**Video Script: Section 7 Video 4 – downloading a file.**

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Description | Action on screen | Narration |
| 1 | Introduction  (Outcome and why it is desirable)  This should give the viewer an idea of the outcome of the task at the beginning of the videos and set the stage and expectations of the viewer. | Refer to PPT | In this video, we are going to see how to save data and plots into files from a Shiny app |
| 2 | Context(Problem/Solution)  Present the viewer with a real-world solution and how the situation would pose as a challenge. It always helps to draw the viewer's attention using a use-case. Metadata template can be used here. |  | After having analysed your data in an interactive webpage, it would be useful to be able to save the results in a file. |
| 3 | Guidance (How to do it and how it works): |  | Shiny makes it easy to download a file using downloadHandler(). Let’s see it in action first. |
| 4 |  | Open R and run:  library(“shiny”)  runApp(“activity\_07\_04”)  A description... | Open R and run the shiny app: activity\_07\_04  shiny::runApp(“activity\_07\_04”)  the app shows the distribution of a random sample.  Select the sample size on the left. |
| 5 |  | Click on ‘download as CSV file’ | Clicking on ‘download as a csv file’ creates a CSV file with the raw data and sends it to the browser. |
| 6 |  | Click on ‘download as a png file’ | Clicking on ‘download as a png’ creates a png image with the data and sends it to the browser. |
| 7 |  | Open UI.R and server.R in the editor | Let’s look at the code. |
| 8 |  | Go to UI.R  Highlight downloadButton | The two download buttons are built with downloadButton.  It only requires an id and a label. |
| 9 |  | Go to server.R  Highlight downloadHandler | In server.R, you need to use the reactive expression ‘downloadHandler’ to create and export a file. |
| 10 |  | Highlight filename | downloadHandler has two arguments:  filename: either a string or a function that returns a string. |
| 11 |  | Highlight content | Content: a function with one argument (file) which creates the file you want to export. |
| 12 |  |  | You don’t usually need to specify the file type (text or binary); Shiny infers it from the file extension.  This is all you need to enable the user to download a file. |
| 13 |  |  | Note that we declared the functions randomSample and makeGgplotObject as reactive.  <pause>  Remember that reactive functions save their latest calculated values.  Since *randomSample* and *makeGgplotObject* are used by a couple of functions, making them reactive prevents them to be recalculated at each call. This is more efficient. |
| 14 |  |  |  |
| 15 |  |  |  |
| 16 | Conclusion:The video concludes by showing the viewer that the goal has been achieved, and reminding them why they should be happy about that. A PowerPoint summary slide with the key points emphasized would make it easier for the viewer to remember what was covered in the video | Back to PPT | In this video, we saw how it’s possible for a user to download results from the webpage, either in a text file or a binary image file.  In the next video, we’ll see how to share your shiny app. |