

```

1 // Module: IT8904 BDD
2 // Assignment 1 - CA1
3 // Student No: P7461072
4 // Student Name: ONG WEI CHUAN
5
6 // Filename: movie.js
7 // Directory: P7461072_BDD_CA1/model/
8 // This file is under the model layer and is focused in
9 // processing of data between Controller and Database
10 // Controller <--> Model (movie.js) <--> Database
11 // This file is to connect to mySQL and write fctions
12 // to query the database table and retrieve the results.
13
14 var db = require("./databaseConfig");
15
16 var movieDB = {
17     // POST /movie - Add new movie
18     addMovie: function (
19         movie_name,
20         movie_description,
21         release_date,
22         image_url,
23         genre_id,
24         active,
25         callback
26     ) {
27         var dbConn = db.getConnection(); //get configuration settings of mySQL DB
28
29         // Callback function to handle result from connection
30         dbConn.connect(function (err) {
31             // Error from connection detected
32             if (err) {
33                 console.log("movie DB connect error msg: " + err);
34                 return callback(err, null);
35             }
36             // Successful connection, proceed to do the query
37             else {
38                 console.log("Database Connected Successfully!");
39                 var sql =
40                     "insert into
41 movie(movie_name,movie_description,release_date,image_url,genre_id,active)
42 values(?,?,?,?,?,?)";
43                 dbConn.query(
44                     sql,
45                     [
46                         movie_name,
47                         movie_description,
48                         release_date,
49                         image_url,
50                         genre_id,
51                         active,
52                     ],
53                     function (err, results) {
54                         dbConn.end(); //End the connection
55                         console.log("movie sql error: " + err);
56                         console.log("movie sql results: " + JSON.stringify(results));
57
58                         return callback(err, results);
59                     }
60                 ); //dbConn.query
61             }
62         }); //dbConn.connect

```

```

61 }, //addMovie
62
63 // GET /movie?active=Y - Retrieve all active screening movies
64 // (Can also set active=N to retrieve non-active movies)
65 getActiveMovie: function (active, callback) {
66     var dbConn = db.getConnection(); //get configuration settings of mySQL DB
67
68     // Callback function to handle result from connection
69     dbConn.connect(function (err) {
70         // Error from connection detected
71         if (err) {
72             console.log("movie DB connect error msg: " + err);
73             return callback(err, null);
74         }
75         // Successful connection, proceed to do the query
76         else {
77             console.log("Database Connected Successfully!");
78
79             var sql = "select * from movie where active=?";
80             // var sql = "select * from movie";
81             dbConn.query(sql, [active], function (err, results) {
82                 dbConn.end(); //End the connection
83                 console.log("movie sql error: " + err);
84                 console.log("movie sql results: " + JSON.stringify(results));
85
86                 return callback(err, results);
87             }); //dbConn.query
88         }
89     }); //dbConn.connect
90 }, //getMovie?active=Y
91
92 // GET /movie?substr=<pat>&genreid=<num> (user is to supply substring of movies
and genreid)
93 // Retrieve movies based on substring of movie name OR genre id,
94 // sort in ascending release date
95 getSortedMovie: function (substr, genre_id, callback) {
96     var dbConn = db.getConnection(); //get configuration settings of mySQL DB
97
98     // Callback function to handle result from connection
99     dbConn.connect(function (err) {
100         // Error from connection detected
101         if (err) {
102             console.log("movie DB connect error msg: " + err);
103             return callback(err, null);
104         }
105         // Successful connection, proceed to do the query
106         else {
107             console.log("Database Connected Successfully!");
108
109             // user provides both substr and genreid
110             // if (substr !== undefined && genre_id !== undefined) {
111             var sql =
112 "select * from movie where movie_name like ? or genre_id=? order by
release_date";
113             dbConn.query(sql, [substr, genre_id], function (err, results) {
114                 dbConn.end(); //End the connection
115                 console.log("movie sql error: " + err);
116                 console.log("movie sql results: " + JSON.stringify(results));
117
118                 return callback(err, results);
119             }); //dbConn.query
120         }

```

```
121     }); //dbConn.connect
122   }, //getSortedMovie
123 }; //movieDB
124
125 module.exports = movieDB;
126
```