Visualisasi Data dengan R-Shiny Dashboard

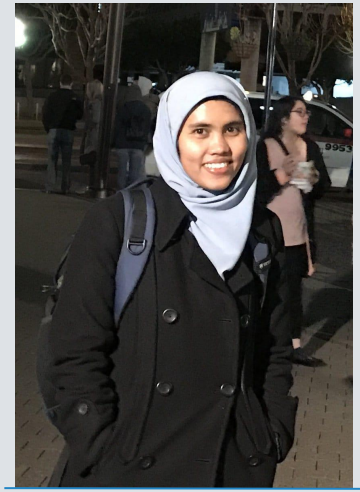
Case Study: Data Harga Pangan Pusat Informasi Harga Pangan Strategis Nasional

By: Erika Siregar

Presented at

Virtual Public Course V, Prodi Statistika, Fakultas Sains dan Teknologi, Universitas PGRI Adi Buana Surabaya - 30 October 2021

Hello, my name is Erika



Erika Siregar

What I am doing now:

- **BPS**
- **R-Ladies Jakarta : Cofounder (IG: @rladiesjkt, youtube: R-Ladies Jakarta, GitHub, Whatsapp Group)**
- **Jakarta Machine Learning: Head of Program**

Education

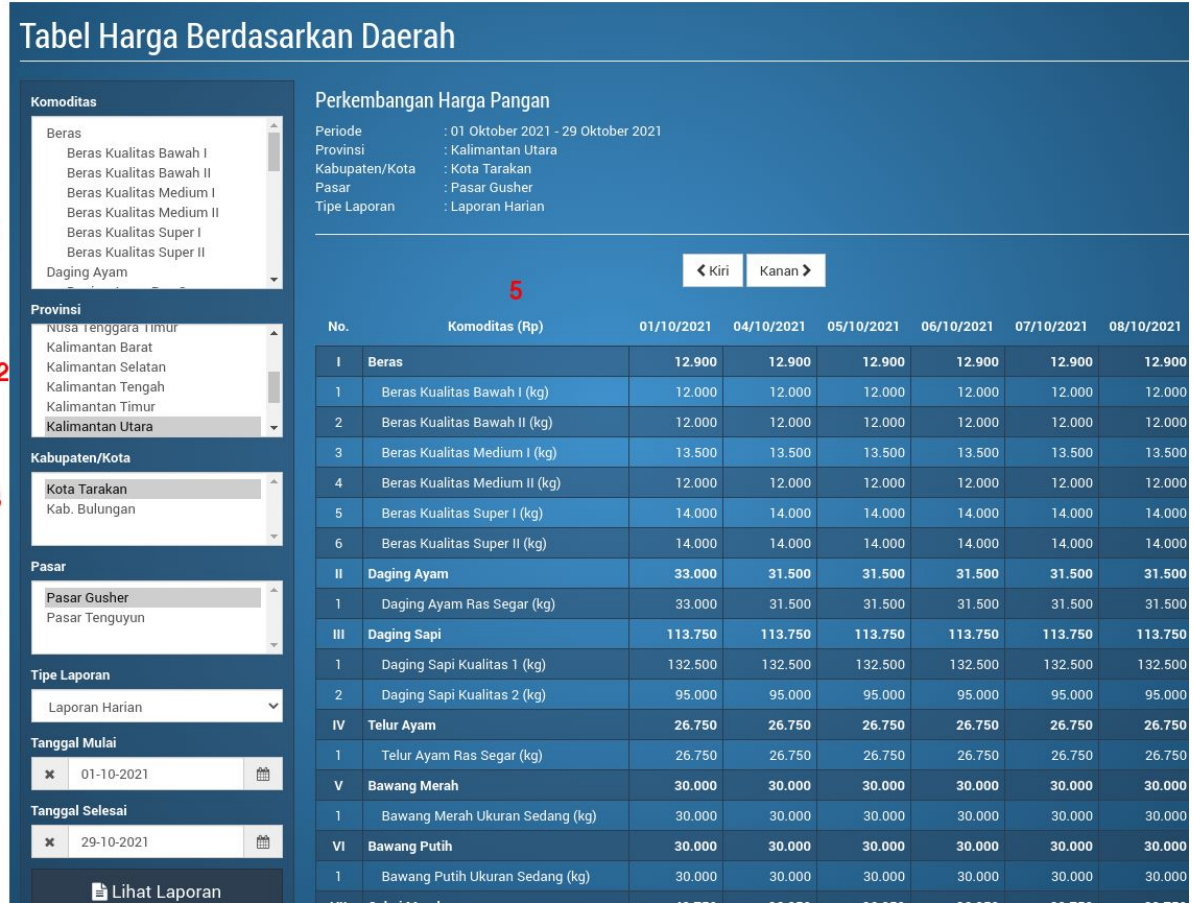
- **Master in Computer Science from Old Dominion University, US**
- **Bachelor of Applied Science from STIS**

Connect with Me:
Email : erika.mukhlisina@gmail.com
GitHub : https://github.com/erikaris
Twitter : @erikaris
Instagram : @erikaris15
Linkedin : https://www.linkedin.com/in/erika-siregar/

Self Check

1. Pernah menggunakan R sebelumnya?
2. Pernah melakukan web scraping sebelumnya?
3. Experience with web programming?
4. Inspect element?
5. Pernah membuat dashboard sebelumnya?

Today's Agenda: (1) Scraping data harga pangan



https://hargapangan.id/tabel-harga/pasar-tradisional/daerah

A glimpse into the data

Sekilas tentang data pangan dari Pusat Informasi Harga Pangan Strategis Nasional:

1. Data resmi dari survei yang diselenggarakan oleh Bank Indonesia
2. Pencacahan data dilakukan setiap hari kerja (Senin s.d. Jumat) pada pukul 09.00 s.d. 11.00 wib.
3. kota yang menjadi sampel pengambilan data adalah 82 kota/kabupaten yang menjadi sampel untuk penghitungan inflasi Indeks Harga Konsumen (IHK) Nasional oleh Badan Pusat Statistik
4. Jumlah sampel pasar yang disurvei adalah 2(dua) pasar tradisional untuk masing-masing kota/kabupaten
5. Jumlah pedagang yang disurvei setiap pasar tradisional adalah 2 pedagang untuk setiap komoditi
6. More detail: https://hargapangan.id/informasi/faq

Agenda 2: Creating the Dashboard

Web Scraping 101

Web Scraping 101

1. Extracting data/information from a website and converting it into a format of your choice (HTML, JSON, CSV, etc.)
2. It's basically copy and paste certain part of the page, but instead of doing it yourself, you ask the computer to do it for you.
3. When do you scrape? → When there is too much to do manually.
4. What can be scraped?
    a. Basically any page
    b. Could be tricky when the page is complex.
    c. Complex == rich of javascript.

3 Types of Web Scraping

1. Simple static page
    a. e.g. Wikipedia, liburnasional.com
    b. rvest (R), scrapy (Python)
2. Dynamic, javascript-heavy page
    a. e.g: E-Commerce, Linkedin
    b. selenium
3. Social Media (Twitter)
    a. official API → rtweet, tweepy
    b. others, e.g: twint (python)

Please Do Keep in Mind

1. The web code and design can change anytime.
2. Be mindful in maintaining the number of requests
3. If there is an API, use it

Rvest

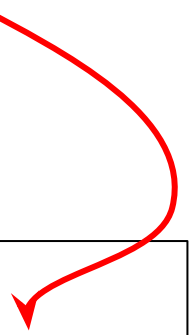
1. check the documentation: https://rvest.tidyverse.org/
2. dependent to other libraries such as: xml2, etc
3. important functions:
    a. **read_html()** --> convert a website into an XML object.
    b. **html_elements()** --> extract the relevant nodes from the XML object
    c. **html_text()** --> extract the tagged data from the wanted nodes.
    d. **html_attrs()** --> return a list of the attributes.

Basic component of web page:

1. Webpage terdiri dari elements
2. Tiap element punya atribut

Steps in Web Scraping

1. Open the webpage
2. Explore it → **inspect particular part we'd like to scrape** → **THE MOST IMPORTANT STEP**
3. scrape the page using scripts → obtain the data
4. preprocessed the data



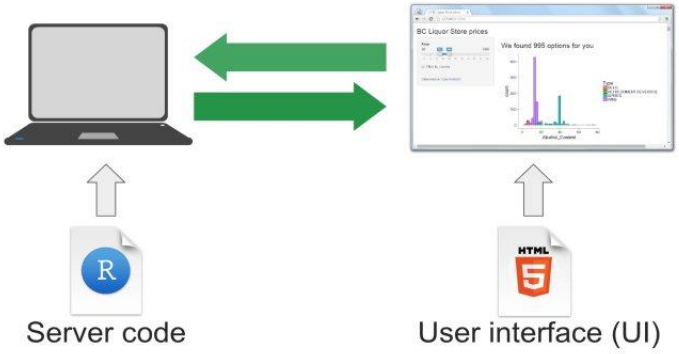
The Key to Web Scraping is Selecting the Right Element (Node)

1. Know about css selector → most commonly used class (.), id (#), and xpath
   a. **Spend time to learn about it!**
2. How:
   a. chrome extension: SelectorGadget
   b. inspect element
3. This is a trial-and-error step → so, bear with it.

Shiny 101

What is Shiny?

1. an **R package** for building an interactive web apps straight from R.
2. Built on bootstrap → responsive page
3. can be hosted as a standalone apps on a webpage ; or embed them in R Markdown documents ; or **build dashboards**.



R code → convert HTML → liatkan inspect element.

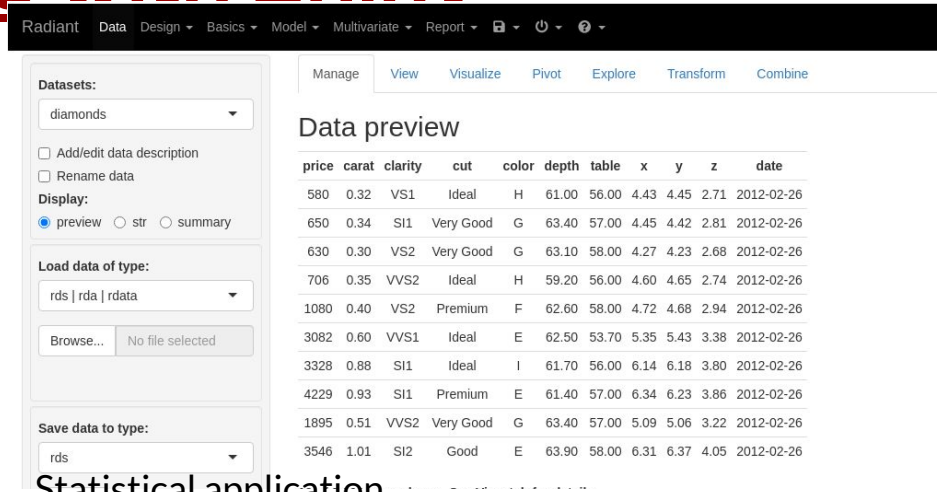
More details: https://shiny.rstudio.com/

Why Shiny?

- Interactivity & animation
- Free
- Open source (no license required) → https://github.com/rstudio/shiny
- Extensible & highly customized
  - Web-based → customize with **CSS**, htmlwidgets, and JavaScript (plotly, d3, etc).
  - Available extension libraries → make your dashboard fancier. → shinyflexdashboard, shinywidgets, etc → https://awesomeopensource.com/project/nanxstats/awesome-shiny-extensions
- R-based → get the computational power of R.

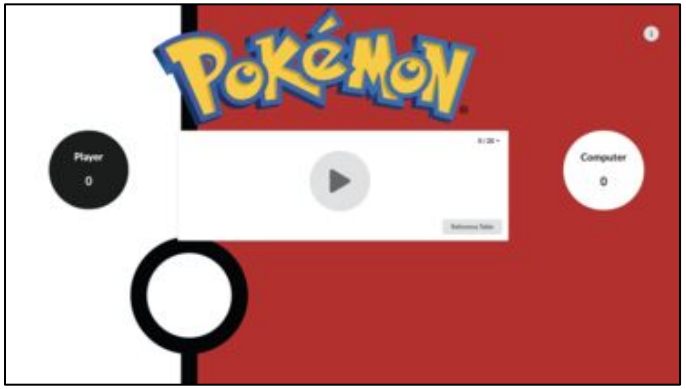
What can We Do with Shiny

dashboard

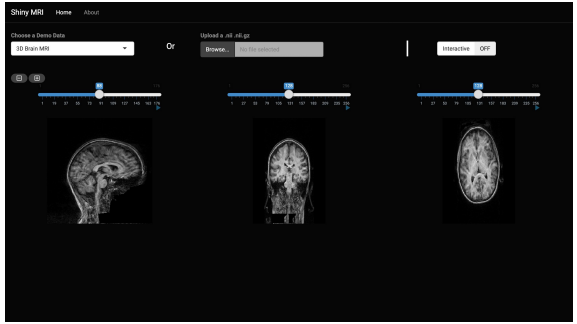


price	carat	clarity	cut	color	depth	table	x	y	z	date
580	0.32	VS1	Ideal	H	61.00	56.00	4.43	4.45	2.71	2012-02-26
650	0.34	SI1	Very Good	G	63.40	57.00	4.45	4.42	2.81	2012-02-26
630	0.30	VS2	Very Good	G	63.10	58.00	4.27	4.23	2.68	2012-02-26
706	0.35	VVS2	Ideal	H	59.20	56.00	4.60	4.65	2.74	2012-02-26
1080	0.40	VS2	Premium	F	62.60	58.00	4.72	4.68	2.94	2012-02-26
3082	0.60	VVS1	Ideal	E	62.50	53.70	5.35	5.43	3.38	2012-02-26
3328	0.88	SI1	Ideal	I	61.70	56.00	6.14	6.18	3.80	2012-02-26
4229	0.93	SI1	Premium	E	61.40	57.00	6.34	6.23	3.86	2012-02-26
1895	0.51	VVS2	Very Good	G	63.40	57.00	5.09	5.06	3.22	2012-02-26
3546	1.01	SI2	Good	E	63.90	58.00	6.31	6.37	4.05	2012-02-26

Statistical application



game



MRI visualization apps

Source:
https://shiny.rstudio.com/gallery/

How to get Started with Shiny?

1. Get the skill
   a. R → must
   b. Web-programming (css & javascript) → nice to have.
2. Install library → install.packages(shiny)
3. Load library → library(shiny)
4. Start building your dashboard.

The Components of Shiny

1. UI → frontend
   a. Input
   b. Output

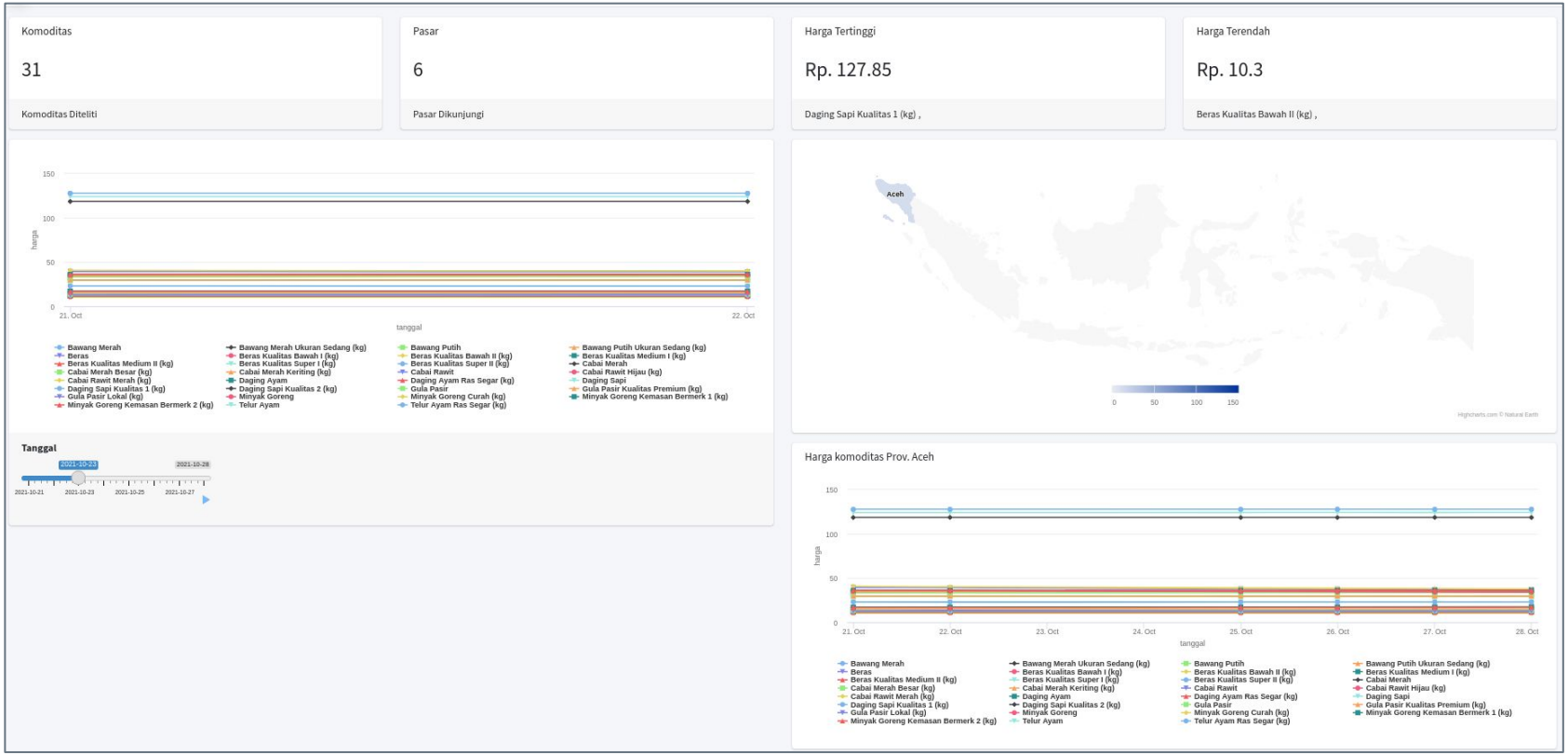
   } layout

2. Server → backend → where the logic of the app is implemented (calculation, aggregation, etc.)

```
# Load the shiny package                    1
library(shiny)

# Define UI for the application
ui <- fluidPage(
  # Add the text "Shiny is fun"              2
  "nyobain shiny untuk pertama kalinya"
)

# Define the server logic
server <- function(input, output) {}        3

# Run the application
shinyApp(ui = ui, server = server)          4
```

UI & server are combined with shinyApp()

Final Dashboard



Hands-On Time!!!

Thank You. Question??