

# HCL Notes Designer: struggling with configuration variables

PART 4 /4 : NOTESNAMEDDOCUMENTS

# NamedDocuments vs Profiles

- ▶ Both
  - ▶ Do not appear in views/folders nor FT Search
- ▶ Profiles documents
  - ▶ Stores preferences at system level or user level
  - ▶ Can be edited easily by the user with @functions and LS
  - ▶ Potential cache problem to access real-time modifications
  - ▶ Potential performance problems
  - ▶ Preferred for static values
  - ▶ More CPU usage due to cached values
  - ▶ Less I/O due to cache
  - ▶ Formula AND LS/Java accesses

# NamedDocuments vs Profiles – Cont'd

- ▶ NamedDocuments
  - ▶ Available starting release 12.0.1
  - ▶ Need code for CRUD actions on it
  - ▶ Behaves like a standard Notes Document
  - ▶ Dedicated for frequent updates and access
  - ▶ Preferred for dynamic values
  - ▶ Less CPU usage
  - ▶ More I/O (no cache)
  - ▶ No @formulas functions to get/set

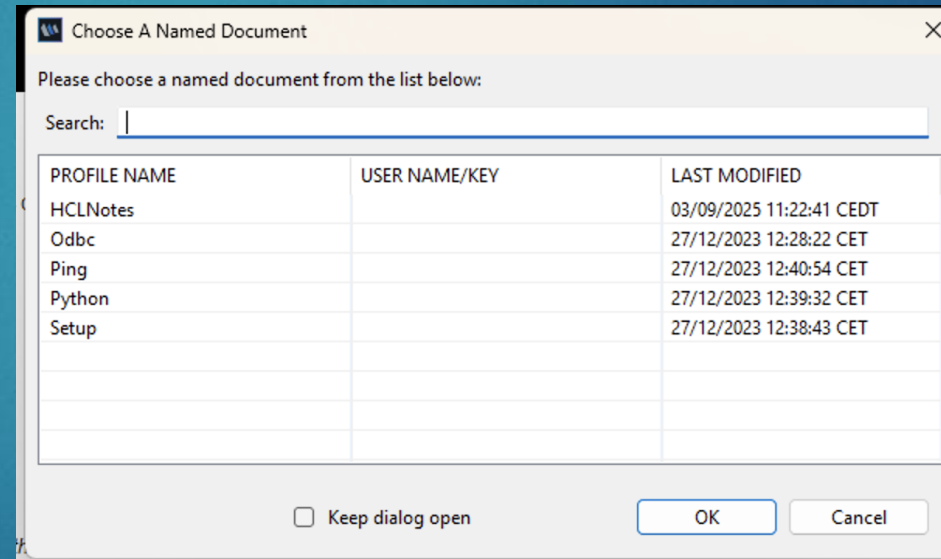
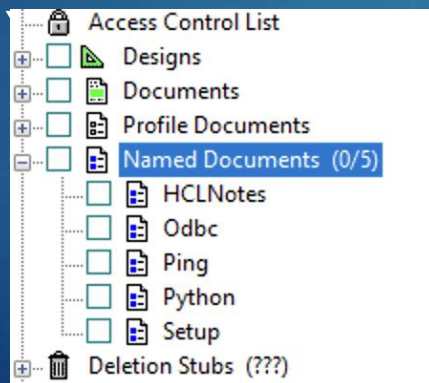


# NamedDocuments - Basics

- ▶ Can be used
  - ▶ You need real-time access to updated values
  - ▶ Values change frequently (user or code)
  - ▶ Sequenced numbers
  - ▶ Centralized Agents configuration
  - ▶ Multiple scripts rely on current data
  - ▶ Etc.

# NamedDocuments – Tools for Access

- ▶ HCL with Panagenda
  - ▶ In Notes client, you can use the advanced properties to access the Named documents
    - ▶ "PROFILE NAME" is a typo



# NamedDocuments - Samples

- ▶ Starting with Release 14, Admin Central is a good example for using NotesNamedDocuments

# NamedDocuments and CRUD

- ▶ Create
  - ▶ Use the getter `NotesDatabase.CreateNamedDocument`
    - ▶ Field `$Name` prefix is "`$DGHST_<Name of NamedDocument>`"
- ▶ Read and Update
  - ▶ Once you get the `nameddocument`, use all functions on `NotesDocument` class
  - ▶ For update you need to call the `NotesDocument.Save` method
- ▶ Delete
  - ▶ Once you get the `nameddocument`, you just use the `NotesDocument.Remove(true)` method



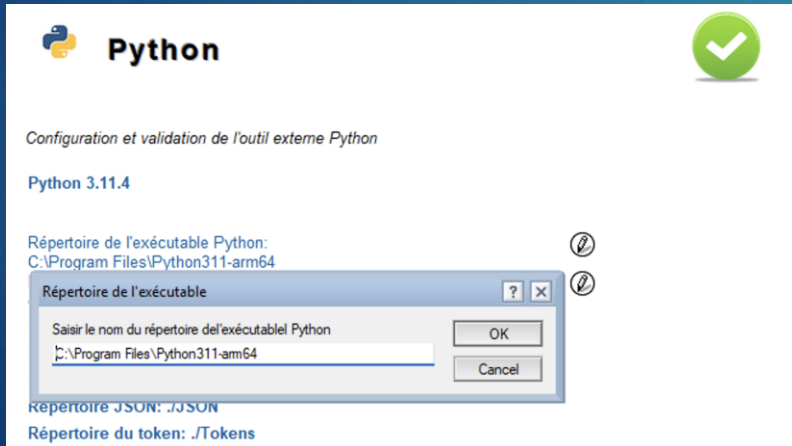
# NotesNamedDocuments

- ▶ Name of a NamedDoc
  - ▶ Can contains \* and period (.)
- ▶ Setter
  - ▶ NotesDatabase.CreateNamedDocument
- ▶ Getters
  - ▶ NotesDatabase.GetNamedDocument(NamedDocName [, uniqueKey])
  - ▶ NotesDatabase.GetNamedDocCollection([NamedDocName])
    - ▶ If NamedDocName is specified, the method retrieves ALL namedDocuments (all NamedDoc with 'NamedDocName' created with or without uniqueKey)



# NotesNamedDocuments – Best Practices

- ▶ Create a class for getters and setters
- ▶ Create a form to get / set values from a form back to the NamedDocument



Répertoire de l'exécutable Python:  
C:\Program Files\Python311-arm64



# NotesNamedDocument - Java

- ▶ Getter and setter in Database
  - ▶ Document `getNamedDocument("NamedDocName" [, UniqueKey])`
    - ▶ If the namedDocument does not exist it will be created using this method call.
- ▶ Getters in Database
  - ▶ Document `getNamedDocument("NamedDocName" [, UniqueKey])`
  - ▶ DocumentCollection `getNamedDocumentCollection(["NamedDocname"])`
    - ▶ If "NamedDocName" is specified you will get all UniqueKey documents in a collection
- ▶ Save
  - ▶ You need to call `NotesDocument.Save` method to save the NamedDocument