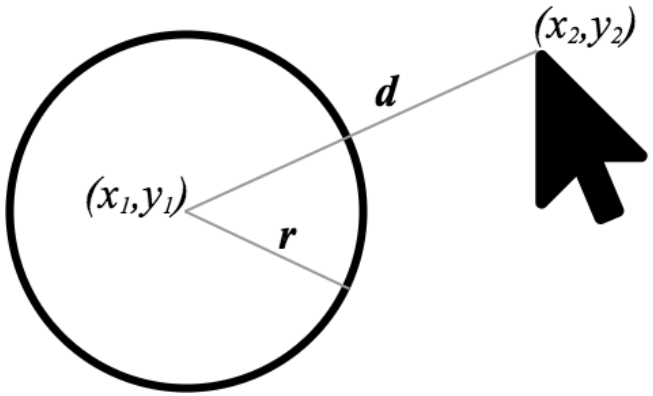
```java
//Declaration
HashMap<String, int[]> myTemperatures;
//Initialization
myTemperatures = new HashMap<String, int[]>();
//Insert a value
myTemperatures.put("New York", new int[]{35, 23, 21, 20});
//Get a value
int[] value = myTemperatures.get("New York");
//Remove a value
myTemperatures.remove("New York");
```

Interaction

Detect hovering

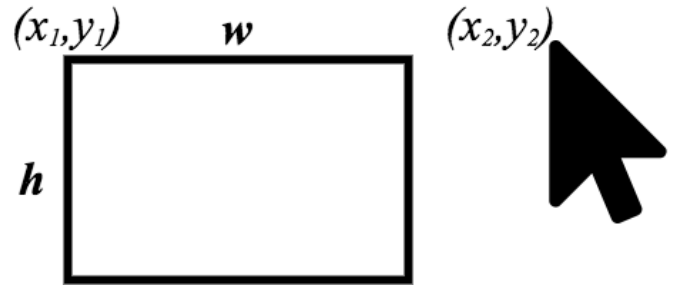
Detect hovering

- Detect if mouse is over an element
  - **Circles**
  - **Rectangles**
  - **Bounding Box:** Enclose the element inside a box
  - **Inside/outside calculation:** Calculate if a point is inside the element



$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

$$d \leq r$$



$$(x_2 \geq x_1) \wedge (x_2 \leq x_1 + w)$$

$$(y_2 \geq y_1) \wedge (y_2 \leq y_1 + h)$$

Live coding

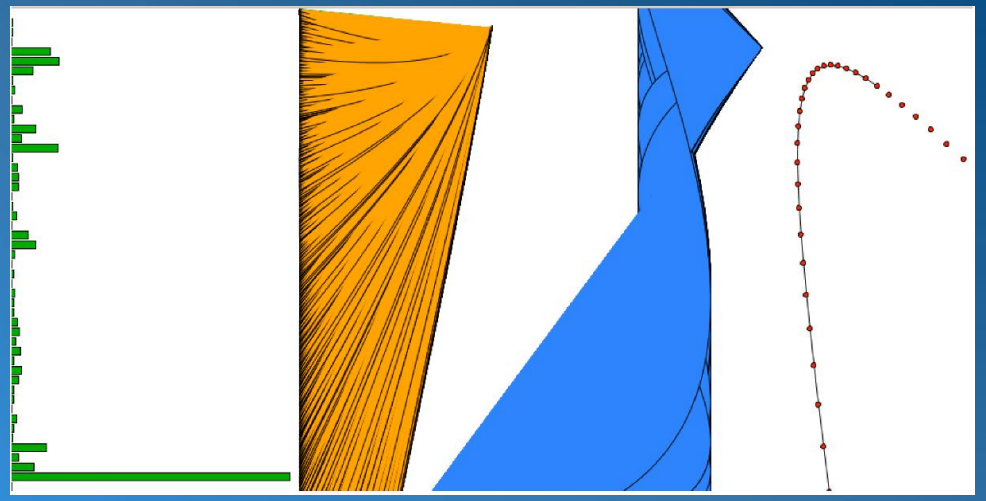
Load data

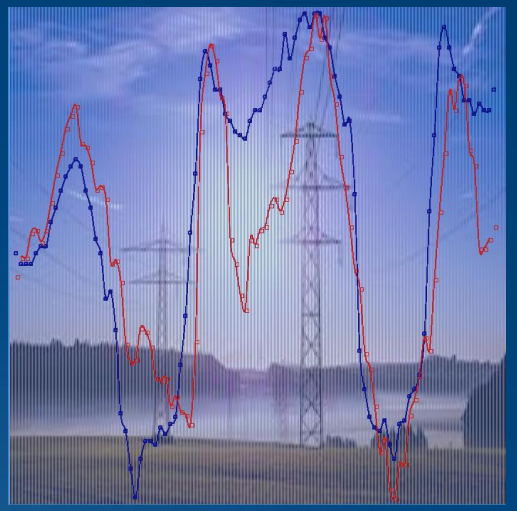
Draw curve

Detect hovering

Assignment 6

- Create a data visualization based on real-time data via the Internet. Utilize Processing's curve functions. Please be prepared to discuss the benefits and challenges of utilizing a curves-based versus linear approach.





More examples…

https://github.com/d3/d3/wiki/Gallery

Questions?

jibarralopez@mail.smu.edu

World Changers Shaped Here

SMU