

#PRAC4---(07/01/2020) ---[CouchDB Fauxton | R-Tool (R x64 3.6.2)]

#AIM: Practical of Data collection, Data curation and management for Unstructured data (NoSQL).

#REQUIREMENTS:

- Visit <https://couchdb.apache.org/> . **Download** and **Install** it.

#THEORY:

For Database:

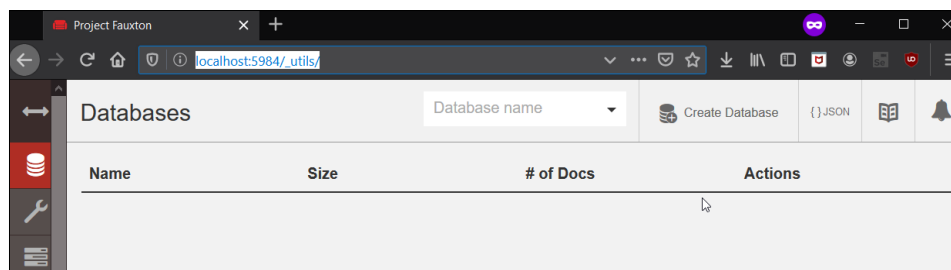
- **CouchDB Fauxton:** a web based built-in administration *interface* for CouchDB.

For Programming Language:

- **R-Tool:** a statistical programming language. We'll use the “**sofa**” package which provides a *programming interface to the NoSQL database CouchDB*.

#SETUP:

- Run *couchdb.cmd* from “bin” folder. Then start Fauxton by typing *http://localhost:5984/_utils/* in browser.



(NOTE: This is the CouchDB database's web-based interface)

- Perform the commands in the R-Tool.

#STEPS:

- 1) Install “sofa” package:
> `install.packages('sofa')`
- 2) Load the “sofa” package in R-Tool's library:
> `library('sofa')`
- 3) Create a connection object (x):
> `x=Cushion$new()`
- 4) Check whether object (x) created or not:
> `x$ping()`
\$couchdb
[1] "Welcome"

\$version
[1] "2.3.1"

```
$git_sha  
[1] "c298091a4"
```

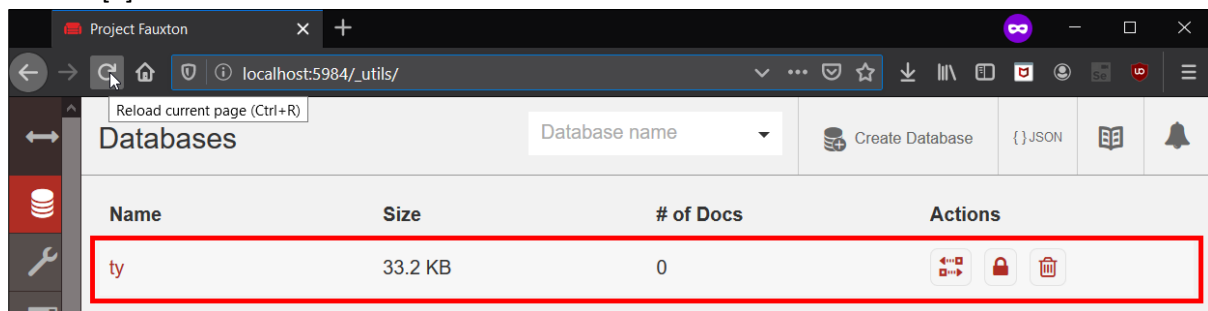
```
$uuid  
[1] "835257892b46a8e3635292c3a25dcf36"
```

```
$features  
$features[[1]]  
[1] "pluggable-storage-engines"
```

```
$features[[2]]  
[1] "scheduler"
```

```
$vendor  
$vendor$name  
[1] "The Apache Software Foundation"
```

- 5) Create database "ty":
> **db_create(x, dbname='ty')**
\$ok
[1] TRUE



--- (Reload the page and the database appears) ---

- 6) List all databases:
> **db_list(x)**
[1] "ty"
- 7) Create three JSON documents (doc1, doc2, doc3) with docid (a_1, a_2, a_3) respectively:
> **doc1 = '{"rollno":"01", "name":"masha", "grade":"a"}'**
> **doc_create(x, doc1, dbname="ty", docid="a_1")**
\$ok
[1] TRUE
- ```
$id
[1] "a_1"
```
- ```
$rev  
[1] "1-664cdfdc0b6473dc29db24517bb12265"
```

```

> doc2 = '{"rollno":"02", "name":"SCOOPY", "grade":"A"}'
> doc_create(x, doc2, dbname="ty", docid="a_2")
$ok
[1] TRUE

$doc
[1] "a_2"

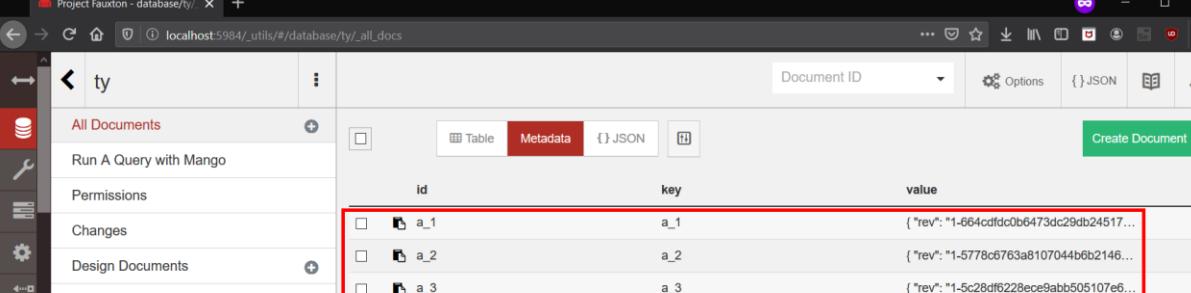
$rev
[1] "1-5778c6763a8107044b6b2146c063651e"

> doc3 = '{"rollno":"03", "name":"spike", "grade":"b", "remark":"pass"}'
> doc_create(x, doc3, dbname="ty", docid="a_3")
$ok
[1] TRUE

$doc
[1] "a_3"

$rev
[1] "1-5c28df6228ece9abb505107e601caeee"

```



	id	key	value
<input type="checkbox"/>	a_1	a_1	{ "rev": "1-664cdfc0b6473dc29db24517..." }
<input type="checkbox"/>	a_2	a_2	{ "rev": "1-5778c6763a8107044b6b2146..." }
<input type="checkbox"/>	a_3	a_3	{ "rev": "1-5c28df6228ece9abb505107e6..." }

--- (docid's reflected in the database) ---

8) List the changes made in the database:

```

> db_changes(x, 'ty')
$results
$results[[1]]
$results[[1]]$seq
[1] "1-
g1AAAAF1eJzLYWBg4MhgTmEQTM4vTc5ISXLlyU9OzMnILy7JAUoxJTikyf__z8rkQGPOiQFIJlk
D1KXwZzImAvksScZpCQaJJth04PPJAeQSfGEbUwAqasnC6PBUgyNAApoNL5xKhDAFG7nxi1By
Bq7xOj9gFELci9WQBNonmE"

$results[[1]]$id
[1] "a_3"

$results[[1]]$changes
$results[[1]]$changes[[1]]
$results[[1]]$changes[[1]]$rev
[1] "1-5c28df6228ece9abb505107e601caeee"

```

```
$results[[2]]
$results[[2]]$seq
[1] "2-
g1AAAAHleJzLYWBg4MhgTmEQTM4vTc5ISXLlyU9OzMnILy7JAUoxJTlkyf____z8rgzmRIRcowG6
SkpxsaWCQwsBZmpeSmpaZl5qCR3uSApBMsoeawAg2lckgJdEg2QybHnwmOYBMikcxydjMyDj
VwJBUXxJAItWDTEpkwKMujwVIMjQAKaDS-
YgQME1KTTVPJTIEIKYsgJiynxgbD0DU3idG7QOIWqT4SUw1N0IOS0J2XRYAFUuYyA"
```

```
$results[[2]]$id
[1] "a_2"
```

```
$results[[2]]$changes
$results[[2]]$changes[[1]]
$results[[2]]$changes[[1]]$rev
[1] "1-5778c6763a8107044b6b2146c063651e"
```

```
$results[[3]]
$results[[3]]$seq
[1] "3-
g1AAAAI7eJyV0EsOgjAQBUBRTNSIJ9Aj1AJFVnIT7TOVVFgoa72J3kRvojfBYk2AxBDYTJNp5svMb
wBgpj0BC54XXAuWmJxTo_PT2divMQW2LMsy1R6Fo21MA8F5jJCAeZEJqQ6ZFB3jbGUr2_6E0
VdgSFDEyb-
ZLimpFF1L8gn2JVOPIfaVdGldhRDBMvZ7XpVNbIWrfSxyq5WQSRnJvTk45e6UR62oAGOFhyIPp7
wa2Wwo8UM6MBunvZ3WylfKKOCKNXdKP5pDsCQ"
```

```
$results[[3]]$id
[1] "a_1"
```

```
$results[[3]]$changes
$results[[3]]$changes[[1]]
$results[[3]]$changes[[1]]$rev
[1] "1-664cdfdc0b6473dc29db24517bb12265"
```

```
$last_seq
[1] "3-
g1AAAAI7eJyV0EsOgjAQBUBRTNSIJ9Aj1AJFVnIT7TOVVFgoa72J3kRvojfBYk2AxBDYTJNp5svMb
wBgpj0BC54XXAuWmJxTo_PT2divMQW2LMsy1R6Fo21MA8F5jJCAeZEJqQ6ZFB3jbGUr2_6E0
VdgSFDEyb-
ZLimpFF1L8gn2JVOPIfaVdGldhRDBMvZ7XpVNbIWrfSxyq5WQSRnJvTk45e6UR62oAGOFhyIPp7
wa2Wwo8UM6MBunvZ3WylfKKOCKNXdKP5pDsCQ"
```

```
$pending  
[1] 0
```

- 9) Query a database for showing all documents (*HINT: id>null*):
> **db_query(x, dbname='ty', selector=list('_id'=list('\$gt'=NULL)))\$docs**

```
[[1]]  
[[1]]$`_id`  
[1] "a_1"  
  
[[1]]$`_rev`  
[1] "1-664cdfdc0b6473dc29db24517bb12265"
```

```
[[1]]$rollno  
[1] "01"
```

```
[[1]]$name  
[1] "masha"
```

```
[[1]]$grade  
[1] "a"
```

```
[[2]]  
[[2]]$`_id`  
[1] "a_2"  
  
[[2]]$`_rev`  
[1] "1-5778c6763a8107044b6b2146c063651e"
```

```
[[2]]$rollno  
[1] "02"
```

```
[[2]]$name  
[1] "SCOOBY"
```

```
[[2]]$grade  
[1] "A"
```

```
[[3]]  
[[3]]$`_id`  
[1] "a_3"  
  
[[3]]$`_rev`  
[1] "1-5c28df6228ece9abb505107e601caeee"
```

```
[[3]]$rollno  
[1] "03"
```

```
[[3]]$name  
[1] "spike"
```

```
[[3]]$grade  
[1] "b"
```

```
[[3]]$remark  
[1] "pass"
```

10) Query a database for showing document(s) whose grade is “a”:

```
> db_query(x, dbname='ty', selector=list(grade='a'))$docs
```

```
[[1]]  
[[1]]$`_id`  
[1] "a_1"
```

```
[[1]]$`_rev`  
[1] "1-664cdfdc0b6473dc29db24517bb12265"
```

```
[[1]]$rollno  
[1] "01"
```

```
[[1]]$name  
[1] "masha"
```

```
[[1]]$grade  
[1] "a"
```

11) Query a database for showing document with remark=pass:

```
> db_query(x, dbname='ty', selector=list(remark='pass'))$docs
```

```
[[1]]  
[[1]]$`_id`  
[1] "a_3"
```

```
[[1]]$`_rev`  
[1] "1-5c28df6228ece9abb505107e601caeee"
```

```
[[1]]$rollno  
[1] "03"
```

```
[[1]]$name  
[1] "spike"
```

```
[[1]]$grade  
[1] "b"
```

```
[[1]]$remark  
[1] "pass"
```

12) Query a database for showing documents with rollno>02:

```
> db_query(x, dbname='ty',
selector=list(rollno=list('$gt'='02')),fields=c('name','grade'))$docs
[[1]]
[[1]]$name
[1] "spike"

[[1]]$grade
[1] "b"
```

13) Query a database for showing all documents in a data frame using “jsonlite” package:

```
> library('jsonlite') #load library "jsonlite"
> res = db_query(x, dbname='ty',
selector=list('_id'=list('$gt'=NULL)),fields=c('name','rollno','grade','remark'),as='json')
> fromJSON(res)$docs #display JSON doc in dataframe
  name rollno grade remark
1 masha    01    a  <NA>
2 SCOOBY   02    A  <NA>
3  spike   03    b  pass
```

14) Delete a document with docid=a_2:

```
> doc_delete(x, dbname='ty', docid='a_2')
$ok
[1] TRUE

$id
[1] "a_2"

$rev
[1] "2-7c83eab80581d84dc900483d61246744"
```

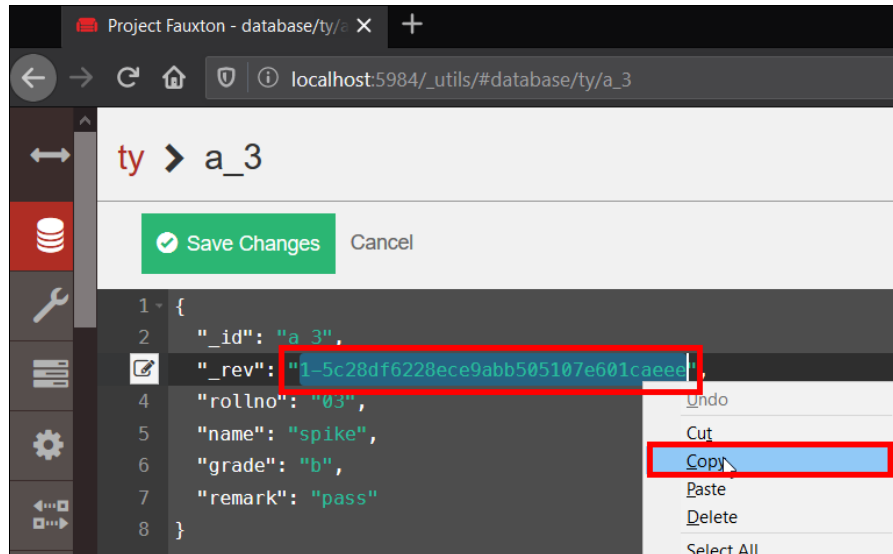


--- (Reload the page and the documents are modified) ---

15) Get that deleted document back to check whether we get an error of deletion:

```
> doc_get(x, dbname='ty', docid='a_2')
Error: (404) - deleted
```

16) Update document whose docid=a_3:



--- (Copying the revision(rev) of docid=a_3 from Fauxton) ---

```
> docNEW='{"diva":"delve", "happy":"hours", "chirpy":"ours"}'
> doc_update(x, dbname='ty', doc=docNEW, docid='a_3', rev='1-5c28df6228ece9abb505107e601caeee')
```

\$ok

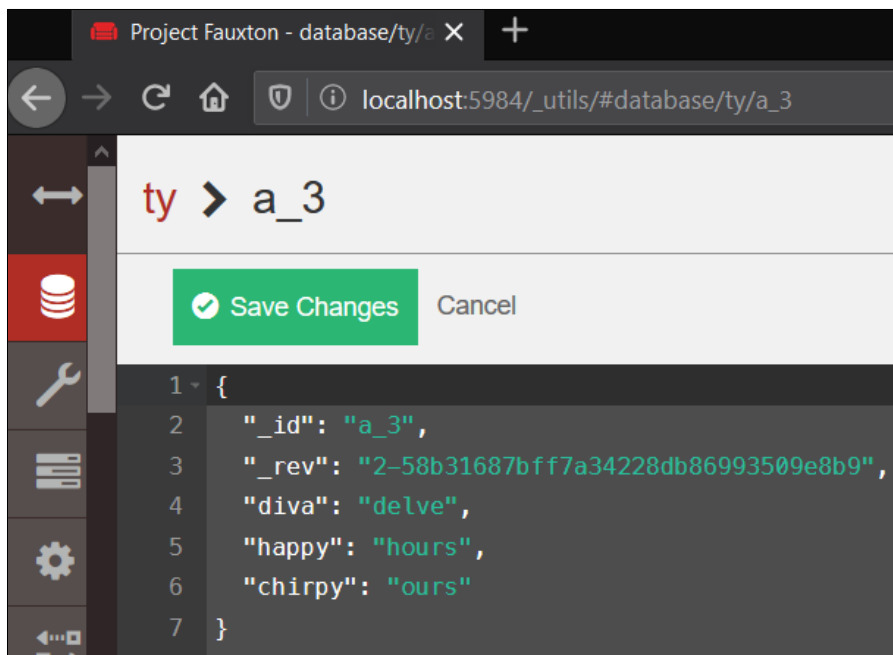
[1] TRUE

\$id

[1] "a_3"

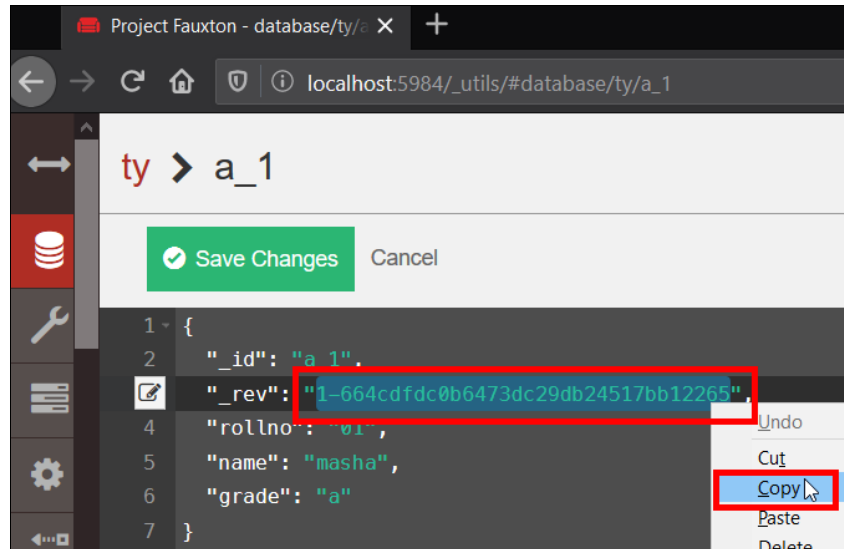
\$rev

[1] "2-58b31687bff7a34228db86993509e8b9"



--- (Refresh & view the Modified document with docid=a_3) ---

17) Update document whose docid=a_1:



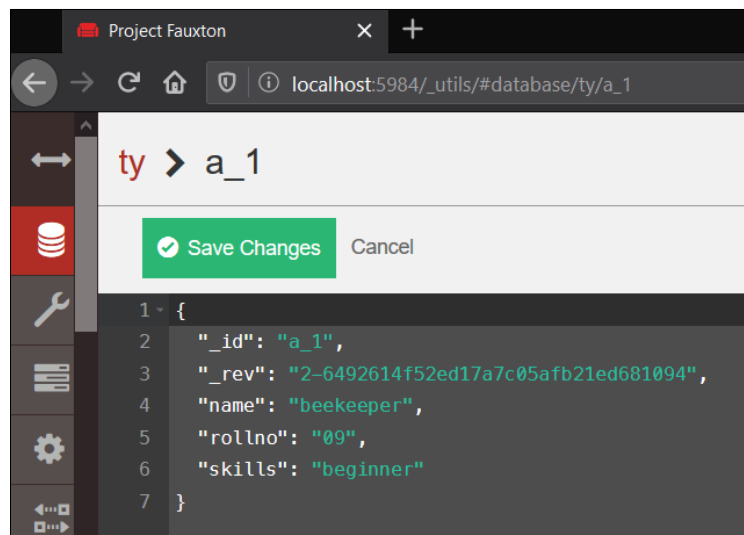
--- (Copying the revision(rev) of docid=a_1 from Fauxton) ---

```
> docNEW={'name':"beekeeper", "rollno":"09", "skills":"beginner"}
> doc_update(x, dbname='ty', doc=docNEW, docid='a_1', rev='1-664cdfdc0b6473dc29db24517bb12265')
```

```
$ok
[1] TRUE
```

```
$id
[1] "a_1"
```

```
$rev
[1] "2-6492614f52ed17a7c05afb21ed681094"
```



--- (Refresh & view the Modified document with docid=a_1) ---

#REFERENCES:

- <https://cran.r-project.org/web/packages/sofa/sofa.pdf>