



JsonWebToken

#Node JS Notes

JSON Web Tokens



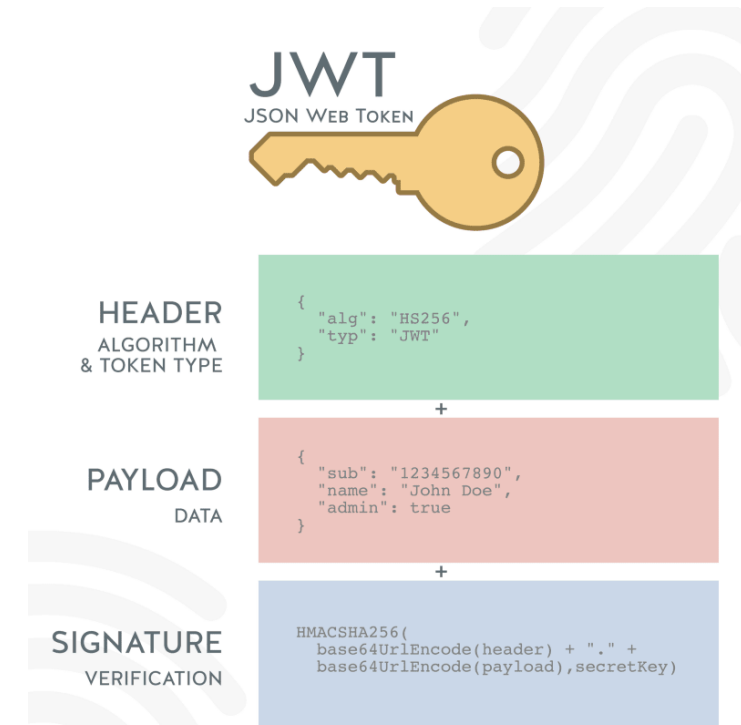
What is JWT ?

- JWT is an open standard for transferring data securely between two parties.
- It is used with authentication systems to make authenticated requests.
- It comprises header, payload, and signature.
- JWT is a stateless authentication mechanism i.e it maintains sessions in the client-side itself instead of storing it in the server.



Structure

- JSON web tokens encode and decode your user's info. They're used for authorization and information exchange.
- They consist of three parts — header, payload, and signature — separated by dots (.) like this: xxxxx.yyyyyy.zzzzzz



100

- JWT STRUCTURE**

JWTs typically feature a Header, a Payload and a Signature

The diagram illustrates the structure of a JWT token. It consists of three parts, each represented by a bracketed segment of a long string of base64-encoded characters. The first segment is labeled 'JWT HEADER', the second 'JWT PAYLOAD', and the third 'JWT SIGNATURE'.

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODFQIiwiaWF0IjoxNjE2MzAxMjM0LmV9.SflKxwRJSMeKKF2QT4fwpMeJf36POk6

JWT HEADER JWT PAYLOAD JWT SIGNATURE



Use

- API secure



Download

- <https://www.npmjs.com/package/jsonwebtoken>
- `npm i jsonwebtoken --save`



100

```
7 //Load JSON
8 var jwt = require('jsonwebtoken');
9 const { JsonWebTokenError } = require('jsonwebtoken');
10
11 //Middleware to Check Auth
12 var AuthJWT = (req,res,next) => {
13   var token = req.headers.authorization;
14   token = token.split(' ')[1];
15   var privateKey = 'sdasdsadsakjslkjsdlfjksdklfjksdklfjksdklfjksdklfjksdkkfjksdkkfjksdkkfjksdkkfjksdkkf';
16   jwt.verify(token,privateKey,function(err,decoded){
17     if(err){
18       console.log(err);
19       res.send({message:'Invalid Token'});
20     }else{
21       next();
22     }
23   })
24 }
25
26 //Display all records API Using JWT
27 router.get('/get-all-users-api', AuthJWT, function(req, res, next) {
28   UsersModel.find({}, function(err, posts) {
29     if (err) {
30       Response.errorResponse(err, res);
31     } else {
32       Response.successResponse('User Listing!', res, posts);
33     }
34   });
35 });
36
37 });
```



100

```
JS users.js X JS app.js
routes > JS users.js > router.post('/login-api') callback > UsersModel.findOne() callback
68
69 router.post('/login-api',function (req, res, next) {
70
71   var email = req.body.user_email;
72   var password = req.body.user_password;
73
74   console.log(req.body);
75   UsersModel.findOne({"user_email": email }, function (err, db_users_array) {
76     if (db_users_array) {
77       var db_email = db_users_array.user_email;
78       var db_password = db_users_array.user_password;
79     }
80     if (db_email == null) {
81       console.log("If");
82       res.send(JSON.stringify({"status": 200,"flag": 1, "message": "Login Fiales No User", "data": ''}));
83     }
84     else if (db_email == email && db_password == password) {
85
86       //Token Key 32 Character
87       var privateKey = "sdasdsadsakjslksdlfjssdklfjssdklfjssdklfjssdklfjssdkkfjsddfsdfkjsdlfjsdf"
88       let params = {
89         email : db_users_array.email,
90         name: db_users_array.user_name
91       }
92       var token = jwt.sign(params, privateKey); // , { expires_in: '500s' }
93       console.log("Token is " + token);
94       res.send(JSON.stringify({"status": 200,"flag": 1, "message": "Login Success", "Token": token}));
95     }
96     else {
97       console.log("Credentials wrong");
98       res.send(JSON.stringify({"status": 200,"flag": 0, "message": "Login Failed", "data": ''}));
99     }
100   });
101 };
102
103
```



Registration API

Working locally in Scratch Pad. [Switch to a Workspace](#)

POST http://127.0.0.1:3000/add-users-api

Save Send

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL

KEY	VALUE	DESCRIPTION	Bulk Edit
<input checked="" type="checkbox"/> user_name	akash		
<input checked="" type="checkbox"/> user_email	akash@gmail.com		
<input checked="" type="checkbox"/> user_mobile	9898123456		
<input checked="" type="checkbox"/> user_password	123		
Key	Value	Description	

Body Cookies (1) Headers (6) Test Results

Status: 200 OK Time: 10 ms Size: 230 B Save Response

Pretty Raw Preview Visualize Text

```
1 {"flag":1,"message":"User Added!","data":{}}
```



Login API

The screenshot displays a REST client interface with a POST request to `http://127.0.0.1:3000/login-api`. The request body is `x-www-form-urlencoded` with the following data:

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> user_email	akash@gmail.com	
<input checked="" type="checkbox"/> user_password	123	
Key	Value	Description

The response is shown in the bottom panel, indicating a `200 OK` status. The response body is:

```
1 {
2   "status": 200,
3   "flag": 1,
4   "message": "Login Success",
5   "Token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1bWllIjoieWthc2giLCJpYXQiOiJlMzZgNDkxNzB9.4LsLGedoNWkiQc-2r25WC75wuHLIbuFEWdYLSumb0Y"
6 }
```



API is not Accessible

The screenshot shows a REST client interface with a GET request to `http://127.0.0.1:3000/get-all-users-api`. The response is a 500 Internal Server Error. The error message in the body is as follows:

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="utf-8">
6   <title>Error</title>
7 </head>
8
9 <body>
10  <pre>Error: Failed to lookup view "error" in views directory "D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\views"<br> &nbsp; &nbsp;at Function.
    render (D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\node_modules\express\lib\application.js:500:17)<br> &nbsp; &nbsp;at ServerResponse.render
    (D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\node_modules\express\lib\response.js:1008:7)<br> &nbsp; &nbsp;at _clear
    (D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\node_modules\req-flash\index.js:18:9)<br> &nbsp; &nbsp;at
    D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\app.js:101:7<br> &nbsp; &nbsp;at Layer.handle_error
    (D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\node_modules\express\lib\router\layer.js:71:5)<br> &nbsp; &nbsp;at trim_prefix
    (D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\node_modules\express\lib\router\index.js:315:13)<br> &nbsp; &nbsp;at
    ...</pre>
```



Authorization

The screenshot shows a REST client interface with a GET request to `http://127.0.0.1:3000/get-all-users-api`. The request has an Authorization header with a Bearer token. The response is a 500 Internal Server Error with a status of 500, time of 21 ms, and size of 2.91 KB. The response body is an HTML document with the following structure:

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="utf-8">
6   <title>Error</title>
7 </head>
8
9 <body>
10  <pre>Error: Failed to lookup view "error" in views directory "D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\views"
    at Function.render (D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\node_modules\express\lib\application.js:690:17)
    at ServerResponse.render (D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\node_modules\express\lib\response.js:1008:7)
    at _clear (D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\node_modules\express-flash\index.js:18:9)
    at D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\app.js:101:7
    at Layer.handle_error (D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\node_modules\express\lib\router\layer.js:71:5)
    at trim_prefix (D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\node_modules\express\lib\router\index.js:315:13)
    at </pre>
```



Authorization : Bearer {token}

Working locally in Scratch Pad. Switch to a Workspace

POST http://127.0.0.1:3000/ POST http://127.0.0.1:3000/ GET http://127.0.0.1:3000/ + ... No Environment

http://127.0.0.1:3000/get-all-users-api Save </>

GET http://127.0.0.1:3000/get-all-users-api Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Headers 7 hidden

KEY	VALUE	DESCRIPTION	...	Bulk Edit	Presets
<input checked="" type="checkbox"/> Authorization	Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1bWVhcnR5dWthc2giLCJ				
Key	Value	Description			

Body Cookies (1) Headers (9) Test Results Status: 500 Internal Server Error Time: 21 ms Size: 2.91 KB Save Response

Pretty Raw Preview Visualize HTML </>

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="utf-8">
6   <title>Error</title>
7 </head>
8
9 <body>
10  <pre>Error: Failed to lookup view "error" in views directory "D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\views";
    render (D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\node_modules\express\lib\application.js:580:17)
    at ServerResponse.render (D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\node_modules\express\lib\response.js:1008:7)
    at _clear (D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\node_modules\req-flash\index.js:18:9)
    at D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\app.js:101:7
    at Layer.handle_error (D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\node_modules\express\lib\router\layer.js:71:5)
    at trim_prefix (D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1\node_modules\express\lib\router\index.js:215:12)
    at ...>
```



Authorization : Bearer Token

http://127.0.0.1:3000/get-all-users-api

GET http://127.0.0.1:3000/get-all-users-api

Params Authorization Headers (9) Body Pre-request Script Tests Settings

Type Bearer Token

The authorization header will be automatically generated when you send the request. [Learn more about authorization](#)

Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend using variables. [Learn more about variables](#)

Token

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...



JWT Website

JSON Web Tokens - jwt.io

JWT

Debugger Libraries Introduction Ask

Crafted by auth0

Algorithm HS256

Encoded PASTE A TOKEN HERE

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MzkwMjQyLCJpc3pwIjoiYV9adQssw5c
```

Decoded EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

```
{  "alg": "HS256",  "typ": "JWT"}
```

PAYLOAD: DATA

```
{  "sub": "1234567890",  "name": "John Doe",  "iat": 1516239022}
```

VERIFY SIGNATURE

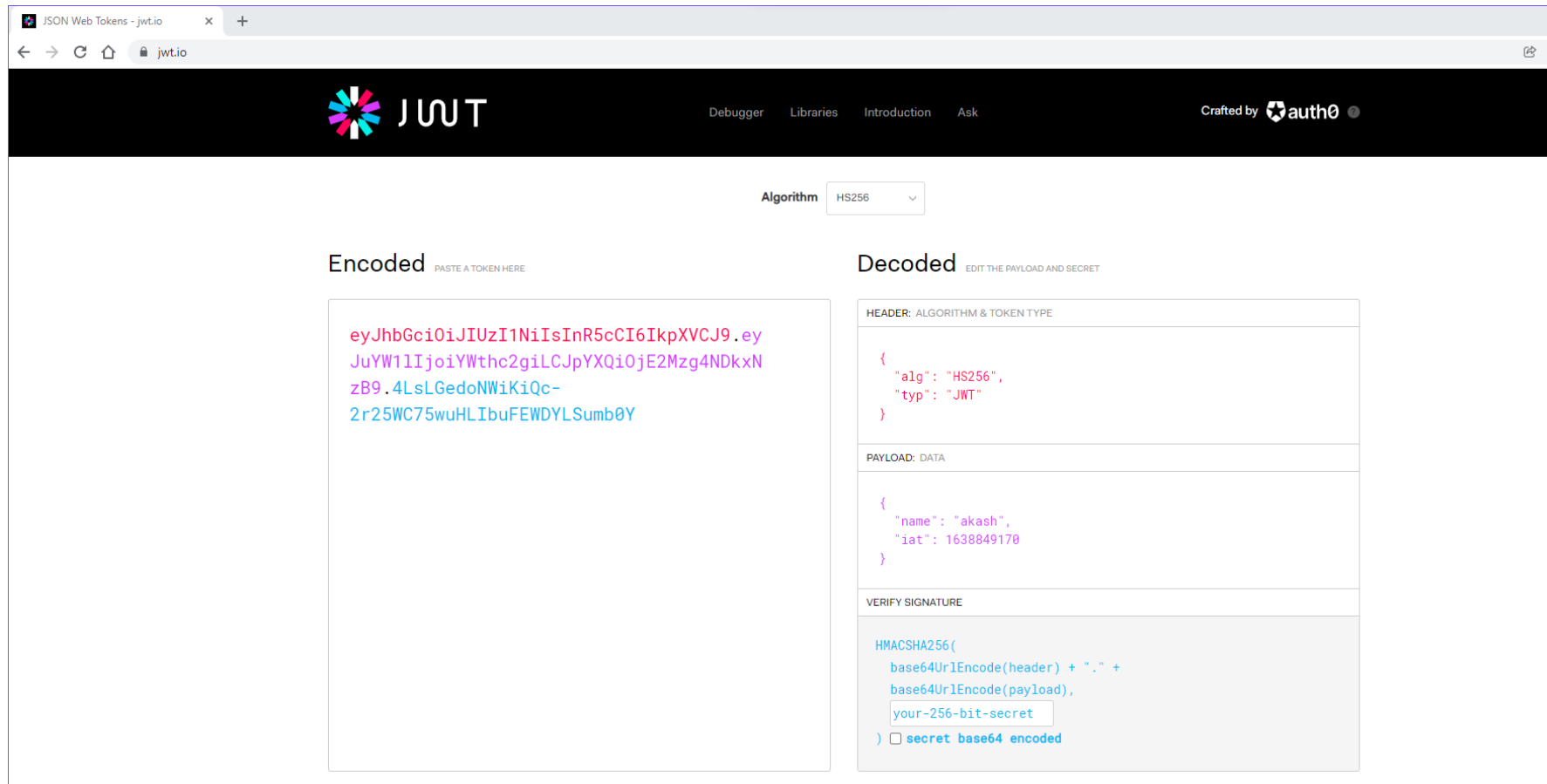
```
HMACSHA256(  base64UrlEncode(header) + "." +  base64UrlEncode(payload),  your-256-bit-secret  ) ☐ secret base64 encoded
```

Signature Verified

SHARE JWT



Decode Token



The screenshot shows the JWT.io website interface. At the top, there's a navigation bar with the JWT logo, links for Debugger, Libraries, Introduction, and Ask, and a note 'Crafted by auth0'. Below the navigation bar, there's a section for decoding a token. The 'Algorithm' is set to 'HS256'. The 'Encoded' section contains a text area with the token: `eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1IjoieWthc2giLCJpYXQiOiJlMzg4NDkxNzB9.4LsLGedoNWiKiQc-2r25WC75wuHLIbuFEWDYLSumb0Y`. The 'Decoded' section shows the decoded token structure:

```
HEADER: ALGORITHM & TOKEN TYPE
{
  "alg": "HS256",
  "typ": "JWT"
}

PAYLOAD: DATA
{
  "name": "akash",
  "iat": 1638849170
}

VERIFY SIGNATURE
HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  your-256-bit-secret
) ☐ secret base64 encoded
```

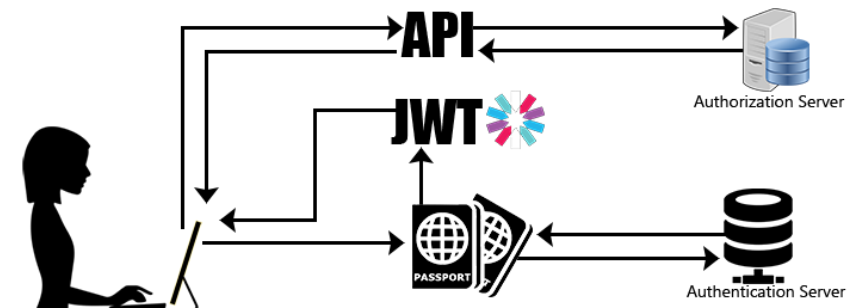


JWT Passport



JWT Passport

- Passport is a Node.js middleware that offers a variety of different request authentication strategies that are easy to implement. By default, it stores the user object in session.
- JSON Web Tokens is an authentication standard that works by assigning and passing around an encrypted token in requests that helps to identify the logged in user, instead of storing the user in a session on the server and creating a cookie. It has different integrations including a Node.js module.



Install Package

- <http://www.passportjs.org/packages/passport-jwt/>
- `npm i passport`
- `npm i passport-jwt`

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```
D:\Code\express-crud-demo-api-master-V1\express-crud-demo-api-master-V1>npm i passport passport-jwt
```



Load Package and Middleware app.js

- `var passport = require('passport')`
- `app.use(passport.initialize());`



Create Middleware to Auth JWT

```
var passport = require('passport');

var JwtStrategy = require('passport-jwt').Strategy;
var ExtractJwt = require('passport-jwt').ExtractJwt;
var opts = {}
opts.jwtFromRequest = ExtractJwt.fromAuthHeaderAsBearerToken();
opts.secretOrKey = 'sdasdsadsakjslkjsdlfjsdklfjsdklfjsdklfjsdkjfsddfsdfkjsdlfjsdf';
passport.use(new JwtStrategy(opts, function(jwt_payload, done) {
  console.log(jwt_payload.email);
  UsersModel.findOne({user_email: jwt_payload.email}, function(err, user) {
    if (err) {
      console.log(err);
      return done(err, false);
    }
    if (user) {
      console.log("User Data " + user);
      return done(null, user);
    } else {
      console.log('else');
      return done(null, false);
      // or you could create a new account
    }
  });
}));
```



```
//Middleware to Check Auth
var passport = require('passport');

var JwtStrategy = require('passport-jwt').Strategy;
var ExtractJwt = require('passport-jwt').ExtractJwt;
var opts = {}
opts.jwtFromRequest = ExtractJwt.fromAuthHeaderAsBearerToken();
opts.secretOrKey = 'sdasdsadsakjslkjsdlfjsdklfjsdklfjsdklfjsdklfjsdkjfsddfsdfkjsdlfjsdf';
passport.use(new JwtStrategy(opts, function (jwt_payload, done) {
  console.log(jwt_payload.email);
  UsersModel.findOne({user_email: jwt_payload.email}, function (err, user) {
    if (err) {
      console.log(err);
      return done(err, false);
    }
    if (user) {
      console.log("User Data " + user);
      return done(null, user);
    } else {
      console.log('else');
      return done(null, false);
      // or you could create a new account
    }
  });
}));
}));
```

Route Auth

```
router.get('/get-all-users-api', passport.authenticate('jwt',{session:false}),function(req, res,
next) {

  UsersModel.find({}, function(err, data) {
    if (err) {
      Response.errorResponse(err, res);
    } else {
      res.send(JSON.stringify({"status": 200,"flag": 1, "message": "Users Data", "data":
data}));
    }
  });

});
```




```
//Display all records API Using JWT
router.get('/get-all-users-api',
  passport.authenticate('jwt',{session:false}),
  function(req, res, next) {

    UsersModel.find({}, function(err, data) {
      if (err) {
        Response.errorResponse(err, res);
      } else {
        res.send(JSON.stringify({"status": 200,"flag": 1, "message": "Users Data", "data": data}));
      }
    });
  });
});
```



Output

http://127.0.0.1:3000/get-all-users-api

GET http://127.0.0.1:3000/get-all-users-api

Params Authorization Headers (9) Body Pre-request Script Tests Settings

Type Bearer Token

Token eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ...

The authorization header will be automatically generated when you send the request. [Learn more about authorization](#)

Body Cookies (1) Headers (9) Test Results

Status: 200 OK Time: 53 ms Size: 795 B Save Response

Pretty Raw Preview Visualize JSON

```
1  {
2    "status": 200,
3    "flag": 1,
4    "message": "Users Data",
5    "data": [
6      {
7        "_id": "61aecb6296eb3521cc59df5b",
8        "user_name": "1",
9        "user_email": "1",
10       "user_mobile": "1",
11       "user_password": "1",
12       "__v": 0
13     },
14     {
15       "_id": "61aecb9796eb3521cc59df5c",
16       "user_name": "akash",
17       "user_email": "akash@gmail.com",
18       "user_mobile": "9898123456",
19       "user_password": "123",
20       "__v": 0
21     }
22   ]
23 }
```



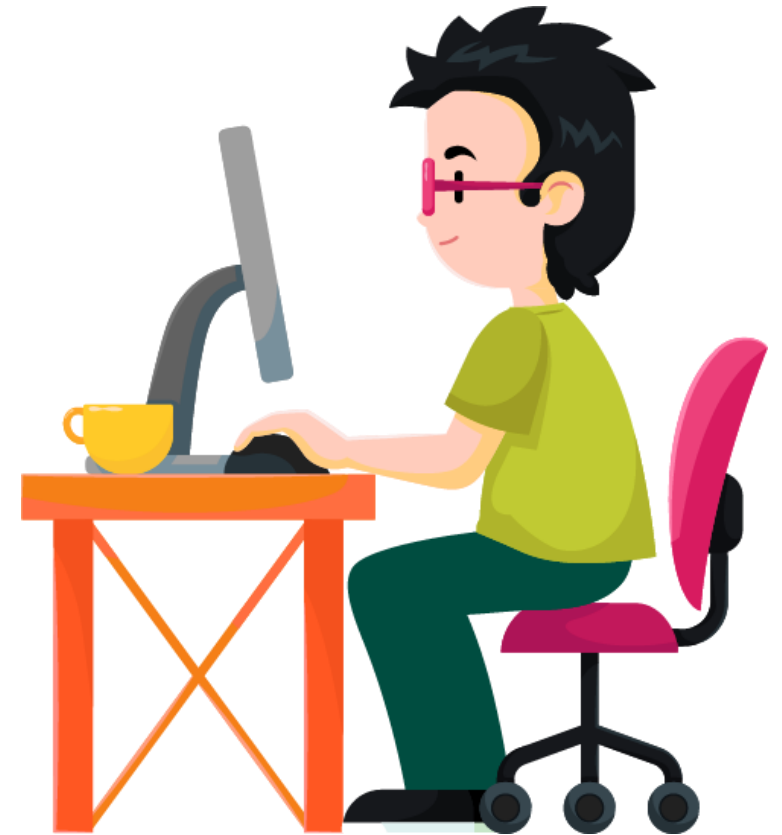
Error

The screenshot shows a REST client interface with the following details:

- URL:** `http://127.0.0.1:3000/get-all-users-api`
- Method:** GET
- Authorization:** Bearer Token with value `sdasd`
- Status:** 401 Unauthorized
- Time:** 16 ms
- Size:** 208 B
- Response Body:** `1 Unauthorized`



Get Exclusive Video Tutorials



www.apptutorials.com

<https://www.youtube.com/user/Akashtips>





Get More Details

www.akashsir.com



If You Liked It !

Rating Us Now



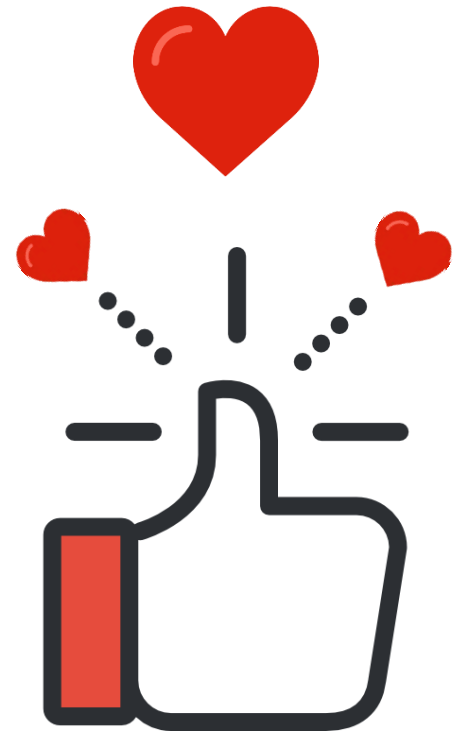
Just Dial

https://www.justdial.com/Ahmedabad/Akash-Technolabs-Navrangpura-Bus-Stop-Navrangpura/O79PXX79-XX79-170615221520-S5C4_BZDET



Sulekha

<https://www.sulekha.com/akash-technolabs-navrangpura-ahmedabad-contact-address/ahmedabad>



Connect With Me



Akash Padhiyar
#AkashSir

www.akashsir.com
www.akashtechlabs.com
www.akashpadhiyar.com
www.apptutorials.com

Social Info



Akash.padhiyar



Akashpadhiyar



Akash_padhiyar



+91 99786-21654



#Akashpadhiyar
#apptutorials