JSON

* What is JSON web token?

- JSON web Token (JWT) is a standard that defines a compact and self-contained way for securely transmitting information between parties as a JSON object.

- This information can be verified and trusted because it is digitally signed. JWTs can be signed using a secret or a public / Private key pair.

* why JSON web Token?
* ~~Only~~ There are several why applications use JSON web token for authentication

- JWT is a good choice to be passed in HTML and HTTP environments due to its smaller footprint when compared to other types of tokens.

- JSON Web Tokens can be signed using a shared secret and also by using public / private key pairs.

- It is easier to work with JWT as JSON parsers are common in most programming languages
- JWT is also suitable for implementing authorization in large-scale web applications.

* Structure of a JWT

→ Structure of a JSON web Token consists of:

- Header - consists of two parts:
  - Type of token i.e. JWT
  - Signing algorithm, ex: SHA512

```
{
  "alg" : "HS512",
  "typ" : "JWT"
}
```

- Payload - Contains the claims that provide information about a user who has been authenticated along with the other information like token expiry time.

```
{
  "sub" : "0987654321",
  "name" : "Smith Johns",
  "admin" : true
}
```

- Signature - Final part of a token that wraps in the encoded header and payload along with the algorithm and a secret

HMACSHA512{
base64URLEncode(header)+"."+
base64UrlEncode(payload),
secret)

* JWT Use Cases

→ Some scenarios where JSON Web Tokens are useful:

- Authorization - This is the most common scenario for using JWT. Once the user is logged in, each subsequent request will include the JWT, allowing the user to access routes, services, and resources that are permitted with that token.

- Information Exchange - JSON web tokens are a good way of securely transmitting information between parties.