

**Date :** 4th - Dec- 2020

**Morning Session :** 12pm – 2.00 PM

**By ~** Akash

## Topics: Redis with Mongo

We need to install 3 packages here 1) Express 2) mongo 3) redis

```
1  const express = require('express');
2  const redis = require('redis');
3  const app = express();
4  const port = process.env.PORT || 9000;
5  const mongodb = require('mongodb').MongoClient;
6  var url = "mongodb://localhost:27017";
7
8  const client = redis.createClient({
9    host: 'localhost',
10   port: 6379
11 });
12
13 app.get('/data', (req, res) => {
14   const userid = Number(req.query.uid);
15
16   return client.get(`uid:${userid}`, (err, result) => {
17     if(result){
18       const output = JSON.parse(result)
19       return res.send(output)
20     }else{
21       mongodb.connect(url, function(err, connection){
22         const dbo = connection.db('attainu');
23         dbo.collection('userdata').findOne({_id:userid}, (err, data) => {
24           const output = data;
25
26           // client.setex(key, timetostay, data)
27           client.setex(`uid:${userid}`, 3600, JSON.stringify({source: 'Redis', output}))
28           return res.send({source: 'Mongo', output})
29         })
30       })
31     }
32   })
33 })
34
35
36 app.listen(port, (err) => {
37   console.log(`Server is running on port ${port}`)
38 })
```

Let run localhost:

Please go through recorded Lecture for code implementation.

## What is OAuth (Open Authorization) ?

OAuth (Open Authorization) is an open standard protocol for authorization of an application for using user information, in general, it allows a third party application access to user related info like name, DOB, email or other required data from an application like Facebook, Google etc. without giving the third party app the user password. It is pronounced as oh-auth.

You might have seen a “login with Google” or “login with Facebook” button on the login/signup page of a website that makes it easier to get using the service or website by simply logging into one of the services and grant the client application permission to access your data without giving Password. This is done with the OAuth.

It is designed to work with **HTTP(HyperText Transfer Protocol)** and it allows access tokens to be issued to the third party application by an authorization server with the approval from the owner.

There are 3 Components in OAuth Mechanism–

1. OAuth Provider – This is the OAuth provider Eg. Google, FaceBook etc.

2. **OAuth Client** – This is the website where we are sharing or authenticating the usage of our information.
3. **Owner** – The user whose login authenticates sharing of information.

OAuth can be implemented via google console for “Login/Sign Up with Google” on a web app.

**Pattern to be Followed –**

1. **Get OAuth 2.0 Client ID from Google API Console**
2. **Next, Obtain an access token from the Google Authorization Server to access the API.**
3. **Send the request with the access token to an API .**
4. **Get a Refresh token if longer access is required.**

```
//////////oAuth////////
```

```
> login(clientid)
> return github will provide one code
> pass to github code,clientid,clientsecert
> return git will provide access_token
> pass access_token
> return github will provide userinfo
```

```
application on github(clientid,clientsecert)
```

```
json-server> dummy api
```

```
>
```

```
npm i -g json-server
```

```
json-server --watch db.json --port 8100
```

---