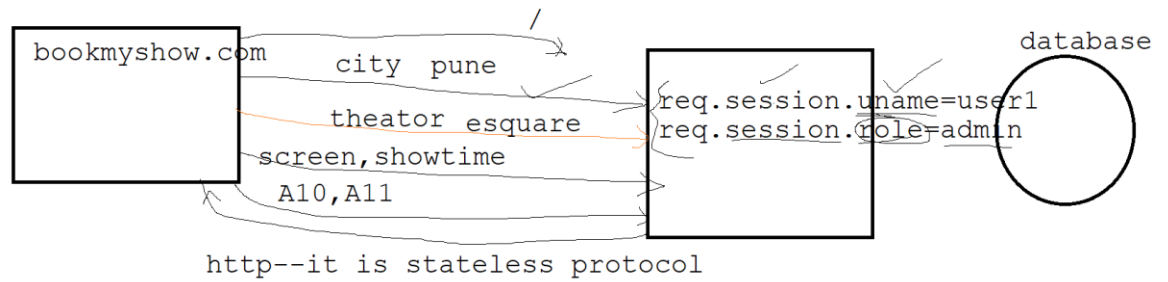Security/Atherization for web application

1. cookie – It gets saved on client machine, it is a text file of max size 4kb,
   Any user can see the cookies, so it is not secure way to store the data throughout the session.
   When you send the url to server,all the cookies associated with the url will be send to the server, and with every response it will come to client machine

2. session---It is object stored in the server. any number of keys, and value pair you may add in the session
   Usually, when you login to any application your session starts,
   And it will remain available may be based on time or till the time you logout or till the time you close the browser window.

3. token(JWT tokent, nft token)
   In web service the standard says that do not store any user data on the server, so we should not use sessions.
   Hence after validation, the server generates a token and the token will have some validity period.
   Within that period, if you are sending any request, then you need to add Autherization header in the request object.
   Autherization: "bearer"+<token>, web service will check the validity of token , and if valid it will send the response, otherwise ask for relogin

   To use Express session
   1. download express-session library
      npm I express-session –save
   2. import it in you application
      const session=require("express-session")
   3. setup the session
      app.use(session({
      // it store the secret key , which is used for encryption / decryption of session data
         secret:'mysecretkey',
        //forces the session to be saved back to session storage
         resave:true,
      //forces a session that is uninitilized to be saved in the store
         uninitialized: true
      }))

   4. set the values in the session
      req.session.uname='user1'

   5. to get the value from the session
      var name=req.session.uname;

   6. to destroy the session
      req.session.destroy(function(){
      console.log("session destroyed"})

bookmyshow.com city pune

theator esquare

screen,showtime

A10,A11

req.session.uname=user1
req.session.role=admin

database

http--it is stateless protocol

Folder structure for express js

db	all database related files will be stored in db
views	all ejs views are stored in this folder
public	all static files like css, js
routes	all routing urls related function files will be this folder
model	all models are classes, you may have created for storing data
package.json	It's a json file to store all dependencies, and its version
node_modules	this folder will be created automatically, when you will install all dependencies

<mark>To install all dependencies from package.json</mark>
npm install

In webservice following request are handled

get	read data	/products , /products/product/10
post	add data	/products/:id
put	update data	/products/:id
delete	delete data	/products/:id

To test webservice
https://www.postman.com/downloads/