毕业证书学位证书认证管理

毕业证书

{编号num，证书类别，学制：4，学校，发证日期，姓名name，出生日期，性别，起始年、月，截止年、月，专业名称}

利用VSC编辑一个json文件

{

"num": "1028812001505060846",

"txtype": "毕业证书",

"schoolsystem": 4,

"school": "南京理工大学",

"certdate": "2017-06-16",

"name": "何时宪",

"sex": "男",

"birthday": "1993-01-10",

"begin": "2011-09",

"end": "2015-06",

"major": "计算机科学与技术",

}

这是一个标准JSON串，但不能加入区块链，需要确定一个key

由类别编码与编号组合一个key，例如：

{

"BYZ1028812001505060846": {

"num": "1028812001505060846",

"txtype": "毕业证书",

"schoolsystem": 4,

"school": "南京理工大学",

"certdate": "2017-06-16",

"name": "何时宪",

"sex": "男",

"birthday": "1993-01-10",

"begin": "2011-09",

"end": "2015-06",

"major": "计算机科学与技术"

}

}

这样就建立一个KV。

多个属性之间没有次序。属性名不能用下划线开头。

学位证书

{编号num，证书类别，学位类别，学校，发证日期，姓名name，出生日期，性别，专业}

{

"XWZ1028842017806840":{

"num": "1028842017806840",

"txtype": "学士学位证书",

"degree":"工学",

"school": "南京理工大学",

"certdate": "2017-11-09",

"name": "肖新宇",

"sex": "男",

"birthday": "1992-04-05",

"major": "网络工程"

}

}

一个json文档中可包含多个KV，多个KV之间没有次序要求。

加入区块链

逐个加入

例如，交互方式

invoke put BYZ1028812001505060846 {

"num": "1028812001505060846",

"txtype": "毕业证书",

"schoolsystem": 4,

"school": "南京理工大学",

"certdate": "2017-06-16",

"name": "何时宪",

"sex": "男",

"birthday": "1993-01-10",

"begin": "2011-09",

"end": "2015-06",

"major": "计算机科学与技术"

}

交互方式很难正确执行，原因是，最后一个长串中如果有空格就自动分割为多个参数串，导致错误。

对应一个函数调用

invoke("put", "BYZ1028812001505060846",’{...}’ )

函数调用中书写一个长串，也容易出错，例如，

await fc1.invoke("put", "BYZ1028812001505060846",

'{ "num": "1028812001505060846", \

"txtype": "毕业证书",\

"schoolsystem": 4,\

"school": "南京理工大学",\

"certdate": "2017-06-16",\

"name": "何时宪",\

"sex": "男",\

"birthday": "1993-01-10",\

"begin": "2011-09",\

"end": "2015-06",\

"major": "计算机科学与技术" }');

注意，一个多行长串行尾应该添加续行符\

最简单办法，先创建一个对象，再调用函数，例如

let xwz1 = {

num: "1028842017806840",

txtype: "学士学位证书",

degree:"工学",

school: "南京理工大学",

certdate: "2017-11-09",

name: "肖新宇",

sex: "男",

birthday: "1992-04-05",

major: "网络工程"

}

await fc1.invoke("put","XWZ1028842017806840", JSON.stringify(xwz1));

对象的属性名可以不带双引号。

根据Web页面上输入的各属性值，先构建一个对象，再调用方法写入区块链。

注意，所要求的属性/字段不允许值为null或缺少属性。

批量加入

先创建一个json文本文件，

执行addjson 2.json

用{k1:{v1},k2:{v2},...}形式编写json文件

页面上打开上传一个excel文件，一行表示一个证书，一个文件可包含多个证书，一次性可加入多个证书进入区块链。

对于一行，解析各属性，构造一个value对象，再构造一个key串，形成一个key:value

循环多行，形成一个对象格式json={k1:{v1},k2:{v2},...}

最后调用invoke(“addMKey”, json)

用一次交易写入多个证书。注意，其中Key不能重复。

如何查询/查证？

1由键查询，证书类型与编号

2 由非键属性查询

3 由组合条件查询，如学校，专业，姓名，性别等

1，证书类型；学位证，编号1028842017806840

执行query get XWZ1028842017806840

对应函数调用query("get", "XWZ1028842017806840")

要求输入完整的编号

返回一个V，表示真，或空串，表示假

2，由学校查询

执行query selectBy {"selector":{"school":"南京理工大学"}}

由姓名

query selectBy {"selector":{"name":"肖新宇"}}

由性别

query selectBy {"selector":{"sex":"女"}}

可能返回多个结果，封装为json串。

如果没有结果，就返回“{}”，也是json串。

3，组合条件

组合运算符$and,$or,$not等

例如，查询南京理工大学的网络工程专业颁发的所有证书。

query selectBy {"selector":{"$and":[{"school":"南京理工大学"},{"major":"网络工程"}]}}

前面介绍的每个条件中都隐含“等于$eq”,例如，

{"school":"南京理工大学"}

等价于

{"school":{"$eq":"南京理工大学"}}

所有的”等于$eq”运算符都可简化。但也有其他判断运算，不能简化。

不等于 $ne

大于 $gt

大于等于$gte

小于 $lt

小于等于 $lte

例如，查询2017年6月份颁发的所有证书。

此时需要确定一个日期范围，大于等于6月1日，且小于7月1日

query selectBy {"selector":{"$and":[{"certdate":{"$gte":"2017-06-01"}},{"certdate":{"$lt":"2017-07-01"}}]}}

此时作为查询条件的json串更复杂。命令行执行易出错。

利用VSC工具格式化json有助于发现json错误。

NodeJS编程实现复杂查询，也能简化组合条件的复杂性。例如，

let res2 = await fc1.query("selectBy",'{ \

"selector": {\

"$and": [\

{\

"certdate": {\

"$gte": "2017-06-01"\

}\

},\

{\

"certdate": {\

"$lt": "2017-07-01"\

}\

}\

]\

}\

}')

每一行尾都要添加续行符，也麻烦。可以构建一个对象，然后再执行。例如，

let selector1 = {

selector: {

$and: [

{

certdate: {

$gte: "2017-06-01"

}

},

{

certdate: {

$lt: "2017-07-01"

}

}

]

}

}

let res3 = await fc1.query("selectBy",JSON.stringify(selector1));

如何查询CouchDB？

浏览器上通过如下url可访问couchdb

http://localhost:5984/\_utils/

页面上可以查看所有KV记录，也能执行查询。

也能创建索引，以支持按字段排序。

完整的查询除了selector之外，还可添加fields列出字段，sort对特定字段排序。

例如：

{"selector":{"money":{"$gt":null}},

"fields":["carid","time","money"],

"sort":[{"money":"asc"}]} //sort需要创建索引

如何删除当前值？

删除的原因是什么？不想要了。但不可能完全删除，只是删除当前值，历史不可删改。

只能由key来删除。交互命令为：

invoke delete key

对应调用函数如下：

invoke("delete","BYZ1028812001505060846")

删除之后再用get、selectBy, range等找不到。

只能从历史中查找。例如，

query history BYZ1028812001505060846

返回一个数组，按时间前后次序列出该key的历史。

如何撤销一个证书？

不适合用删除来表示撤销证书，原因是撤销一个证书需要说明正当理由，比如“根据学校XXX文件，因毕业论文抄袭，故撤销毕业证书”等。

删除delete时无法说明理由。

一种办法是添加新属性。

invoke("put","BYZ1028812001505060846",”{...,’status’:’撤销’,’reason’:’论文抄袭，详见文件XXX’}”)

如何修改当前值？

修改的原因是纠正前面输入错误，或者更新状态。

invoke put key newValue

按相同key，重写新值，以前的值作为历史值。

图片加入区块链

图片或音频或小视频文件都可加入区块链。

采用Base64编码为串值，作为base64=串值

方法addBase64(key, infile, {desc})

例如，

addBase64 byz01 毕业证书.jpg {"date":"2017-06","txtype":"本科毕业证书"}

第1个参数是key，

第2个参数是文件名，

第3个参数是一个json串，说明该文件性质，以便于检索。

如果使用以前编码，提示

(node:11647) [DEP0005] DeprecationWarning: Buffer() is deprecated due to security and usability issues. Please use the Buffer.alloc(), Buffer.allocUnsafe(), or Buffer.from() methods instead.

从nodejs v10开始Buffer()弃用。

尽管有弃用提示，文件已加入区块链

通过query get byz01能验证

{byz01=> {base64=...,"date":"2017-06","txtype":"毕业证书"}}

下面加入一个学位证书图片

addBase64 xwz01 学位证书.jpg {"date":"2017-11","txtype":"学士学位证书"}

如何从区块链中提取文件？

方法getBase64(key,outfile)

第1个参数是key

第2个参数是一个输出文件名。若省略，就将解码后的串返回

执行getBase64 byz01 byz01.jpg

将生成一个文件byz01.jpg可用图像查看器打开，也可与原始文件比较。

这两个方法若用于WebServer就可能需要改变。