Experiment no 11

Title: R shiny Platform for developers

Aim : Set up r shiny and create interface to connect to CMS open payment data

---

Theory:

What is R studio

RStudio is an integrated development environment (IDE) specifically designed for R, a programming language widely used for statistical computing and data analysis. It provides a user-friendly interface that makes it easier to write and manage R code, visualize data, and perform analyses.

Key Features of RStudio:

1. **Code Editor**: RStudio includes a powerful code editor with syntax highlighting, code completion, and smart indentation, making it easier to write and debug R code.
2. **Console**: You can interactively run R commands in the console, see immediate results, and experiment with code snippets.
3. **File Management**: RStudio helps manage your project files, scripts, and data sets efficiently. You can organize your work in projects, making it easier to keep related files together.
4. **Visualization Tools**: The environment supports visualizations directly in the IDE, allowing you to see plots and graphs in real-time.
5. **Package Management**: RStudio provides tools for installing, updating, and managing R packages, facilitating access to a wide range of statistical and graphical libraries.
6. **Version Control**: It supports integration with Git and other version control systems, making it easier to manage code changes and collaborate with others.
7. **R Markdown**: RStudio supports R Markdown, which allows you to create dynamic reports that combine code, results, and narrative text in a single document. This is particularly useful for reproducible research and sharing findings.

What is R shiny

R Shiny is a web application framework for R that allows you to build interactive web applications easily. It's particularly popular among data scientists and statisticians because it enables them to create user-friendly dashboards and data visualizations without requiring extensive web development skills.

Key Features:

1. **Interactive User Interfaces**: Shiny lets you create responsive UIs using various input controls (like sliders, dropdowns, and text boxes) that users can interact with.
2. **Reactive Programming**: Shiny applications are built on a reactive programming model, meaning the app updates automatically when inputs change. This makes it easy to create dynamic visualizations based on user input.
3. **Easy Integration with R**: Since it's built on R, you can easily leverage R's extensive libraries for statistical analysis and data visualization (like ggplot2, dplyr, etc.).
4. **Server-Client Architecture**: Shiny separates the server logic (where computations and data processing happen) from the UI, allowing for cleaner code and easier maintenance.
5. **Deployment Options**: You can deploy Shiny apps locally, on your own server, or through services like ShinyApps.io, making it accessible to a wide audience.

Steps to create interface using R shiny:

Steps to Create a Shiny App Interface:

1. Install Shiny:

   install.packages("shiny")

2. Load the Shiny Library:

   library(shiny)

3. Define the User Interface (UI):

   - Use `fluidPage()`, `sidebarLayout()`, and input/output functions to create the layout.

4. Define the Server Logic:

   - Create a function that processes inputs and generates outputs.

5. Combine UI and Server:

   - Use `shinyApp()` to connect the UI and server

   shinyApp(ui = ui, server = server

6. Run the App:

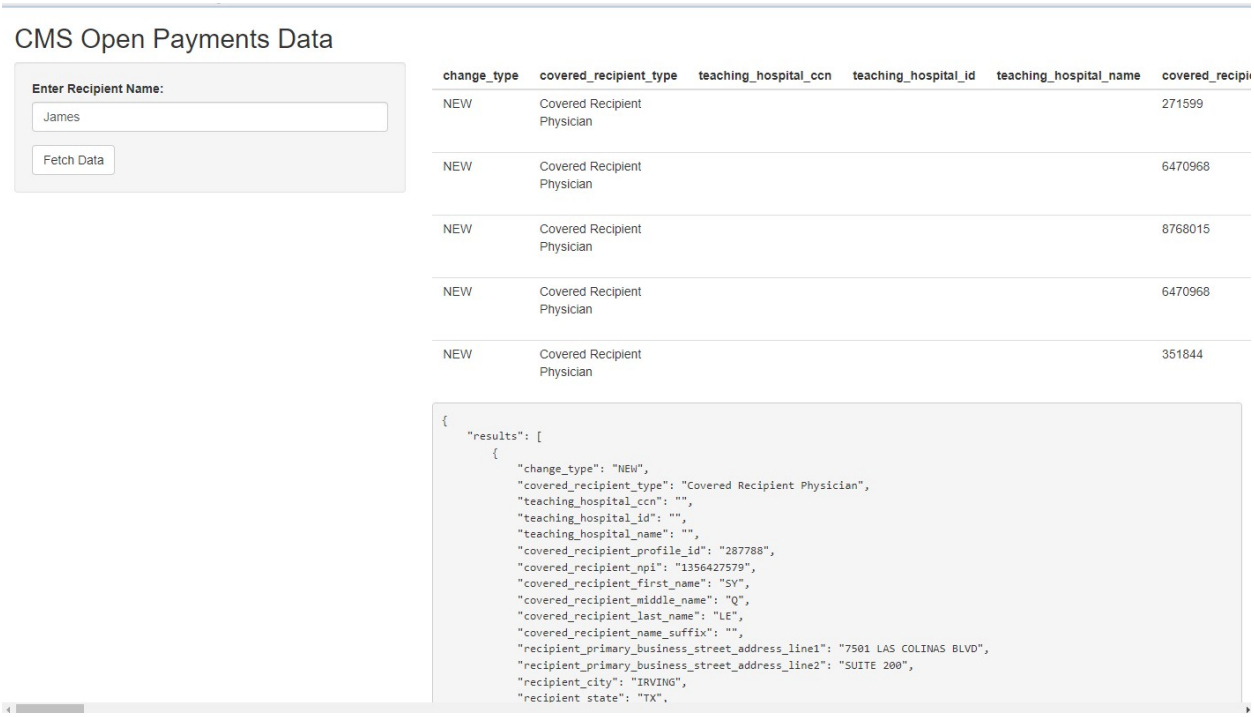
   - Execute the script in RStudio or R console to launch the app.

7. Interact with the App:

   - Adjust the slider and see the output update in real-time.

These steps will get you started with a basic Shiny app interface!

Output:



CMS Open Payments Data

Enter Recipient Name:

Sarah

Fetch Data

change_type	covered_recipient_type	teaching_hospital_ccn	teaching_hospital_id	teaching_hospital_name	covered_recip
NEW	Covered Recipient Physician				1224324
NEW	Covered Recipient Physician				9259
NEW	Covered Recipient Physician				1382689
NEW	Covered Recipient Physician				1382689
NEW	Covered Recipient Physician				814013
NEW	Covered Recipient Physician				5703335
NEW	Covered Recipient Physician				10230420


```
{
    "results": [
        {
            "change_type": "NEW",
            "covered_recipient_type": "Covered Recipient Physician",
            "teaching_hospital_ccn": "",
            "teaching_hospital_id": "",
            "teaching_hospital_name": "",
            "covered_recipient_profile_id": "287788",
```