Radis cache for duplicate transaction

> ℹ️ Intended for dev and Architect.
>
> as per the discussion with Taniya.Biswas@mtn.com for cashback we came across a scenario in which we are checking whole table for duplicate transaction caching is better solution.
>
> this also relevant for duplicate transaction check @Umamaheswara Makineni   @Rupesh Kansara   @Omololu Fabelurin

Problem Statement: | Solution Approach | Step1: | Step2: | Step3: | Step4 | Test Result : | Data-structure setup | Source Code: | Latest Code:

> ✅ Objective to validate the duplicate transaction through payload setting
>
> - Create Redis Server
> - Create Cache Module
> - Create encrypted structure for the with **hashValue**.
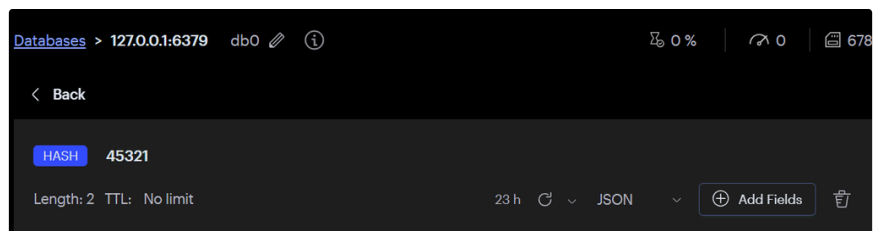
> ℹ️ **Problem Statement:**

we have to create a module in which we have to find out duplicate transaction through through payload and cache that value.

- Spring boot
- Radis Server
- Radis Cache

> ℹ️ **Solution Approach**

Step1:

create a Radis server and then connect with it .

Step2:

connect the spring boot with it.

```
        dKey.java        DemoDataRepo.java        WebConfig.java        appli
   1        spring:
   2          data:
   3            redis:
   4              port: '6379'
   5              timeout: '60000'
   6              database: '0'
   7              host: '127.0.0.1'
   8            application:
   9              name: mtnUrlShortener
  10
  11        ---
```

Step3:

Create a **hash Algo** in which

- **key**-transaction number
- **hashKey**-payload hash string value
- **hashvalue**-hash payload

Step4

Enable caching in Redis Client

```java
@Caching(
        cacheable = {@Cacheable(value = "dataDemo", key = "#transactionId")}
    )
public DataDemo get(String key,String hashKey) {
    return (DataDemo) lettuceRadisTemplate.opsForHash().get(key, hashKey);
}
```

Test Result :

Data-structure setup

please see the Radis data structure.

```
HASH    45321
Length: 2   TTL:  No limit                          1d  ⟳  ∨    JSON      ∨   ⊕ Add Fields  🗑

Field                                    Q    ||    Value

dbc094483ea6d9181d11188113acabc8442a4c         {                                              🗑
8ff7b97e59e134f5a08e8bf34f                       "@class":
                                                 "com.mtn.publicConnector.bean.mtnCommonBean.entity.
                                                 DataDemo",
                                                    "transaction_id": "45321",
                                                    "firstname": "lindan",
                                                    "lastname": "nekol"
                                                 }
```

 Key- 45321

HashKey-dbc094483ea6d9181d11188113acabc8442a4c8ff7b97e59e134f5a08e8bf34f

HashPayload-

```
1  { "@class": "com.mtn.publicConnector.bean.mtnCommonBean.entity.DataDemo", "transaction_id": "45321",
   "firstname": "lindan", "lastname": "nekol" }
```

for same input payload

Request body required

```
{
  "transaction_id": "45321",
  "firstname": "lindan",
  "lastname": "nekol"
}
```

Duplicate is **true**

OK

Media type

application/hal+json

Controls Accept header.

Example Value | Schema

```
true
```

Source Code:

mtnCommon... tor.zip
16 Jul 2024, 10:11 AM

Latest Code:

indranb13/mtnCommonConnector