Package 'modgetxl'

October 13, 2022

Title A 'shiny' Module for Reading Excel Sheets

Version 0.4

License MIT + file LICENSE

Author Yadu Balehosur <yadu@vizualytics.com>

Imports shiny, DT, shinydashboard, readxl

Maintainer Yadu Balehosur <yadu@vizualytics.com>

Description This is a shiny module that presents a file picker user interface to get an Excel file name, and reads the Excel sheets using 'readxl' package and returns the resulting sheet(s) as a vector and data in dataframe(s).

RoxygenNote 7.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2020-07-09 04:50:03 UTC

R topics documented:

---

getxl   *getxl*

Description

Convert excel sheets to dataframes, display in DT, and return the dataframes

Usage

```
getxl(input, output, session)
```

Arguments

input	is shiny input variable
output	is shiny output variable
session	is shiny session variable

Details

This is a shiny module that presents a file picker UI to get an Excel file name, and reads the Excel sheets using readxl package and returns the resulting sheet(s) as a vector and data in dataframe(s).

See Also

See xlex for examples

---

getxlUI *getxlUI*

Description

UI to get a excel file name

Usage

```
getxlUI(id)
```

Arguments

id	is the caller's id

---

xldisp *xldisp*

Description

Server function to display excel data as DT

Usage

```
xldisp(input, output, session, xlfile, skiprows)
```

Arguments

input	is shiny input variable
output	is shiny output variable
session	is shiny session variable
xlfile	is the name of the xl file being got
skiprows	is number of rows to skip in the excel file

---

xldispUI	*xldispUI*

Description

UI to display excel sheets read as individual data tables

Usage

```
xldispUI(id, xlfile)
```

Arguments

id	is caller's id
xlfile	is the name of the xl file being got

---

xlex	*getxl Examples*

Description

use shiny module getxl/UI for interactive input of Excel files

Examples

```
library(shiny)
library(modgetxl)
app<- shinyApp(
 ui= uiOutput('xltest'),

 server= function(input, output) {
  sink(file=stderr())
  options(shiny.maxRequestSize=1*1024^2) # 1MB

  output$xltest<- renderUI({
  getxlUI('server')
  })
```

```
  xl<- callModule(getxl, 'server')
  # excel sheets info and data are available once user uploads excel file in UI
  # returned xl has reactiveValues of two variables as below
  observeEvent(xl$sheets,{
  print(xl$sheets)
  print(head(xl$sheetdata[[1]]))
  })
  }

)
## Not run:
runApp(app)

## End(Not run)
```

Index