Chapter 7: Authentication

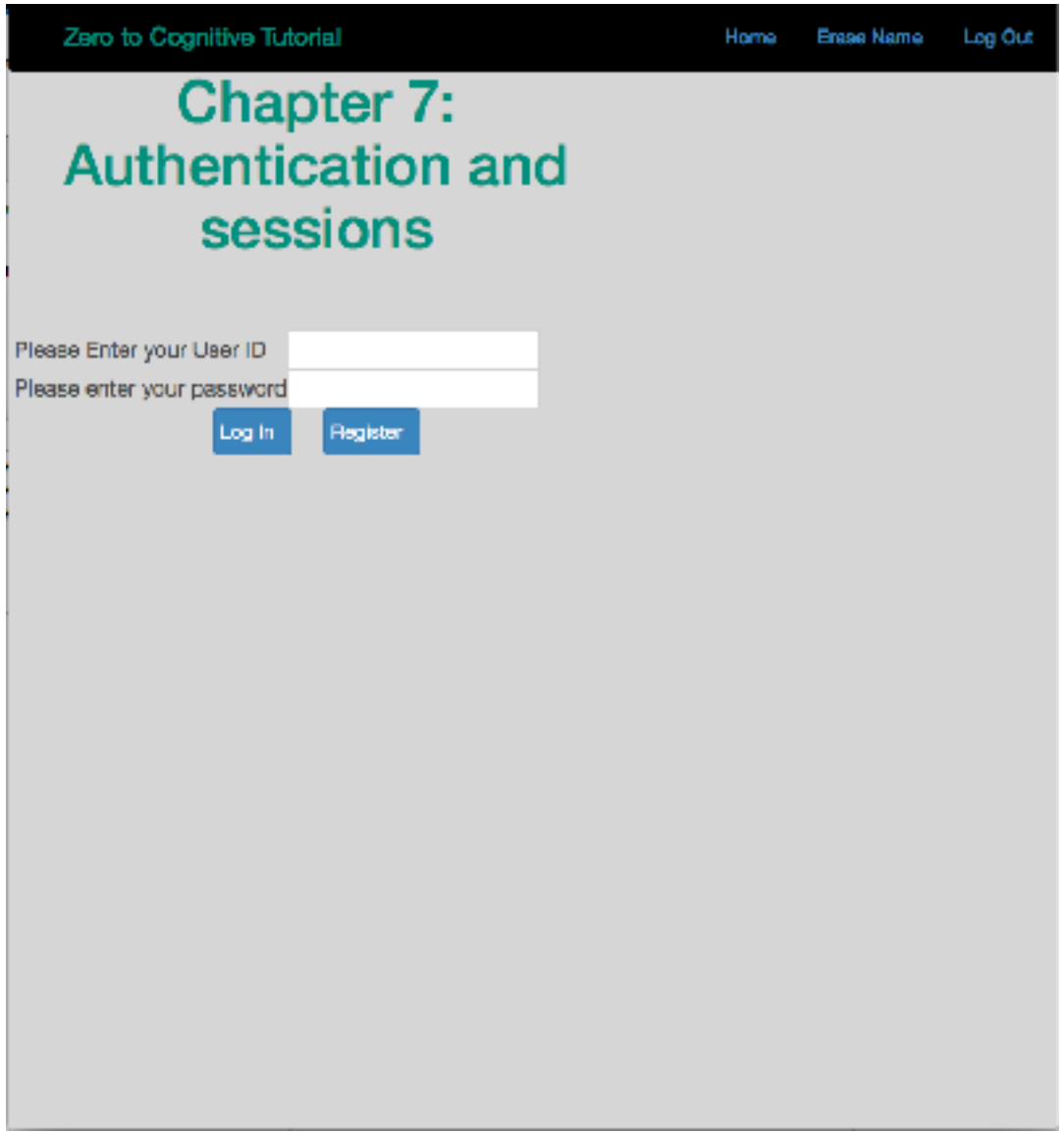
Learning Bluemix & Cognitive

Bob Dill, IBM Distinguished Engineer, CTO Global Technical Sales

Chapter 7: Creating a secure application

- SSL
- Sessions
- Cloudant
  - Storing session information
  - Storing userIDs

- Responding to the user:
  - no userid
  - no password
  - not registered
  - userid password don’t match
  - success!!
- Enabling your app to dynamically detect local or bluemix environments



New in index.js

Modules

```
var cookieParser = require('cookie-parser');
var session = require('express-session');
var cloudant = require('cloudant');
var myDB = require('./controller/restapi/features/cloudant_utils');
myDB.authenticate(myDB.create, '');
var sessionBase  = require('./controller/sessionManagement');
var sessionStore = Object.create(sessionBase.SessionObject);
```

Environment

```
var sessionSecret = env.sessionSecret;
```

Tests

```
if (cfenv.getAppEnv().isLocal == true)
```

Session persistence

```
app.use( session( {
    store: sessionStore,
    secret: sessionSecret, resave: false, saveUninitialized: true,
    cookie: {secure: true, maxAge:24*60*60*1000},
    genid: function (req) {return uuid.v4()}
  }));
```

Paths

```
app.get('/login*', function (req, res) {console.log("login session is: "+req.session); loadSelectedFile(req, res);});
```

```
router.post('/auth/authenticate*', auth.authenticate);
router.post('/auth/register*', auth.register);
router.post('/auth/logout*', auth.logout);
```

New files

- index.html updated
- body.html - new
- login_1.html - new

- z2c-login.js - new
- authenticate.js - new

- sessionManagement.js - new
- cloudant_utils.js - new

z2c-login.js

```
// utility functions
var authenticated = false;
var toLoad = "";
var login; var register; var userID = "";
function checkAuthenticated() { return(authenticated);}

function authenticate()
{
    activateLoginButtons();
}

function activateLoginButtons()
{
  loginButton = $('#login');
  registerButton = $('#register');
  loginButton.on('click',  function () {login();} );
  registerButton.on('click', function(){ register(); });
}
function login()
{

}

function register()
{

}

function logout()
{

}
```


authenticate.js

```
var encrypt = require('crypto');
var secret = require('../../env.json').sessionSecret;
// const cipher = encrypt.createCipher('aes192', secret);
// const decipher = encrypt.createDecipher('aes192', secret);

var myUsers = require('./cloudant_utils');
var u_db = 'userids';
myUsers.authenticate(myUsers.create, u_db);
exports.authenticate = function(req, res, next)
{


}
exports.register = function(req, res, next)
{


}
exports.logout = function(req, res, next)
{

}
function getCookieValue(_cookie, _name)
{
  var name = _name+"=";
  var cookie_array= _cookie.split(";");
  for (each in cookie_array)
    { var c = cookie_array[each].trim();
      if(c.indexOf(name) == 0) return(c.substring(name.length, c.length));
    }
    return("");
}
```


The Plan: 30 minute Chapters with an hour or two of practice

1. The Story, Architecture for this app	
2. Setting up Bluemix	
3. Building your first Watson App	(Watson Speech to Text)
4. Getting Watson to talk back	(Watson Text to Speech)
5. Understanding Classifiers	(Watson NLC)
6. Creating a custom dialog with Watson	(custom Q&A, session management)
7. Authentication	(puts C2 thru 6 together)
8. Alchemy News	(Watson Alchemy)
9. Visual Recognition and Images	(Watson Visual Recognition)
10. Watson Conversations	(Watson Conversations)
11. Rank & Retrieve	(Watson Alchemy + Rank & Retrieve)
12. Getting started on my first client prototype	Design Thinking, Stories, Architecture, Keeping it simple

Chapter 8: Understanding Alchemy News

Learning Bluemix & Cognitive

Bob Dill, IBM Distinguished Engineer, CTO Global Technical Sales