Storing Files in a Database

Storing and Displaying Files in a Database

Ignition can store different types of files into a database by storing the raw file bytes into a special database column. Ignition can also pull these file bytes out and display certain files within a Client.

> Each database has different column types that are used to store files, so it is important to check with your database documentation to see which data type the column would need to be set to for it to accept file bytes. For example, in MySQL, the datatype that accepts PDF bytes is a **LongBlob** datatype, so you will need to set the PDF Data column to the **LongBlob** datatype. MS SQL accepts the **Varbinary** datatype, so you'll need to set the PDF Data column to a **Varbinary** datatype.

On this page ...

- Storing and Displaying Files in a Database
- Example - PDF File
  - Uploading PDF Files to the Database
  - Displaying PDF Files from the Database

Example - PDF File

One of the most common file types that is stored is PDF files. This allows you to store each PDF file within the central database where each client will have access to it, instead of placing the file in a shared drive that all Client computers have access to.

Storing Files in a Database

Watch the Video

Uploading PDF Files to the Database

With a simple script on a Button component, we can store a PDF file into the database so that any user can view it later. This part does not use a Named Query, because the Named Query Parameters do not have a data type that allows us to pass in the raw file bytes. We can instead use system.db.runPrepUpdate to call a query from the script.

This example requires that you have a table with a byte array column in it. For example: MySQL user the BLOB data type and MSSQL uses the varbinary() data type.

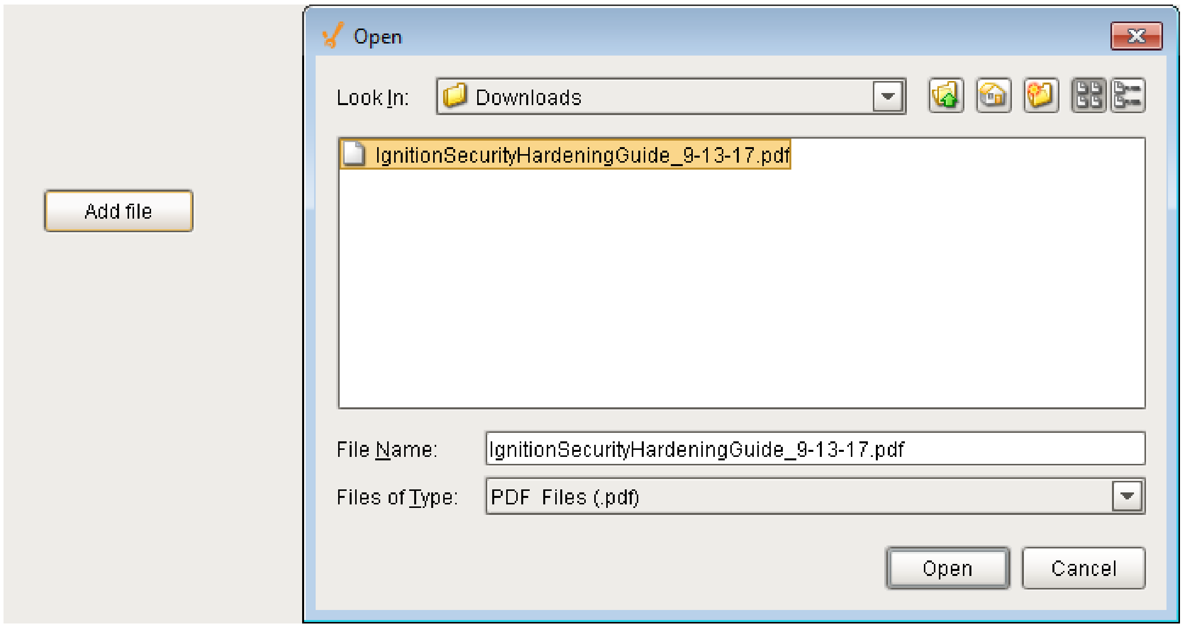
> This function will not work using default Client Permission settings. The Legacy Database Access will need to be enabled for this to work.

1. Add a Button component to a new Main Window.
2. Change the Text property to say Add File.
3. Right click on the Button and select Scripting.
   a. Navigate to the Script Editor Tab of the actionPerformed Event Handler. Here we can put a script that will grab the file bytes using the file path and the system.file.readFileAsBytes function, and then insert that into the database, along with a user selected file name.

Python - Uploads PDF Files to a Database Using a Button

```
# Find the path to the PDF file.
path = system.file.openFile("pdf")

# Check to ensure that the user actually selected a filepath.
if path != None:

 # Read the file as bytes.
 data = system.file.readFileAsBytes(path)

 # Ask the user to enter in a filename.
 # Will grab just the filename and extension from the path as a suggestion.
 name = system.gui.inputBox("Enter a name for the file", path.split('\\')[-1])

 # Check to ensure that the user entered a name for the file.
 if name != None:

  # Insert the data and name into the database.
  system.db.runPrepUpdate("INSERT INTO files (fileName, fileBytes) VALUES (?,?)",
[name,data])
```

4. Test out the script by putting the Designer into Preview Mode and clicking the Button.
   a. First, select the file to load.

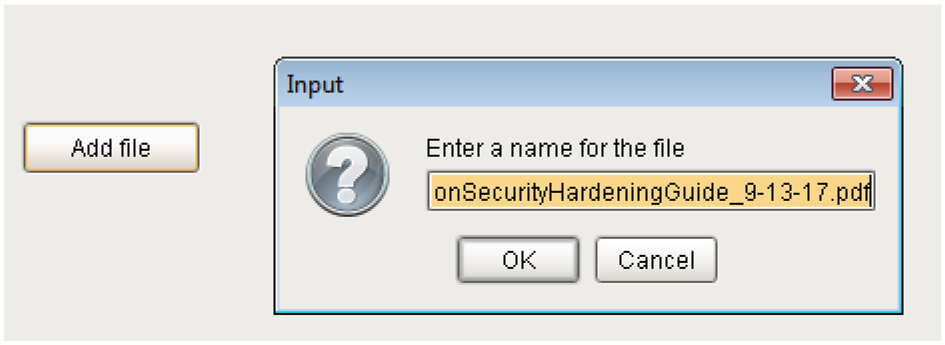


IgnitionSecurityHardeningGuide_9-13-17.pdf

File Name: IgnitionSecurityHardeningGuide_9-13-17.pdf

Files of Type: PDF Files (.pdf)

   b. Then enter in a file name.



Enter a name for the file

onSecurityHardeningGuide_9-13-17.pdf

Displaying PDF Files from the Database

Ignition can render a PDF document inside the PDF Viewer component, which is a part of the Reporting Module. To view PDF files in the Client, your Ignition server must have the Reporting Module installed. Once the module is installed, you can load the bytes from the database into the PDF Viewer component.

1. Create a new Named Query that will be used to select the file names and ids for a Dropdown selection. Set up security to fit your needs, and name it appropriately. For more information on creating Named Queries, see Using Named Queries - Example.
   a. With no Parameters, add a query to select all the files in our files table.

SQL - Selecting all Files

```
SELECT id, fileName FROM files
```

2. Create a second Named Query, same as in step 1. This will be used to grab the file name and bytes after the user has chosen a file.
   a. Add a single Value type Parameter, "fileID" that will be an Int4 data type.
   b. Add the query to select the file name and bytes based on the selected ID.

SQL - Selecting a File based on an ID

```
SELECT fileName, fileBytes FROM files WHERE id = :fileID
```

3. On the Main Window, add a Dropdown component and a PDF Viewer component.

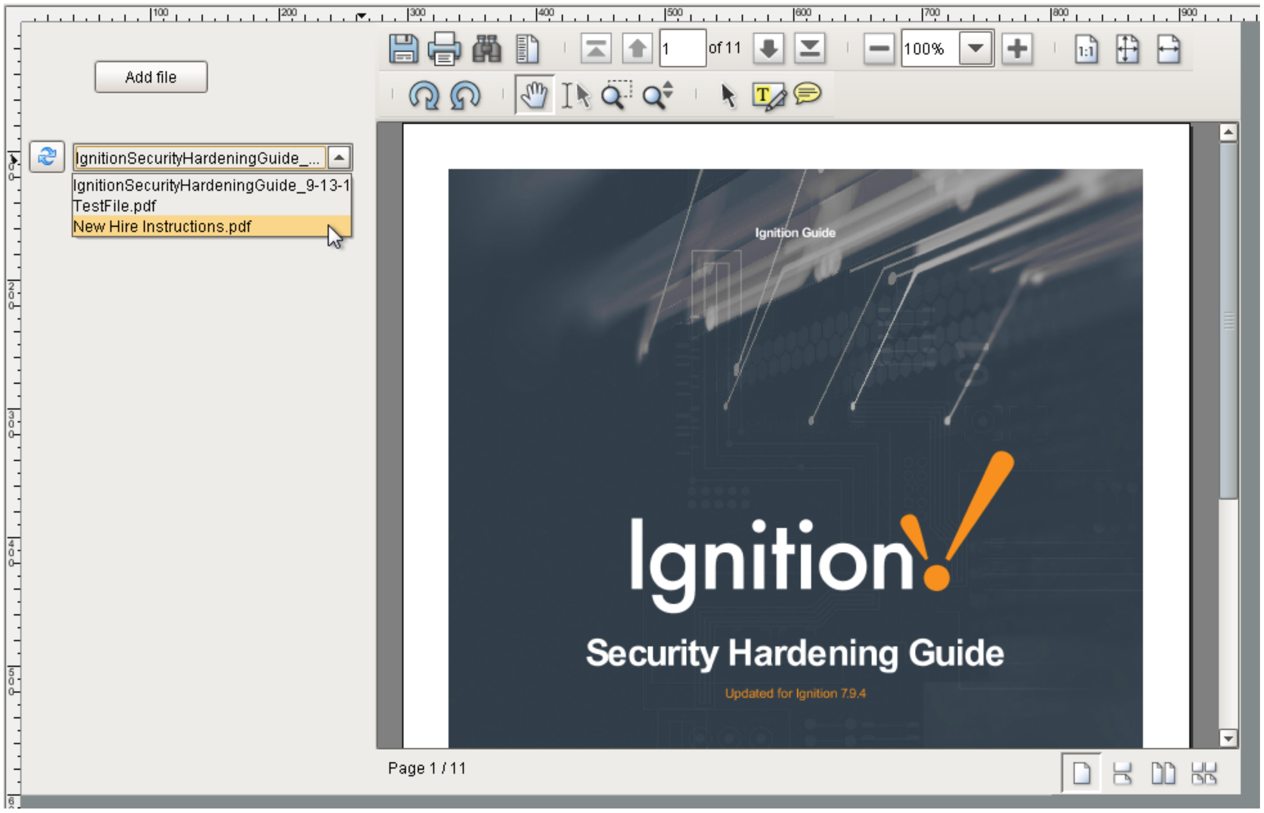
4. On the Dropdown component, add a Named Query binding to the Data property.
   a. Set the binding to the Named Query that was made in step 1.
   b. Place a small refresh Button next to the Dropdown that will refresh this query. See Refreshing a SQL Query for more information for how to refresh on a button.

5. Right click on the Dropdown component and select Scripting.
   a. Navigate to the Script Editor Tab of the actionPerformed Event Handler.
   b. This script will take any new selected value and use it in the Named Query we made in step 2 to get the file name and bytes. We can then load the bytes into the PDF Viewer.

Python - Displays PDF Files from a Database Using the PDF Viewer

```
# Check to see if the property that changed was the Selected Value property.
if event.propertyName == "selectedValue":

 # Run the query to grab the file name and bytes using the new selected ID value.
 system.db.runNamedQuery("Read File", {"fileID":event.newValue})

 # Grab the file bytes and name from the same row.
 bytes = data.getValueAt(0, "fileBytes")
 name = data.getValueAt(0, "fileName")

 # Load the bytes into the PDF Viewer component.
 event.source.parent.getComponent('PDF Viewer').loadPDFBytes(bytes, name)
```

6. Place the Designer into Preview Mode, and try selecting one of the stored files.


Add file
IgnitionSecurityHardeningGuide_...
IgnitionSecurityHardeningGuide_9-13-1
TestFile.pdf
New Hire Instructions.pdf
1
of 11
100%
Ignition Guide
Ignition
Security Hardening Guide
Updated for Ignition 7.9.4
Page 1 / 11

Related Topics ...

- PDF Viewer