Z2RF1WMJ3

JAVA ADVANCE

JANUARY 14, 2019

JAVA ADVANCE PROJECT *– JSP, DATABASE & WEB SERVICE PROGRAMMING*

Table of Contents

Program details.

Description of the program:

The program is a distributed application, that can be used as a forum for various topics. The interface is a website only. And users are able to register with their personal information. Users also have the ability to create topics (posts), post comments and post replies to other user's comments. Comments & replies are viewable by all users. The web service provides all the functionality & **generates a log file to the desktop.**

Instructions on which OS to use:

Any platform can be used to run the program.

Type of & version of programs used:

NetBeans IDE 8.2 was used on the Windows 10 operating system. Microsoft SQL Server Management Studio v17 was used to develop the database.

What do you need to run the program?

The following resources are needed to run the program.

- Microsoft SQL Server Management Studio. **https://www.microsoft.com/en-us/sql-server/sql-server-downloads**
- Microsoft JDBC Driver 6.2 for SQL server. (**Included in root folder)**
- Java SDK. **https://www.oracle.com/technetwork/java/javase/downloads/index.html**
- NetBeans IDE. **https://netbeans.org/downloads/8.0.2/**
- Glassfish Server (I'm using glassfish server, you can use apache or similar). Will work flawlessly on Glassfish. It is tested on using Glassfish Server. **https://www.oracle.com/technetwork/java/javaee/downloads/index.html**
- A web browser. Firefox is preferred. It is faster.

How to run the program?

To open the program the following are the instructions: (Assuming you are going to run on a newly installed windows OS, if not you can start from step 6).

1) Install Java SDK (this will include the jdk).
2) Install Microsoft SQL Server Management Studio & SQL Server Express Edition.
3) Install NetBeans IDE 8.2 & above is preferred.
4) Download Java EE SDK from their website (this contains GlassFish server).
5) Download the Microsoft JDBC Driver 6.2 for SQL driver ***(included in project dir.)***
6) Run NetBeans IDE (Assuming you have GlassFish server installed)
7) Click on File > Open Project > browse for project name **"Forum2"**
8) Expand **Forum2** in Projects Window > Expand **Web Pages** > Expand **Foxcrypt** > Right click on **Topics.jsp** and click **Run File**.

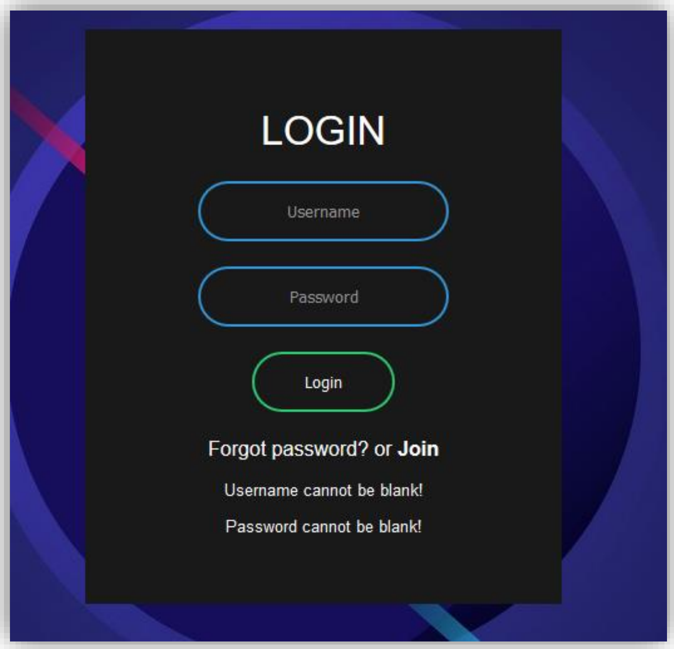
To install GlassFish server:

1) Click on Services window.
2) Right click on Servers. And click **Add Server**.
3) Select **GlassFish Server**. Select Location of GlassFish Server **(located in Java EE folder)** & click Finish.

Error handling & user input.

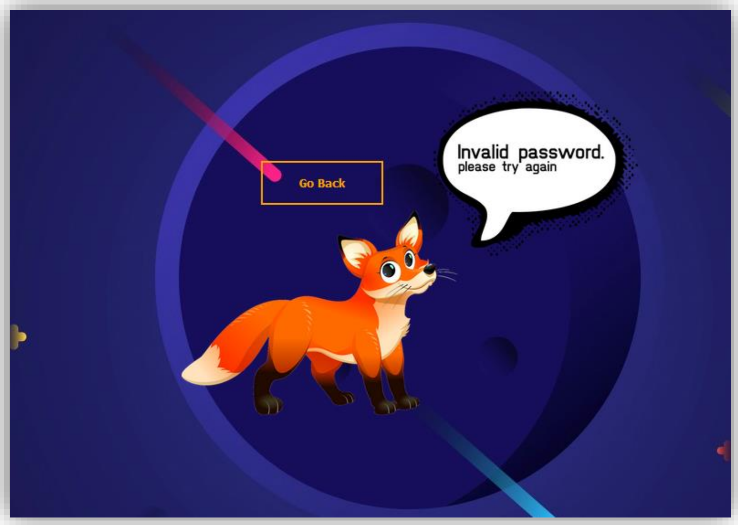
For error handling, **try…catch** blocks are used & exceptions are caught. Regular expressions is used for email validation. If statements are mostly used to validate input fields. Errors and exceptions are printed to console. Error pages are also used to handle certain errors. And provide the user with alternate options.

Example of handling empty fields.



Username cannot be blank!

Password cannot be blank!

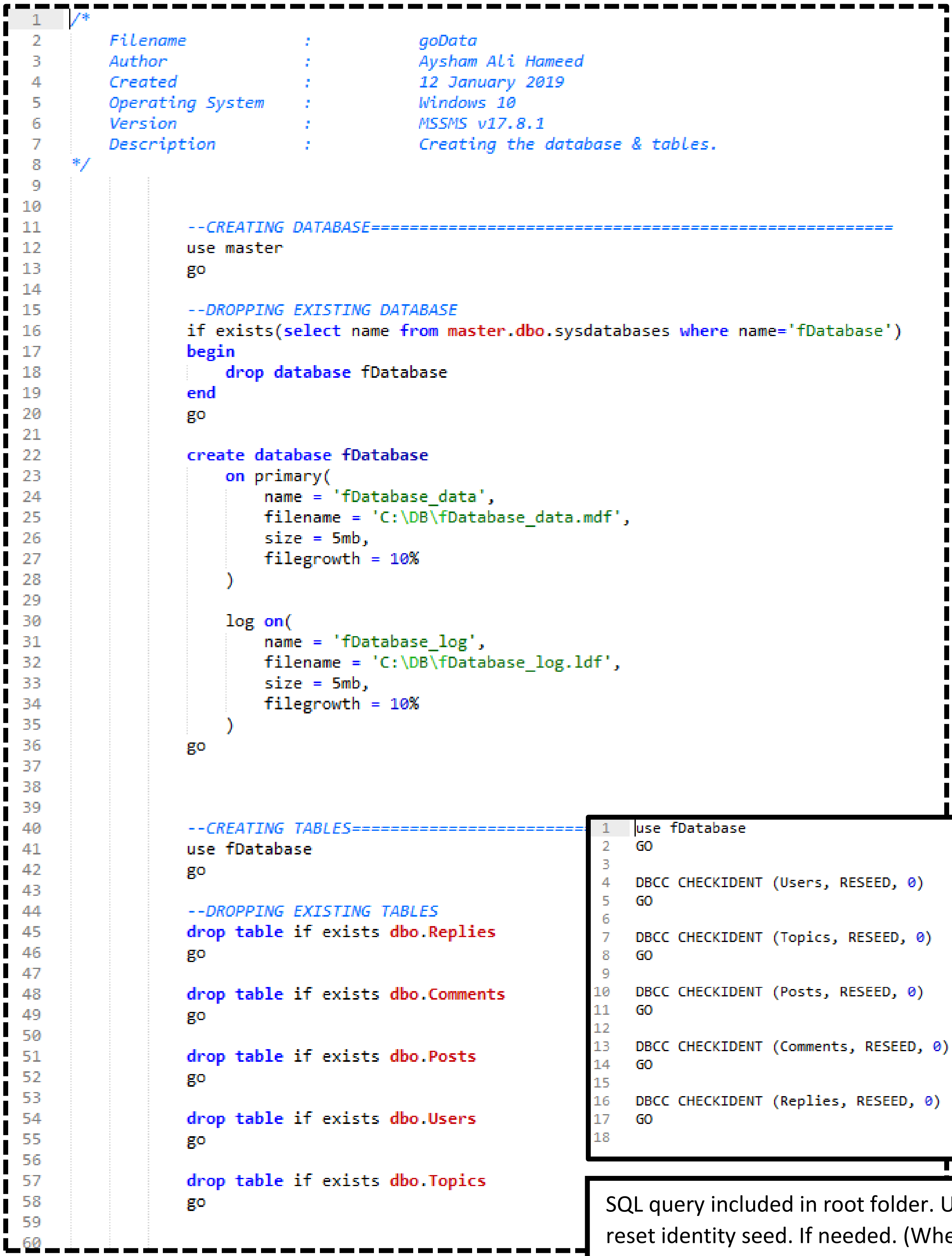


Invalid password.
please try again

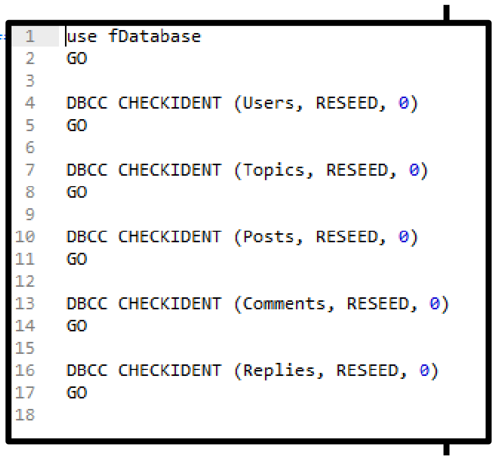
Example of handling invalid password.

Creating the database.

This SQL query creates a database.



```
/*
    Filename            :           goData
    Author              :           Aysham Ali Hameed
    Created             :           12 January 2019
    Operating System    :           Windows 10
    Version             :           MSSMS v17.8.1
    Description         :           Creating the database & tables.
*/


        --CREATING DATABASE==========================================================
        use master
        go

        --DROPPING EXISTING DATABASE
        if exists(select name from master.dbo.sysdatabases where name='fDatabase')
        begin
            drop database fDatabase
        end
        go

        create database fDatabase
            on primary(
                name = 'fDatabase_data',
                filename = 'C:\DB\fDatabase_data.mdf',
                size = 5mb,
                filegrowth = 10%
            )

            log on(
                name = 'fDatabase_log',
                filename = 'C:\DB\fDatabase_log.ldf',
                size = 5mb,
                filegrowth = 10%
            )
        go



        --CREATING TABLES========================
        use fDatabase
        go

        --DROPPING EXISTING TABLES
        drop table if exists dbo.Replies
        go

        drop table if exists dbo.Comments
        go

        drop table if exists dbo.Posts
        go

        drop table if exists dbo.Users
        go

        drop table if exists dbo.Topics
        go
```

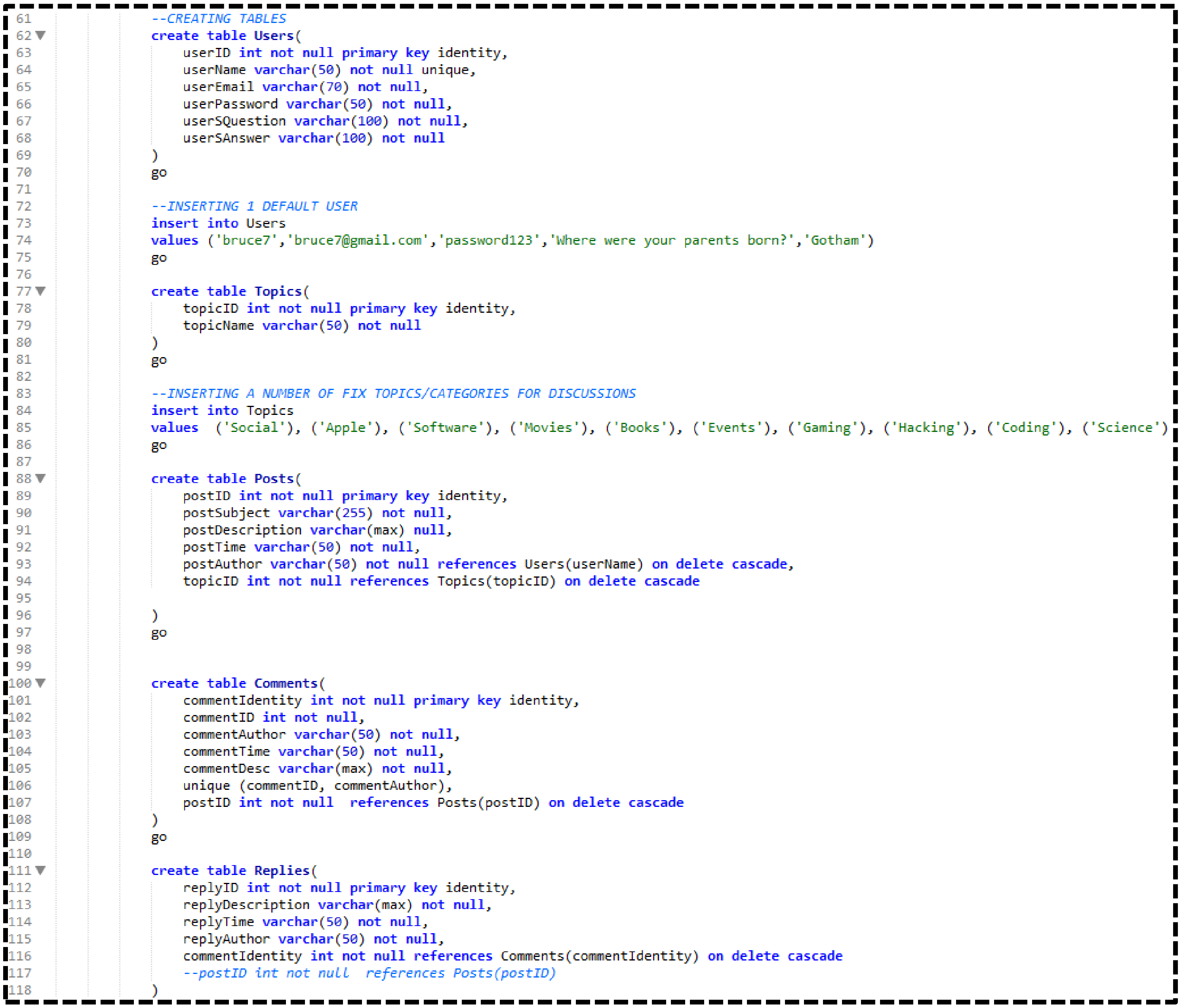


```
use fDatabase
GO

DBCC CHECKIDENT (Users, RESEED, 0)
GO

DBCC CHECKIDENT (Topics, RESEED, 0)
GO

DBCC CHECKIDENT (Posts, RESEED, 0)
GO

DBCC CHECKIDENT (Comments, RESEED, 0)
GO

DBCC CHECKIDENT (Replies, RESEED, 0)
GO

```

SQL query included in root folder. Used to reset identity seed. If needed. (When order of insertion gets messed up)

Creating tables.

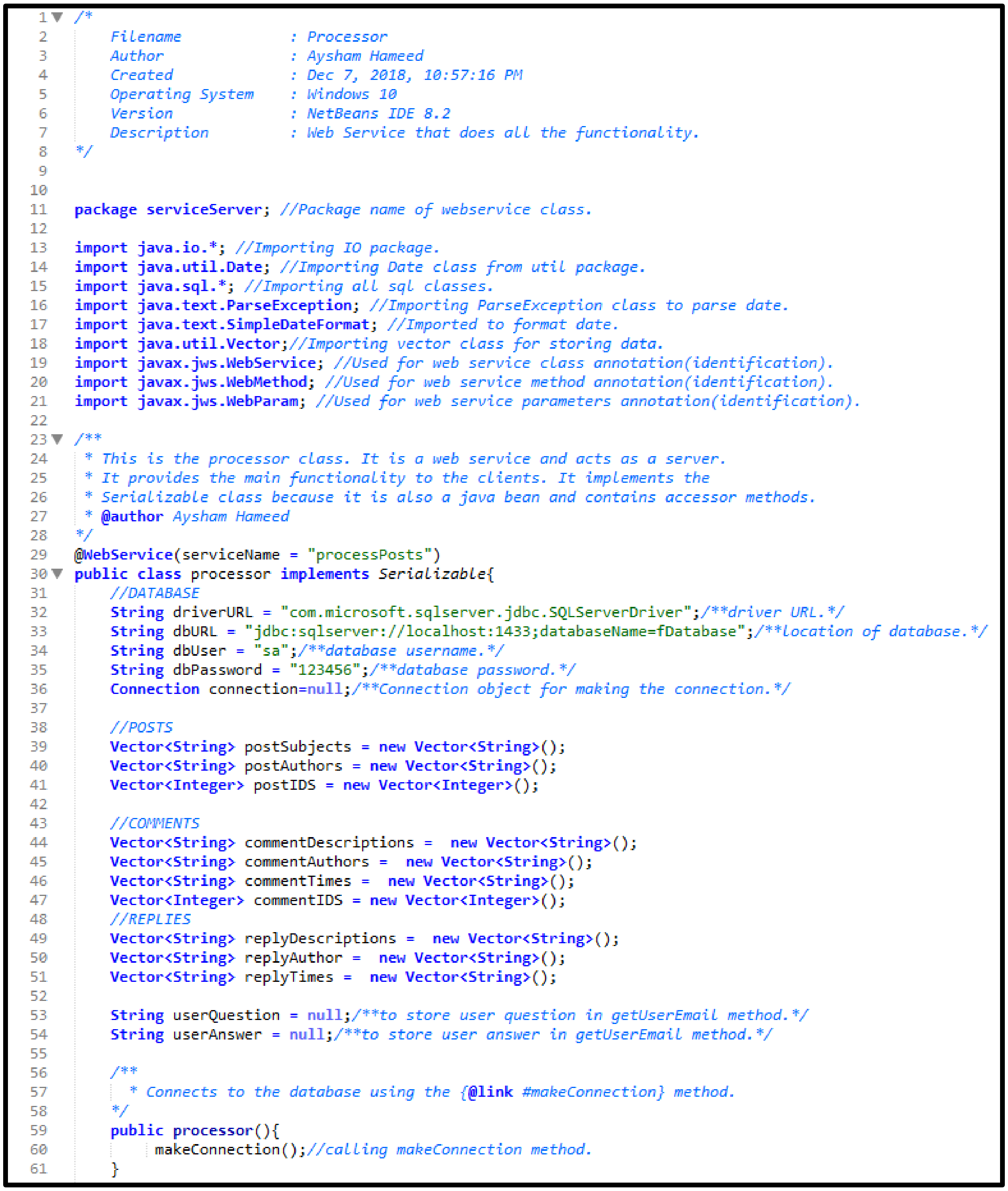
Creates tables inside the database.



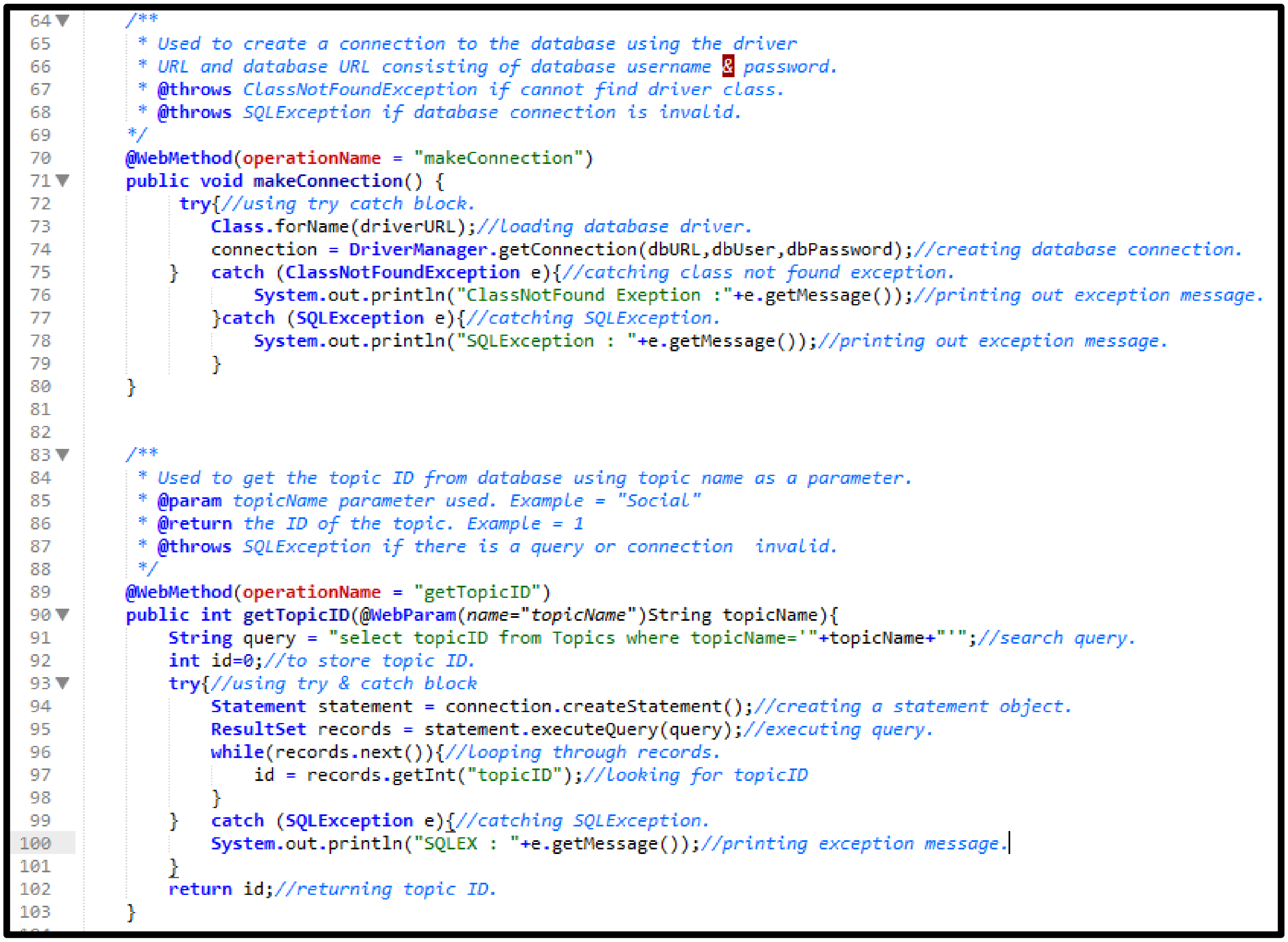
```sql
--CREATING TABLES
create table Users(
    userID int not null primary key identity,
    userName varchar(50) not null unique,
    userEmail varchar(70) not null,
    userPassword varchar(50) not null,
    userSQuestion varchar(100) not null,
    userSAnswer varchar(100) not null
)
go

--INSERTING 1 DEFAULT USER
insert into Users
values ('bruce7','bruce7@gmail.com','password123','Where were your parents born?','Gotham')
go

create table Topics(
    topicID int not null primary key identity,
    topicName varchar(50) not null
)
go

--INSERTING A NUMBER OF FIX TOPICS/CATEGORIES FOR DISCUSSIONS
insert into Topics
values  ('Social'), ('Apple'), ('Software'), ('Movies'), ('Books'), ('Events'), ('Gaming'), ('Hacking'), ('Coding'), ('Science')
go

create table Posts(
    postID int not null primary key identity,
    postSubject varchar(255) not null,
    postDescription varchar(max) null,
    postTime varchar(50) not null,
    postAuthor varchar(50) not null references Users(userName) on delete cascade,
    topicID int not null references Topics(topicID) on delete cascade

)
go


create table Comments(
    commentIdentity int not null primary key identity,
    commentID int not null,
    commentAuthor varchar(50) not null,
    commentTime varchar(50) not null,
    commentDesc varchar(max) not null,
    unique (commentID, commentAuthor),
    postID int not null  references Posts(postID) on delete cascade
)
go

create table Replies(
    replyID int not null primary key identity,
    replyDescription varchar(max) not null,
    replyTime varchar(50) not null,
    replyAuthor varchar(50) not null,
    commentIdentity int not null references Comments(commentIdentity) on delete cascade
    --postID int not null  references Posts(postID)
)
```

Web Service

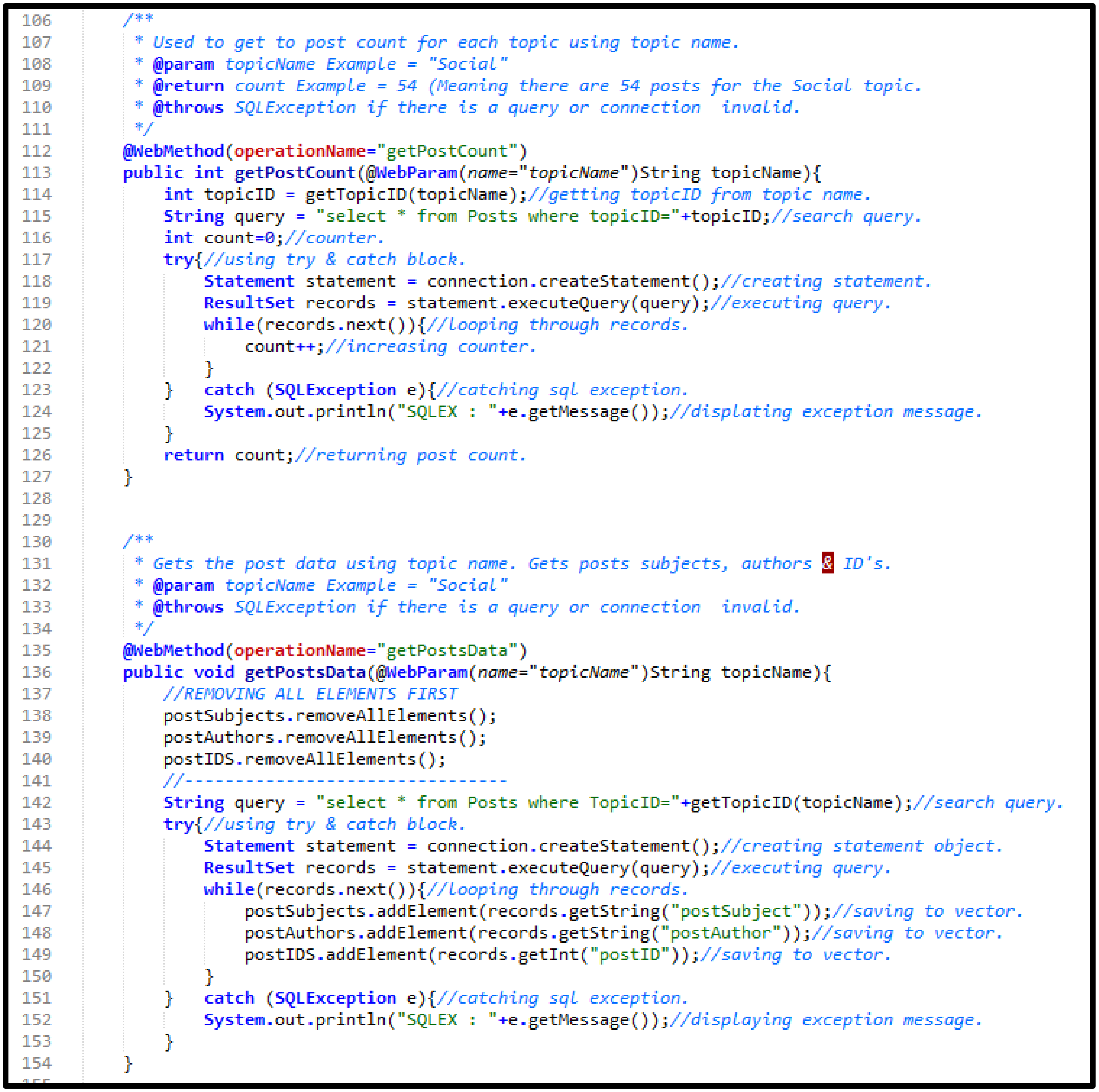
Processor (Web service provider). Responsible for all the processing.



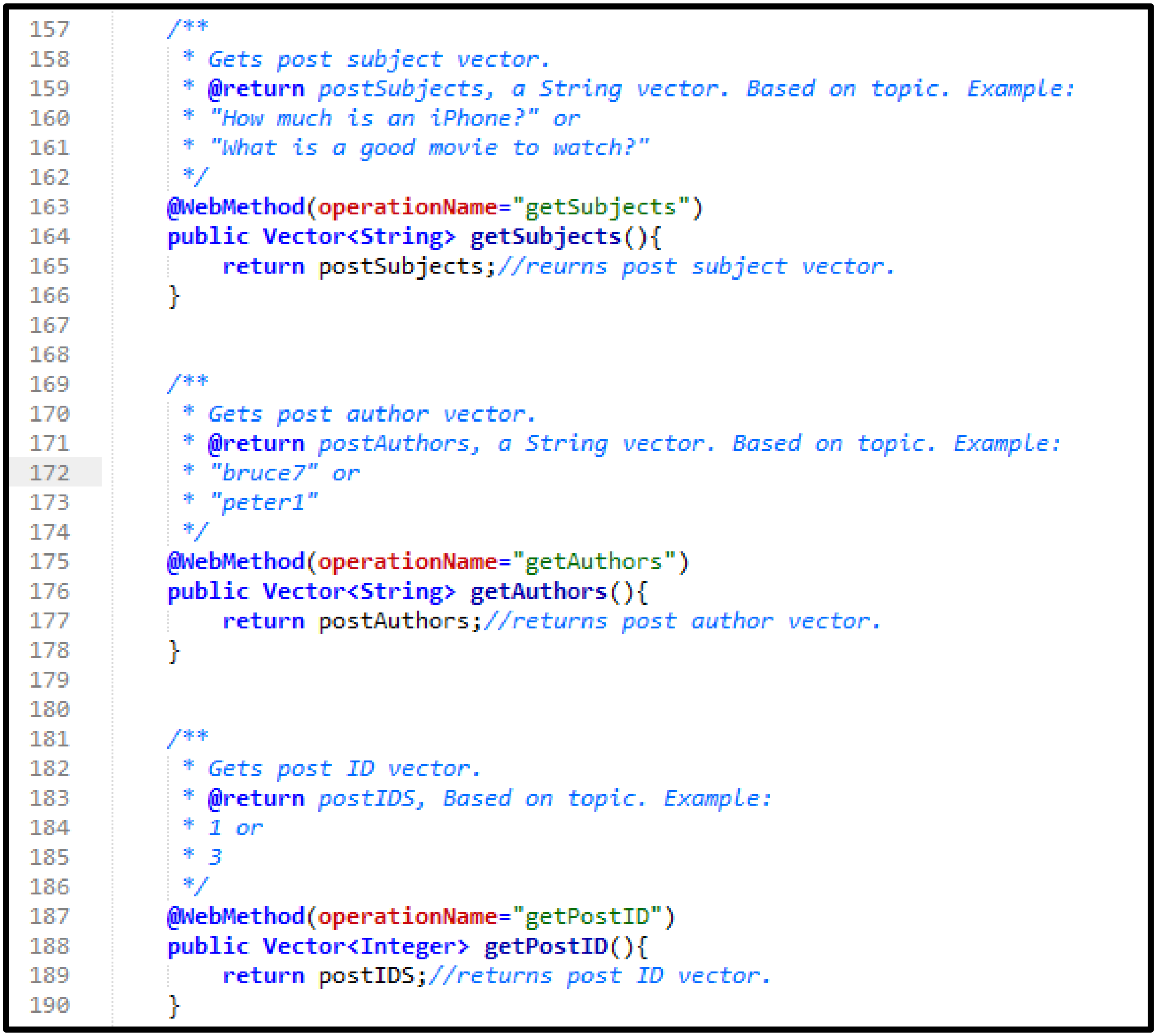
```java
/*
    Filename                : Processor
    Author                  : Aysham Hameed
    Created                 : Dec 7, 2018, 10:57:16 PM
    Operating System        : Windows 10
    Version                 : NetBeans IDE 8.2
    Description             : Web Service that does all the functionality.
*/


package serviceServer; //Package name of webservice class.

import java.io.*; //Importing IO package.
import java.util.Date; //Importing Date class from util package.
import java.sql.*; //Importing all sql classes.
import java.text.ParseException; //Importing ParseException class to parse date.
import java.text.SimpleDateFormat; //Imported to format date.
import java.util.Vector;//Importing vector class for storing data.
import javax.jws.WebService; //Used for web service class annotation(identification).
import javax.jws.WebMethod; //Used for web service method annotation(identification).
import javax.jws.WebParam; //Used for web service parameters annotation(identification).

/**
 * This is the processor class. It is a web service and acts as a server.
 * It provides the main functionality to the clients. It implements the
 * Serializable class because it is also a java bean and contains accessor methods.
 * @author Aysham Hameed
*/
@WebService(serviceName = "processPosts")
public class processor implements Serializable{
    //DATABASE
    String driverURL = "com.microsoft.sqlserver.jdbc.SQLServerDriver";/**driver URL.*/
    String dbURL = "jdbc:sqlserver://localhost:1433;databaseName=fDatabase";/**location of database.*/
    String dbUser = "sa";/**database username.*/
    String dbPassword = "123456";/**database password.*/
    Connection connection=null;/**Connection object for making the connection.*/

    //POSTS
    Vector<String> postSubjects = new Vector<String>();
    Vector<String> postAuthors = new Vector<String>();
    Vector<Integer> postIDS = new Vector<Integer>();

    //COMMENTS
    Vector<String> commentDescriptions =  new Vector<String>();
    Vector<String> commentAuthors =  new Vector<String>();
    Vector<String> commentTimes =  new Vector<String>();
    Vector<Integer> commentIDS = new Vector<Integer>();
    //REPLIES
    Vector<String> replyDescriptions =  new Vector<String>();
    Vector<String> replyAuthor =  new Vector<String>();
    Vector<String> replyTimes =  new Vector<String>();

    String userQuestion = null;/**to store user question in getUserEmail method.*/
    String userAnswer = null;/**to store user answer in getUserEmail method.*/

    /**
     * Connects to the database using the {@link #makeConnection} method.
    */
    public processor(){
        makeConnection();//calling makeConnection method.
    }
```



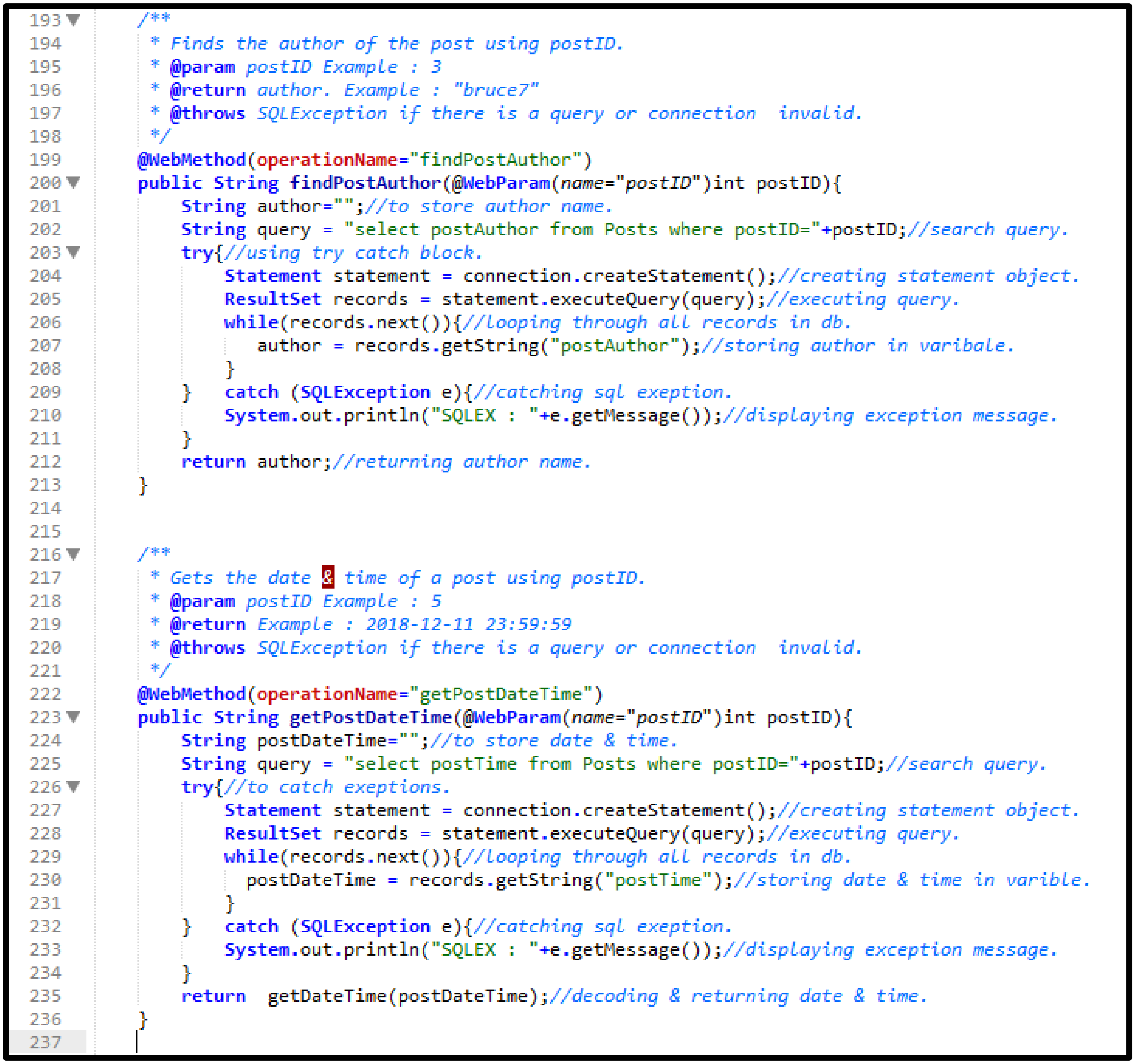
```
    /**
     * Used to create a connection to the database using the driver
     * URL and database URL consisting of database username & password.
     * @throws ClassNotFoundException if cannot find driver class.
     * @throws SQLException if database connection is invalid.
    */
    @WebMethod(operationName = "makeConnection")
    public void makeConnection() {
         try{//using try catch block.
            Class.forName(driverURL);//loading database driver.
            connection = DriverManager.getConnection(dbURL,dbUser,dbPassword);//creating database connection.
        }   catch (ClassNotFoundException e){//catching class not found exception.
                System.out.println("ClassNotFound Exeption :"+e.getMessage());//printing out exception message.
            }catch (SQLException e){//catching SQLException.
                System.out.println("SQLException : "+e.getMessage());//printing out exception message.
            }
    }


    /**
     * Used to get the topic ID from database using topic name as a parameter.
     * @param topicName parameter used. Example = "Social"
     * @return the ID of the topic. Example = 1
     * @throws SQLException if there is a query or connection  invalid.
     */
    @WebMethod(operationName = "getTopicID")
    public int getTopicID(@WebParam(name="topicName")String topicName){
        String query = "select topicID from Topics where topicName='"+topicName+"'";//search query.
        int id=0;//to store topic ID.
        try{//using try & catch block
            Statement statement = connection.createStatement();//creating a statement object.
            ResultSet records = statement.executeQuery(query);//executing query.
            while(records.next()){//looping through records.
                id = records.getInt("topicID");//looking for topicID
            }
        }   catch (SQLException e){//catching SQLException.
            System.out.println("SQLEX : "+e.getMessage());//printing exception message.
        }
        return id;//returning topic ID.
    }
```



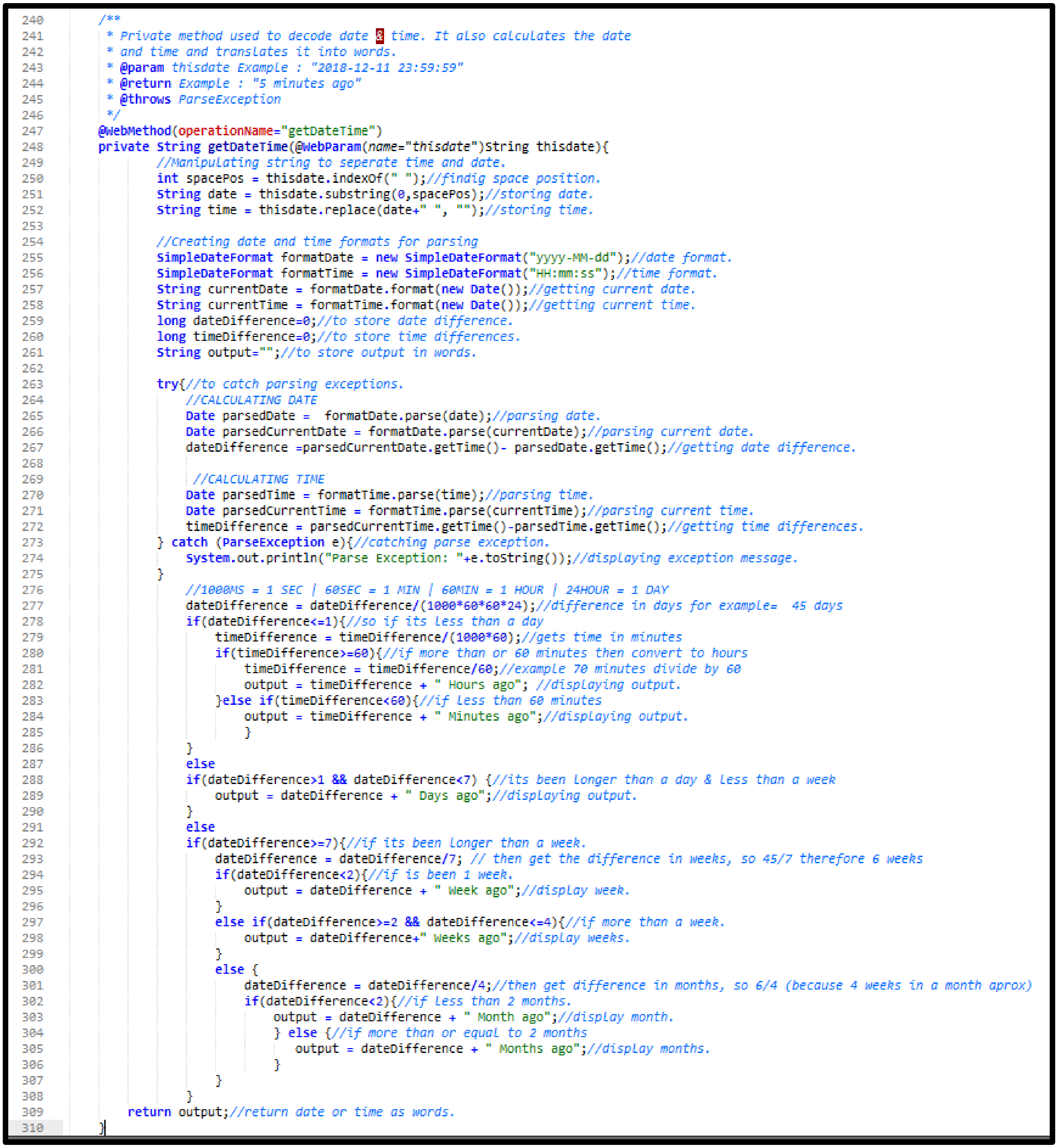
```java
    /**
     * Used to get to post count for each topic using topic name.
     * @param topicName Example = "Social"
     * @return count Example = 54 (Meaning there are 54 posts for the Social topic.
     * @throws SQLException if there is a query or connection  invalid.
     */
    @WebMethod(operationName="getPostCount")
    public int getPostCount(@WebParam(name="topicName")String topicName){
        int topicID = getTopicID(topicName);//getting topicID from topic name.
        String query = "select * from Posts where topicID="+topicID;//search query.
        int count=0;//counter.
        try{//using try & catch block.
            Statement statement = connection.createStatement();//creating statement.
            ResultSet records = statement.executeQuery(query);//executing query.
            while(records.next()){//looping through records.
                count++;//increasing counter.
            }
        }   catch (SQLException e){//catching sql exception.
            System.out.println("SQLEX : "+e.getMessage());//displating exception message.
        }
        return count;//returning post count.
    }


    /**
     * Gets the post data using topic name. Gets posts subjects, authors & ID's.
     * @param topicName Example = "Social"
     * @throws SQLException if there is a query or connection  invalid.
     */
    @WebMethod(operationName="getPostsData")
    public void getPostsData(@WebParam(name="topicName")String topicName){
        //REMOVING ALL ELEMENTS FIRST
        postSubjects.removeAllElements();
        postAuthors.removeAllElements();
        postIDS.removeAllElements();
        //--------------------------------
        String query = "select * from Posts where TopicID="+getTopicID(topicName);//search query.
        try{//using try & catch block.
            Statement statement = connection.createStatement();//creating statement object.
            ResultSet records = statement.executeQuery(query);//executing query.
            while(records.next()){//looping through records.
                postSubjects.addElement(records.getString("postSubject"));//saving to vector.
                postAuthors.addElement(records.getString("postAuthor"));//saving to vector.
                postIDS.addElement(records.getInt("postID"));//saving to vector.
            }
        }   catch (SQLException e){//catching sql exception.
            System.out.println("SQLEX : "+e.getMessage());//displaying exception message.
        }
    }
```



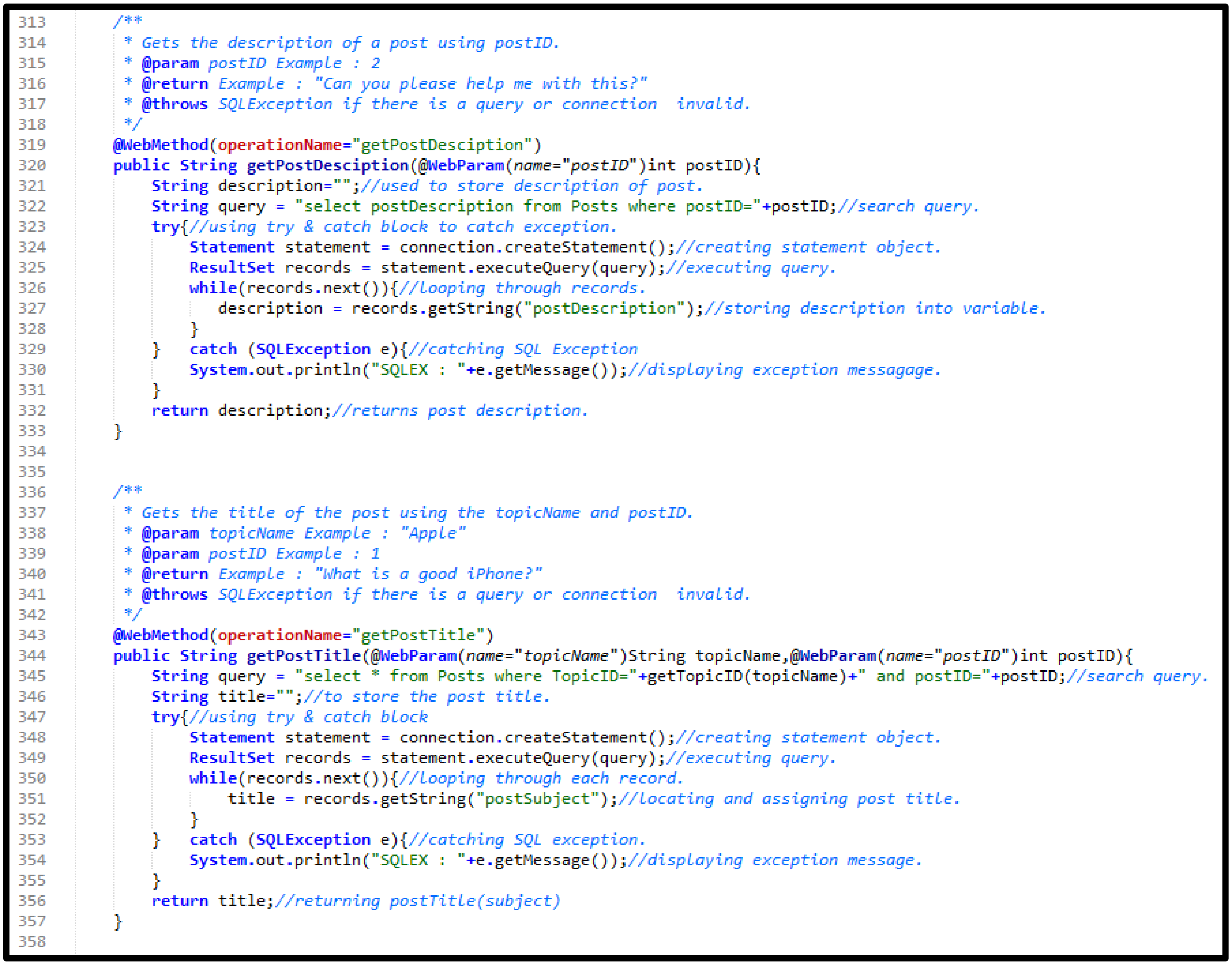
```java
    /**
     * Gets post subject vector.
     * @return postSubjects, a String vector. Based on topic. Example:
     * "How much is an iPhone?" or
     * "What is a good movie to watch?"
     */
    @WebMethod(operationName="getSubjects")
    public Vector<String> getSubjects(){
        return postSubjects;//reurns post subject vector.
    }


    /**
     * Gets post author vector.
     * @return postAuthors, a String vector. Based on topic. Example:
     * "bruce7" or
     * "peter1"
     */
    @WebMethod(operationName="getAuthors")
    public Vector<String> getAuthors(){
        return postAuthors;//returns post author vector.
    }


    /**
     * Gets post ID vector.
     * @return postIDS, Based on topic. Example:
     * 1 or
     * 3
     */
    @WebMethod(operationName="getPostID")
    public Vector<Integer> getPostID(){
        return postIDS;//returns post ID vector.
    }
```



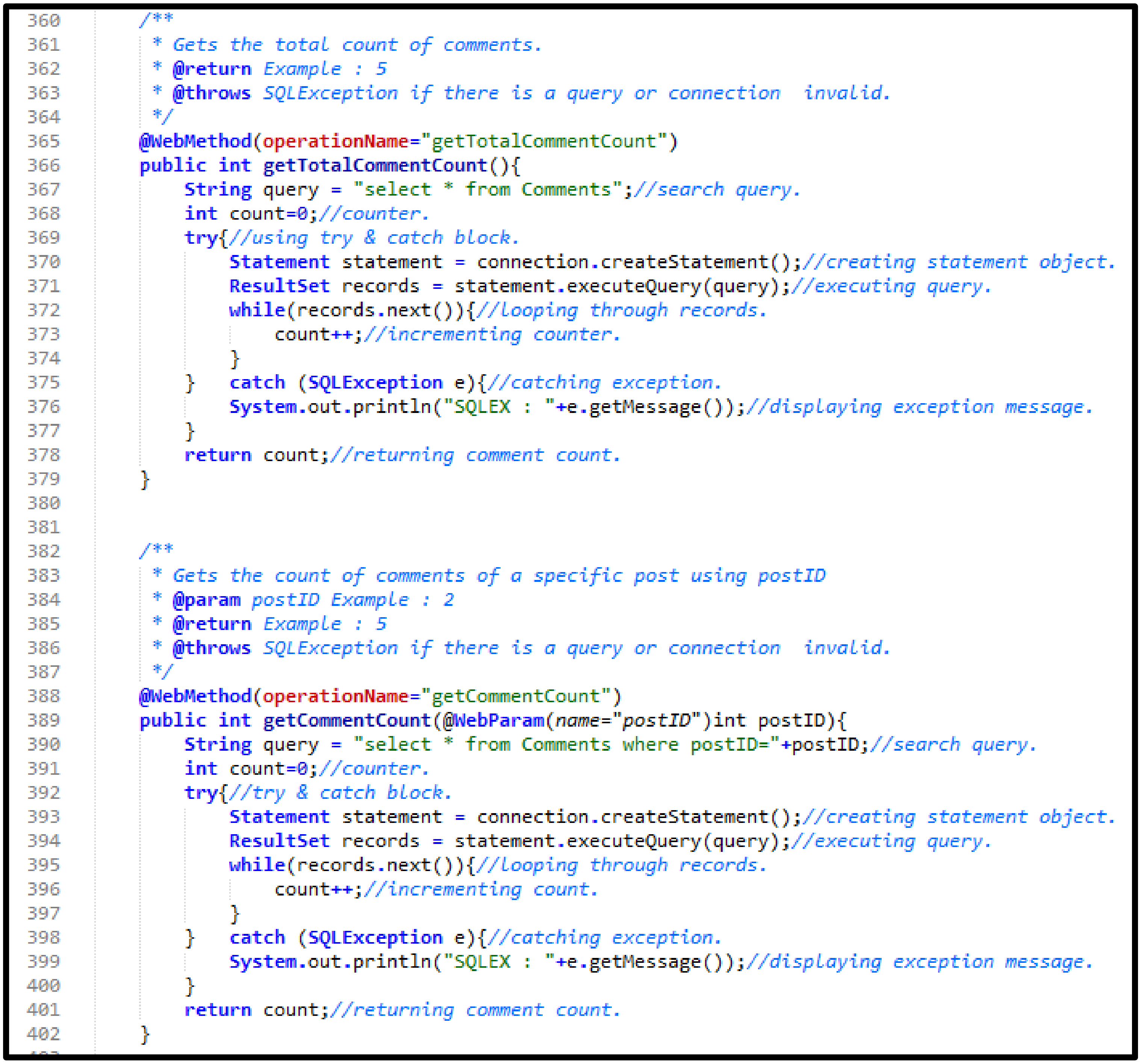
```java
    /**
     * Finds the author of the post using postID.
     * @param postID Example : 3
     * @return author. Example : "bruce7"
     * @throws SQLException if there is a query or connection  invalid.
     */
    @WebMethod(operationName="findPostAuthor")
    public String findPostAuthor(@WebParam(name="postID")int postID){
        String author="";//to store author name.
        String query = "select postAuthor from Posts where postID="+postID;//search query.
        try{//using try catch block.
            Statement statement = connection.createStatement();//creating statement object.
            ResultSet records = statement.executeQuery(query);//executing query.
            while(records.next()){//looping through all records in db.
               author = records.getString("postAuthor");//storing author in varibale.
            }
        }   catch (SQLException e){//catching sql exeption.
            System.out.println("SQLEX : "+e.getMessage());//displaying exception message.
        }
        return author;//returning author name.
    }


    /**
     * Gets the date & time of a post using postID.
     * @param postID Example : 5
     * @return Example : 2018-12-11 23:59:59
     * @throws SQLException if there is a query or connection  invalid.
     */
    @WebMethod(operationName="getPostDateTime")
    public String getPostDateTime(@WebParam(name="postID")int postID){
        String postDateTime="";//to store date & time.
        String query = "select postTime from Posts where postID="+postID;//search query.
        try{//to catch exeptions.
            Statement statement = connection.createStatement();//creating statement object.
            ResultSet records = statement.executeQuery(query);//executing query.
            while(records.next()){//looping through all records in db.
              postDateTime = records.getString("postTime");//storing date & time in varible.
            }
        }   catch (SQLException e){//catching sql exeption.
            System.out.println("SQLEX : "+e.getMessage());//displaying exception message.
        }
        return  getDateTime(postDateTime);//decoding & returning date & time.
    }
```



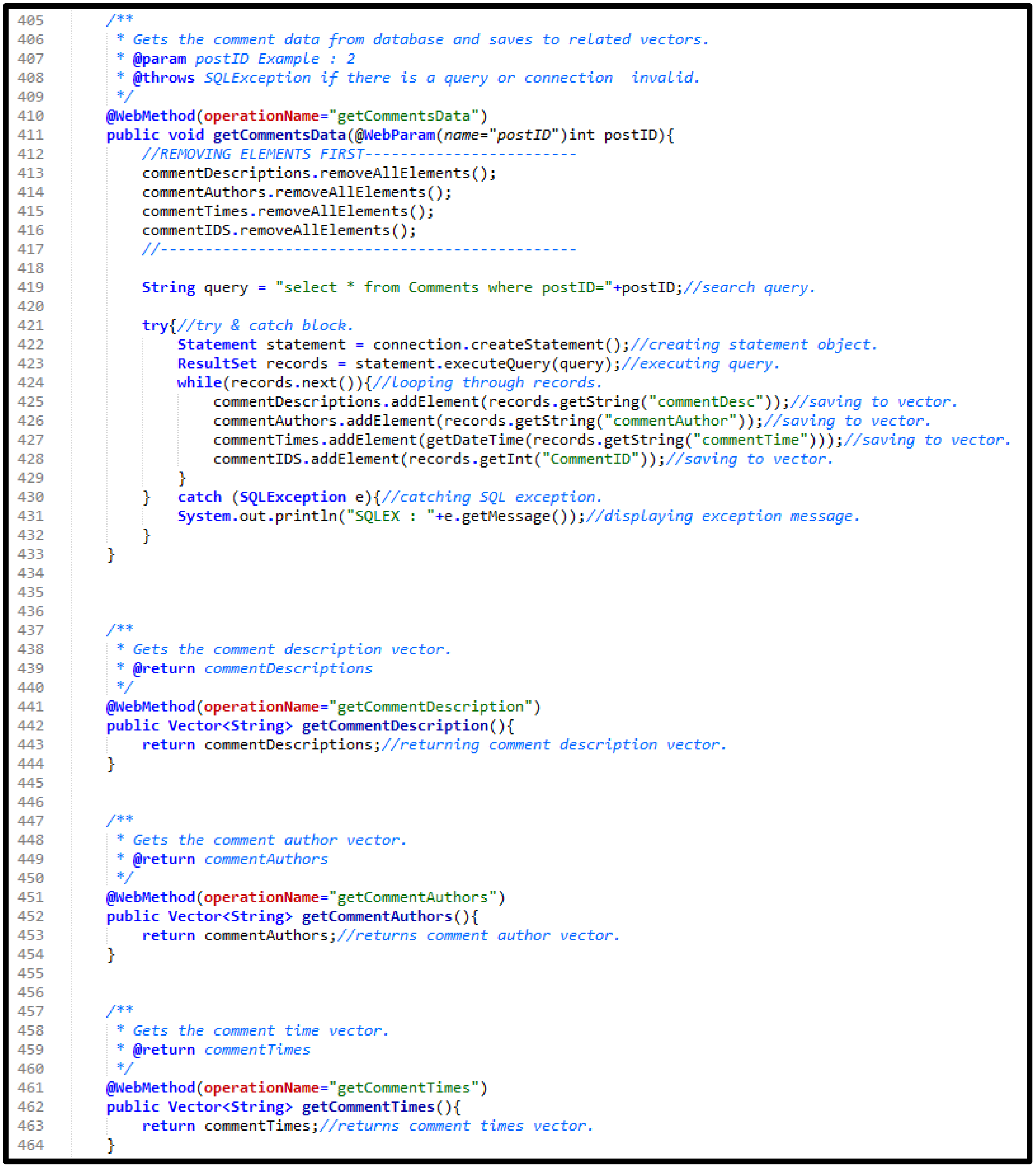
```
    /**
     * Private method used to decode date & time. It also calculates the date
     * and time and translates it into words.
     * @param thisdate Example : "2018-12-11 23:59:59"
     * @return Example : "5 minutes ago"
     * @throws ParseException
     */
    @WebMethod(operationName="getDateTime")
    private String getDateTime(@WebParam(name="thisdate")String thisdate){
            //Manipulating string to seperate time and date.
            int spacePos = thisdate.indexOf(" ");//findig space position.
            String date = thisdate.substring(0,spacePos);//storing date.
            String time = thisdate.replace(date+" ", "");//storing time.

            //Creating date and time formats for parsing
            SimpleDateFormat formatDate = new SimpleDateFormat("yyyy-MM-dd");//date format.
            SimpleDateFormat formatTime = new SimpleDateFormat("HH:mm:ss");//time format.
            String currentDate = formatDate.format(new Date());//getting current date.
            String currentTime = formatTime.format(new Date());//getting current time.
            long dateDifference=0;//to store date difference.
            long timeDifference=0;//to store time differences.
            String output="";//to store output in words.

            try{//to catch parsing exceptions.
                //CALCULATING DATE
                Date parsedDate =  formatDate.parse(date);//parsing date.
                Date parsedCurrentDate = formatDate.parse(currentDate);//parsing current date.
                dateDifference =parsedCurrentDate.getTime()- parsedDate.getTime();//getting date difference.

                 //CALCULATING TIME
                Date parsedTime = formatTime.parse(time);//parsing time.
                Date parsedCurrentTime = formatTime.parse(currentTime);//parsing current time.
                timeDifference = parsedCurrentTime.getTime()-parsedTime.getTime();//getting time differences.
            } catch (ParseException e){//catching parse exception.
                System.out.println("Parse Exception: "+e.toString());//displaying exception message.
            }
                //1000MS = 1 SEC | 60SEC = 1 MIN | 60MIN = 1 HOUR | 24HOUR = 1 DAY
                dateDifference = dateDifference/(1000*60*60*24);//difference in days for example=  45 days
                if(dateDifference<=1){//so if its less than a day
                    timeDifference = timeDifference/(1000*60);//gets time in minutes
                    if(timeDifference>=60){//if more than or 60 minutes then convert to hours
                        timeDifference = timeDifference/60;//example 70 minutes divide by 60
                        output = timeDifference + " Hours ago"; //displaying output.
                    }else if(timeDifference<60){//if less than 60 minutes
                        output = timeDifference + " Minutes ago";//displaying output.
                        }
                }
                else
                if(dateDifference>1 && dateDifference<7) {//its been longer than a day & less than a week
                    output = dateDifference + " Days ago";//displaying output.
                }
                else
                if(dateDifference>=7){//if its been longer than a week.
                    dateDifference = dateDifference/7; // then get the difference in weeks, so 45/7 therefore 6 weeks
                    if(dateDifference<2){//if is been 1 week.
                        output = dateDifference + " Week ago";//display week.
                    }
                    else if(dateDifference>=2 && dateDifference<=4){//if more than a week.
                        output = dateDifference+" Weeks ago";//display weeks.
                    }
                    else {
                        dateDifference = dateDifference/4;//then get difference in months, so 6/4 (because 4 weeks in a month aprox)
                        if(dateDifference<2){//if less than 2 months.
                            output = dateDifference + " Month ago";//display month.
                        } else {//if more than or equal to 2 months
                           output = dateDifference + " Months ago";//display months.
                        }
                    }
                }
        return output;//return date or time as words.
    }
```



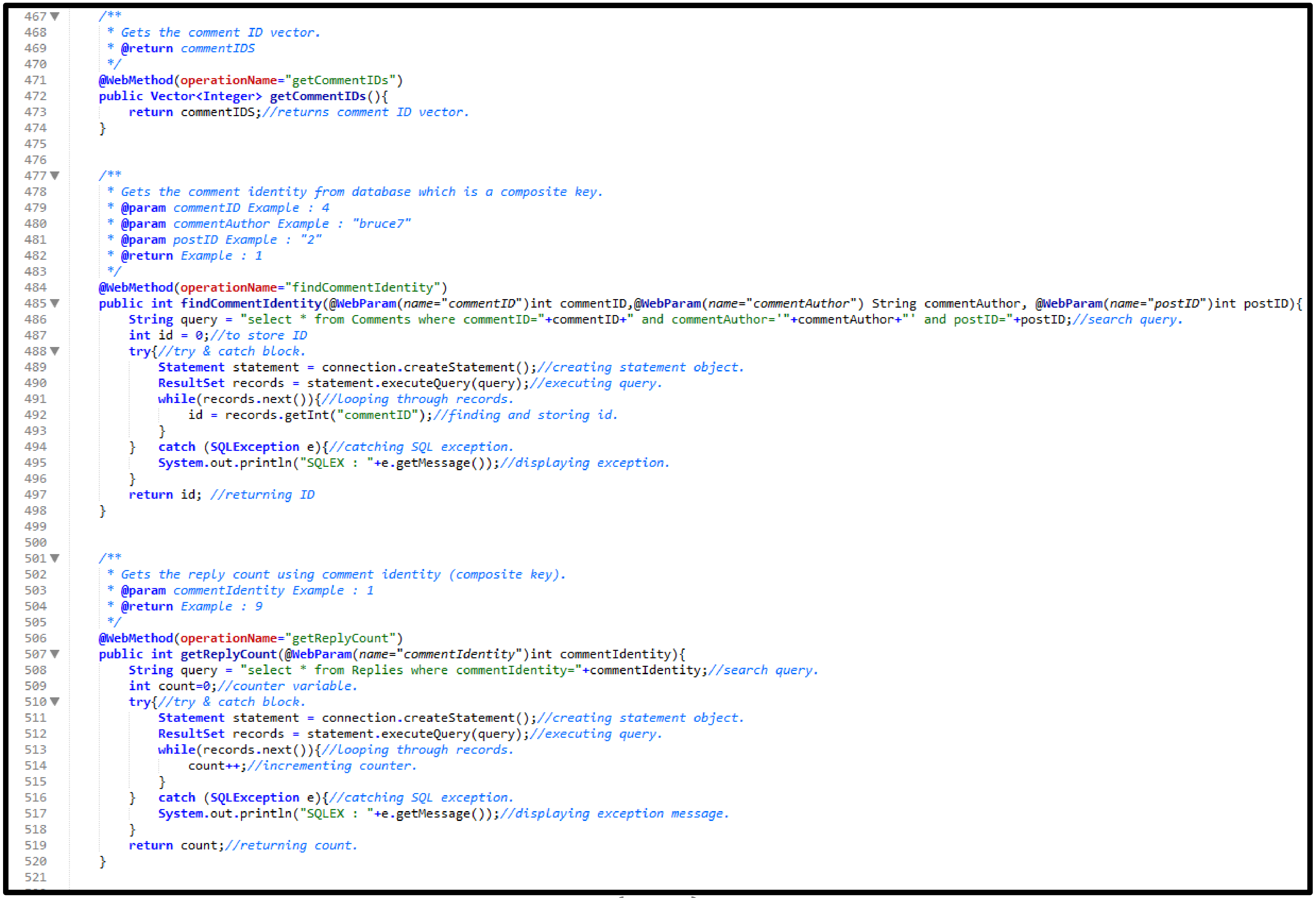
```java
    /**
     * Gets the description of a post using postID.
     * @param postID Example : 2
     * @return Example : "Can you please help me with this?"
     * @throws SQLException if there is a query or connection  invalid.
     */
    @WebMethod(operationName="getPostDescription")
    public String getPostDesciption(@WebParam(name="postID")int postID){
        String description="";//used to store description of post.
        String query = "select postDescription from Posts where postID="+postID;//search query.
        try{//using try & catch block to catch exception.
            Statement statement = connection.createStatement();//creating statement object.
            ResultSet records = statement.executeQuery(query);//executing query.
            while(records.next()){//looping through records.
               description = records.getString("postDescription");//storing description into variable.
            }
        }   catch (SQLException e){//catching SQL Exception
            System.out.println("SQLEX : "+e.getMessage());//displaying exception messagage.
        }
        return description;//returns post description.
    }


    /**
     * Gets the title of the post using the topicName and postID.
     * @param topicName Example : "Apple"
     * @param postID Example : 1
     * @return Example : "What is a good iPhone?"
     * @throws SQLException if there is a query or connection  invalid.
     */
    @WebMethod(operationName="getPostTitle")
    public String getPostTitle(@WebParam(name="topicName")String topicName,@WebParam(name="postID")int postID){
        String query = "select * from Posts where TopicID="+getTopicID(topicName)+" and postID="+postID;//search query.
        String title="";//to store the post title.
        try{//using try & catch block
            Statement statement = connection.createStatement();//creating statement object.
            ResultSet records = statement.executeQuery(query);//executing query.
            while(records.next()){//looping through each record.
                title = records.getString("postSubject");//locating and assigning post title.
            }
        }   catch (SQLException e){//catching SQL exception.
            System.out.println("SQLEX : "+e.getMessage());//displaying exception message.
        }
        return title;//returning postTitle(subject)
    }

```



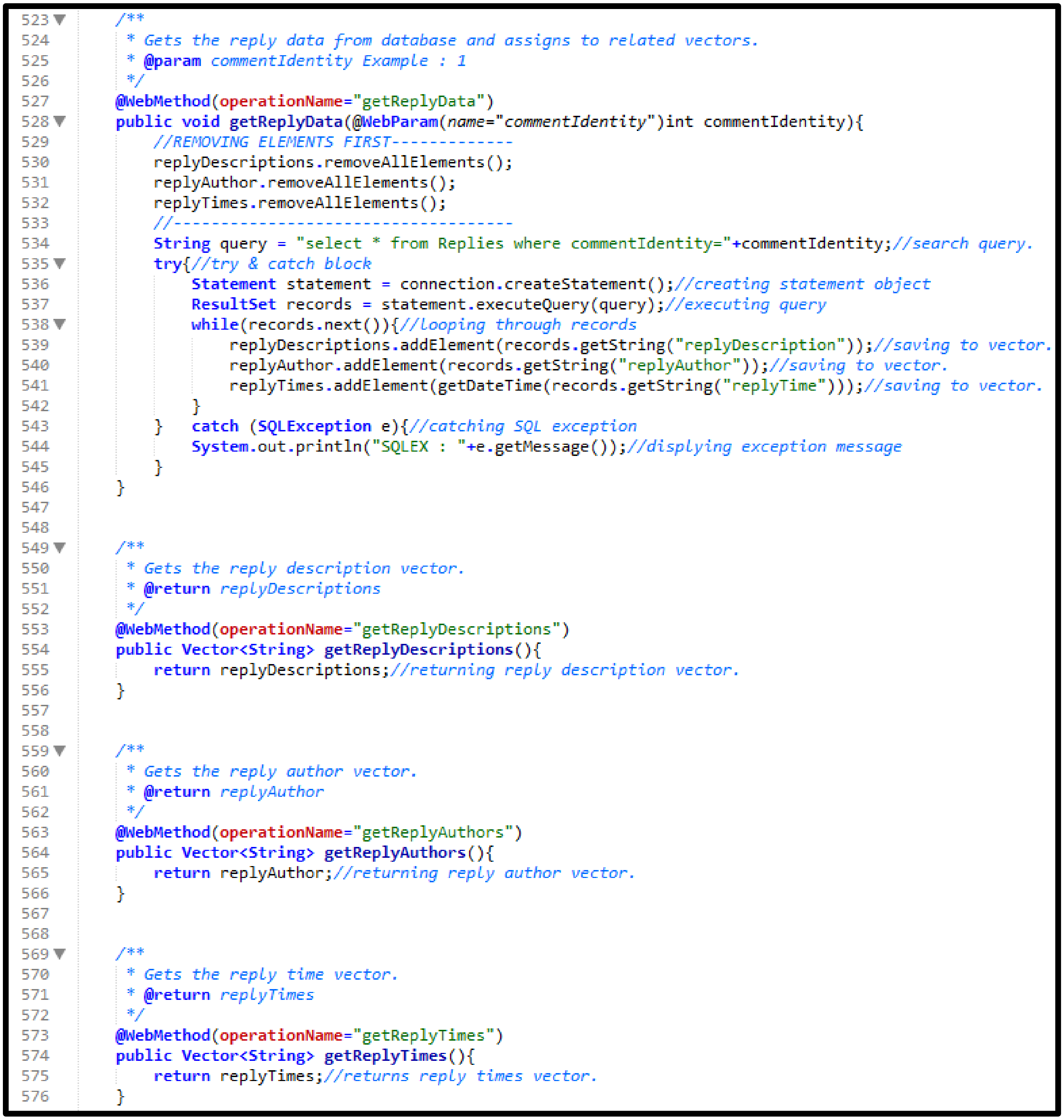
```java
    /**
     * Gets the total count of comments.
     * @return Example : 5
     * @throws SQLException if there is a query or connection  invalid.
     */
    @WebMethod(operationName="getTotalCommentCount")
    public int getTotalCommentCount(){
        String query = "select * from Comments";//search query.
        int count=0;//counter.
        try{//using try & catch block.
            Statement statement = connection.createStatement();//creating statement object.
            ResultSet records = statement.executeQuery(query);//executing query.
            while(records.next()){//looping through records.
                count++;//incrementing counter.
            }
        }   catch (SQLException e){//catching exception.
            System.out.println("SQLEX : "+e.getMessage());//displaying exception message.
        }
        return count;//returning comment count.
    }


    /**
     * Gets the count of comments of a specific post using postID
     * @param postID Example : 2
     * @return Example : 5
     * @throws SQLException if there is a query or connection  invalid.
     */
    @WebMethod(operationName="getCommentCount")
    public int getCommentCount(@WebParam(name="postID")int postID){
        String query = "select * from Comments where postID="+postID;//search query.
        int count=0;//counter.
        try{//try & catch block.
            Statement statement = connection.createStatement();//creating statement object.
            ResultSet records = statement.executeQuery(query);//executing query.
            while(records.next()){//looping through records.
                count++;//incrementing count.
            }
        }   catch (SQLException e){//catching exception.
            System.out.println("SQLEX : "+e.getMessage());//displaying exception message.
        }
        return count;//returning comment count.
    }
```



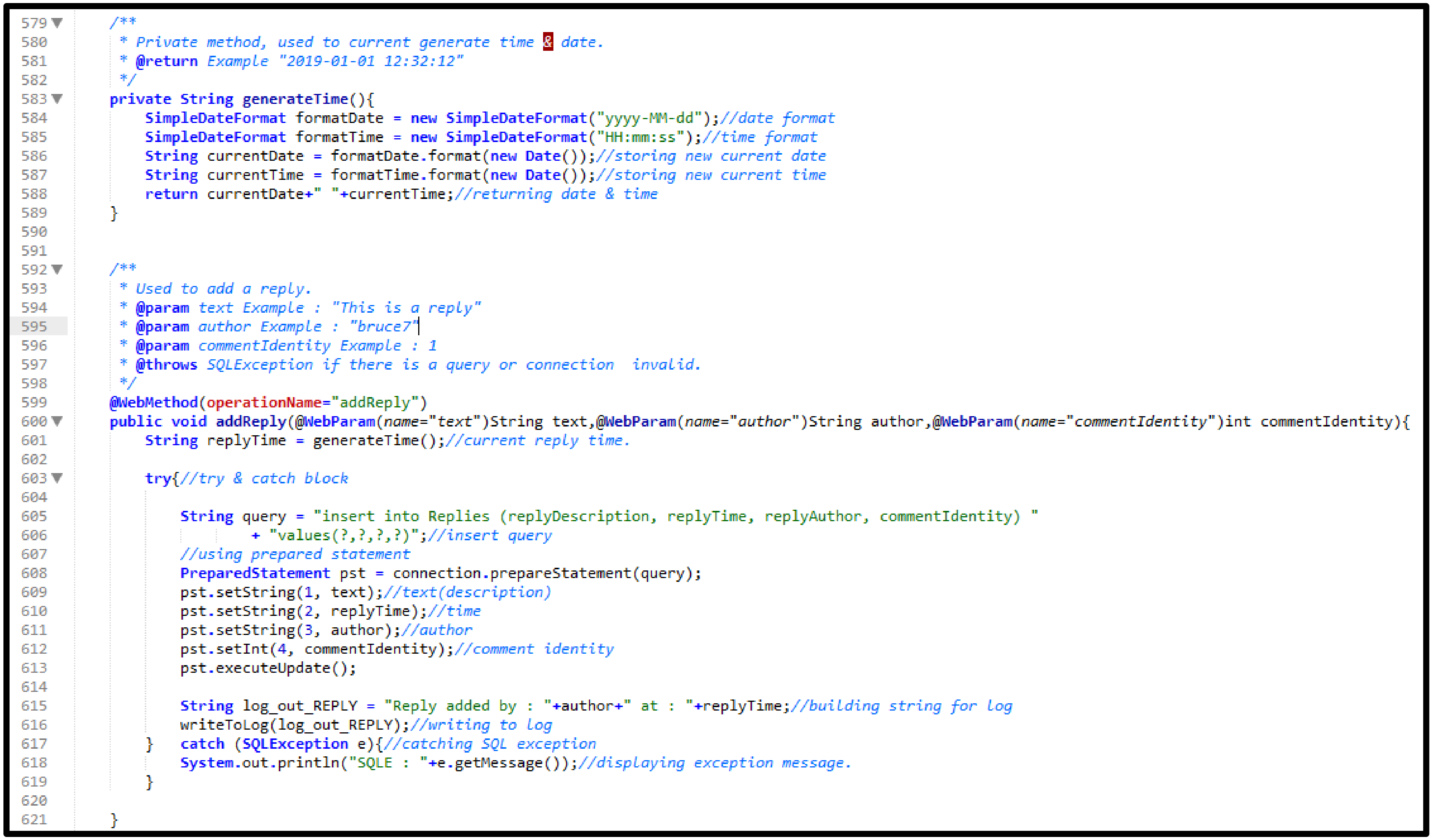
```
    /**
     * Gets the comment data from database and saves to related vectors.
     * @param postID Example : 2
     * @throws SQLException if there is a query or connection  invalid.
     */
    @WebMethod(operationName="getCommentsData")
    public void getCommentsData(@WebParam(name="postID")int postID){
        //REMOVING ELEMENTS FIRST-----------------------
        commentDescriptions.removeAllElements();
        commentAuthors.removeAllElements();
        commentTimes.removeAllElements();
        commentIDS.removeAllElements();
        //-----------------------------------------------

        String query = "select * from Comments where postID="+postID;//search query.

        try{//try & catch block.
            Statement statement = connection.createStatement();//creating statement object.
            ResultSet records = statement.executeQuery(query);//executing query.
            while(records.next()){//looping through records.
                commentDescriptions.addElement(records.getString("commentDesc"));//saving to vector.
                commentAuthors.addElement(records.getString("commentAuthor"));//saving to vector.
                commentTimes.addElement(getDateTime(records.getString("commentTime")));//saving to vector.
                commentIDS.addElement(records.getInt("CommentID"));//saving to vector.
            }
        }   catch (SQLException e){//catching SQL exception.
            System.out.println("SQLEX : "+e.getMessage());//displaying exception message.
        }
    }



    /**
     * Gets the comment description vector.
     * @return commentDescriptions
     */
    @WebMethod(operationName="getCommentDescription")
    public Vector<String> getCommentDescription(){
        return commentDescriptions;//returning comment description vector.
    }


    /**
     * Gets the comment author vector.
     * @return commentAuthors
     */
    @WebMethod(operationName="getCommentAuthors")
    public Vector<String> getCommentAuthors(){
        return commentAuthors;//returns comment author vector.
    }


    /**
     * Gets the comment time vector.
     * @return commentTimes
     */
    @WebMethod(operationName="getCommentTimes")
    public Vector<String> getCommentTimes(){
        return commentTimes;//returns comment times vector.
    }
```



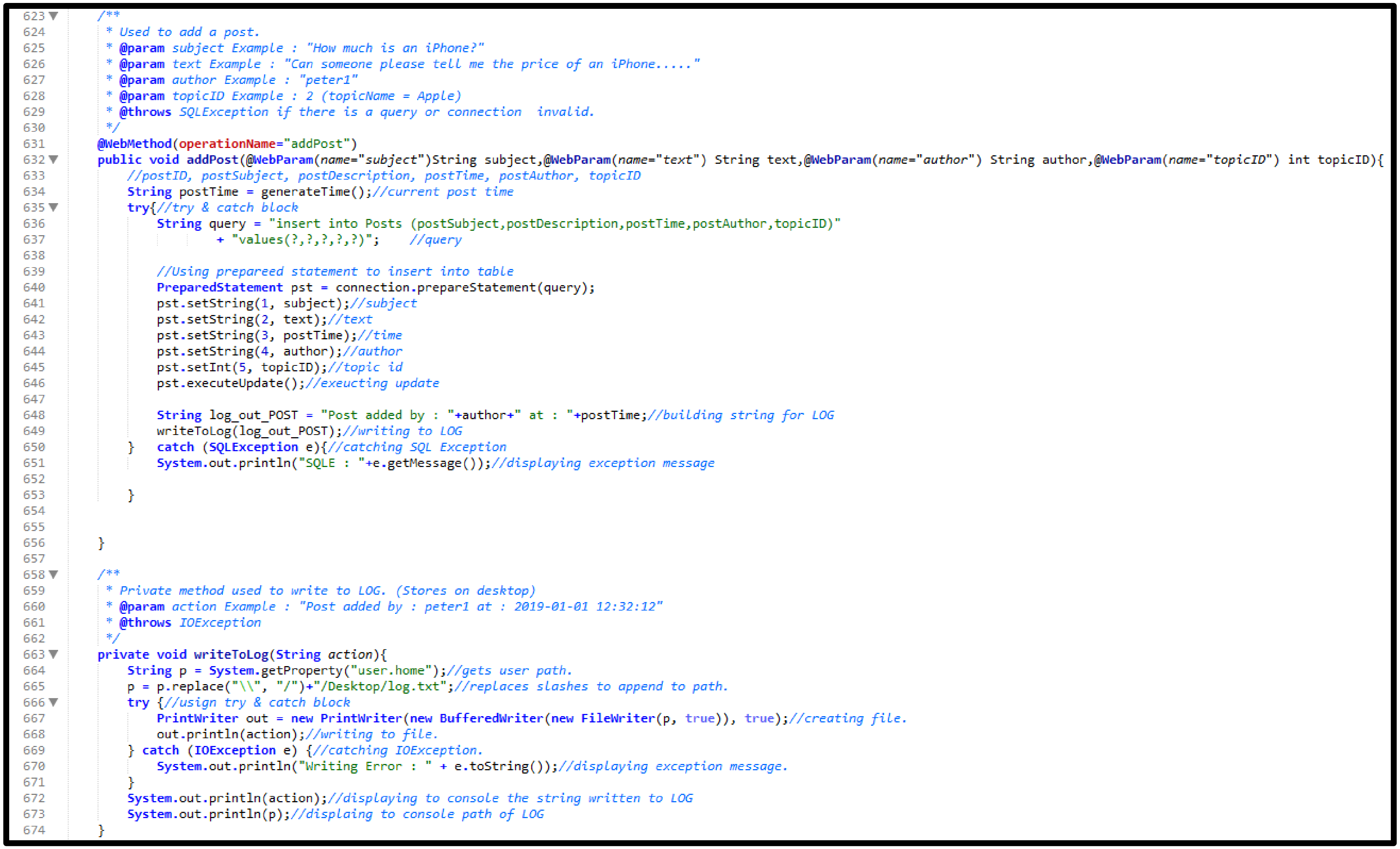
```
/**
 * Gets the comment ID vector.
 * @return commentIDS
 */
@WebMethod(operationName="getCommentIDs")
public Vector<Integer> getCommentIDs(){
    return commentIDS;//returns comment ID vector.
}


/**
 * Gets the comment identity from database which is a composite key.
 * @param commentID Example : 4
 * @param commentAuthor Example : "bruce7"
 * @param postID Example : "2"
 * @return Example : 1
 */
@WebMethod(operationName="findCommentIdentity")
public int findCommentIdentity(@WebParam(name="commentID")int commentID,@WebParam(name="commentAuthor") String commentAuthor, @WebParam(name="postID")int postID){
    String query = "select * from Comments where commentID="+commentID+" and commentAuthor='"+commentAuthor+"' and postID="+postID;//search query.
    int id = 0;//to store ID
    try{//try & catch block.
        Statement statement = connection.createStatement();//creating statement object.
        ResultSet records = statement.executeQuery(query);//executing query.
        while(records.next()){//looping through records.
            id = records.getInt("commentID");//finding and storing id.
        }
    }   catch (SQLException e){//catching SQL exception.
        System.out.println("SQLEX : "+e.getMessage());//displaying exception.
    }
    return id; //returning ID
}


/**
 * Gets the reply count using comment identity (composite key).
 * @param commentIdentity Example : 1
 * @return Example : 9
 */
@WebMethod(operationName="getReplyCount")
public int getReplyCount(@WebParam(name="commentIdentity")int commentIdentity){
    String query = "select * from Replies where commentIdentity="+commentIdentity;//search query.
    int count=0;//counter variable.
    try{//try & catch block.
        Statement statement = connection.createStatement();//creating statement object.
        ResultSet records = statement.executeQuery(query);//executing query.
        while(records.next()){//looping through records.
            count++;//incrementing counter.
        }
    }   catch (SQLException e){//catching SQL exception.
        System.out.println("SQLEX : "+e.getMessage());//displaying exception message.
    }
    return count;//returning count.
}

```



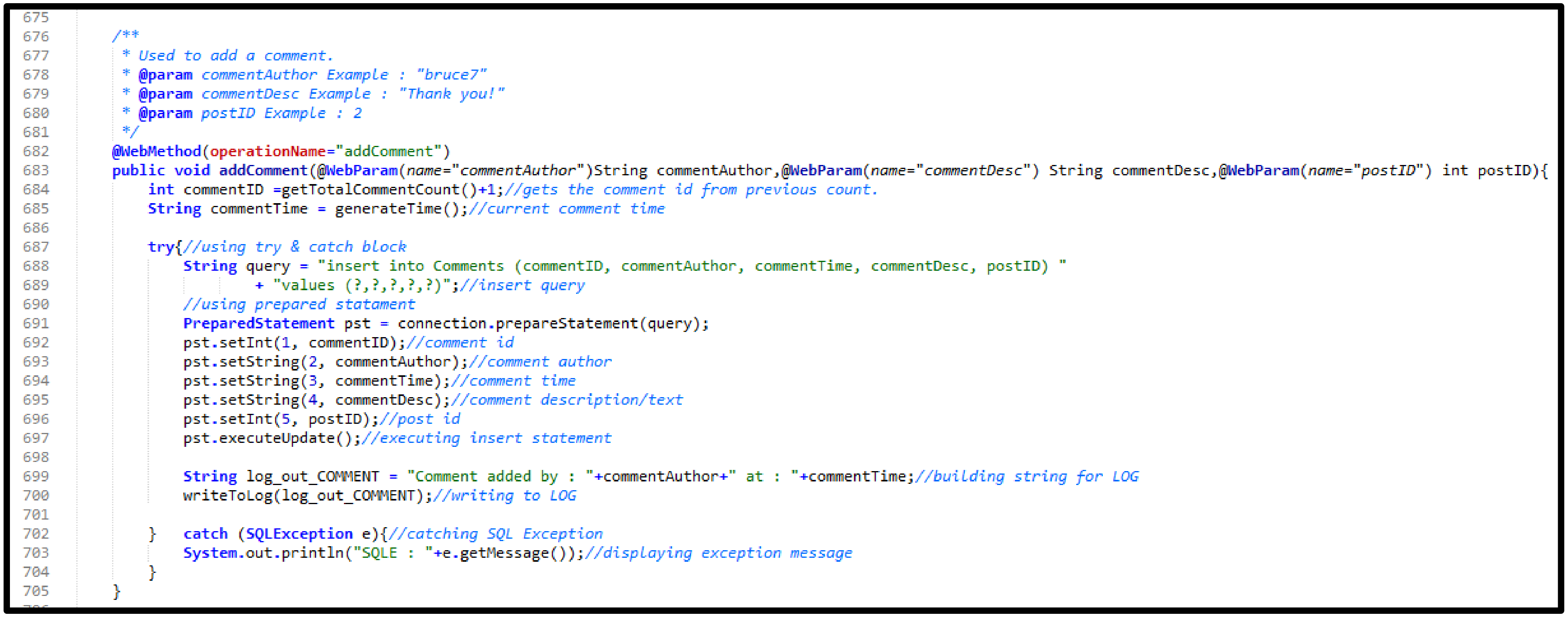
```
    /**
     * Gets the reply data from database and assigns to related vectors.
     * @param commentIdentity Example : 1
     */
    @WebMethod(operationName="getReplyData")
    public void getReplyData(@WebParam(name="commentIdentity")int commentIdentity){
        //REMOVING ELEMENTS FIRST-------------
        replyDescriptions.removeAllElements();
        replyAuthor.removeAllElements();
        replyTimes.removeAllElements();
        //-------------------------------------
        String query = "select * from Replies where commentIdentity="+commentIdentity;//search query.
        try{//try & catch block
            Statement statement = connection.createStatement();//creating statement object
            ResultSet records = statement.executeQuery(query);//executing query
            while(records.next()){//looping through records
                replyDescriptions.addElement(records.getString("replyDescription"));//saving to vector.
                replyAuthor.addElement(records.getString("replyAuthor"));//saving to vector.
                replyTimes.addElement(getDateTime(records.getString("replyTime")));//saving to vector.
            }
        }   catch (SQLException e){//catching SQL exception
            System.out.println("SQLEX : "+e.getMessage());//displying exception message
        }
    }


    /**
     * Gets the reply description vector.
     * @return replyDescriptions
     */
    @WebMethod(operationName="getReplyDescriptions")
    public Vector<String> getReplyDescriptions(){
        return replyDescriptions;//returning reply description vector.
    }


    /**
     * Gets the reply author vector.
     * @return replyAuthor
     */
    @WebMethod(operationName="getReplyAuthors")
    public Vector<String> getReplyAuthors(){
        return replyAuthor;//returning reply author vector.
    }


    /**
     * Gets the reply time vector.
     * @return replyTimes
     */
    @WebMethod(operationName="getReplyTimes")
    public Vector<String> getReplyTimes(){
        return replyTimes;//returns reply times vector.
    }
```



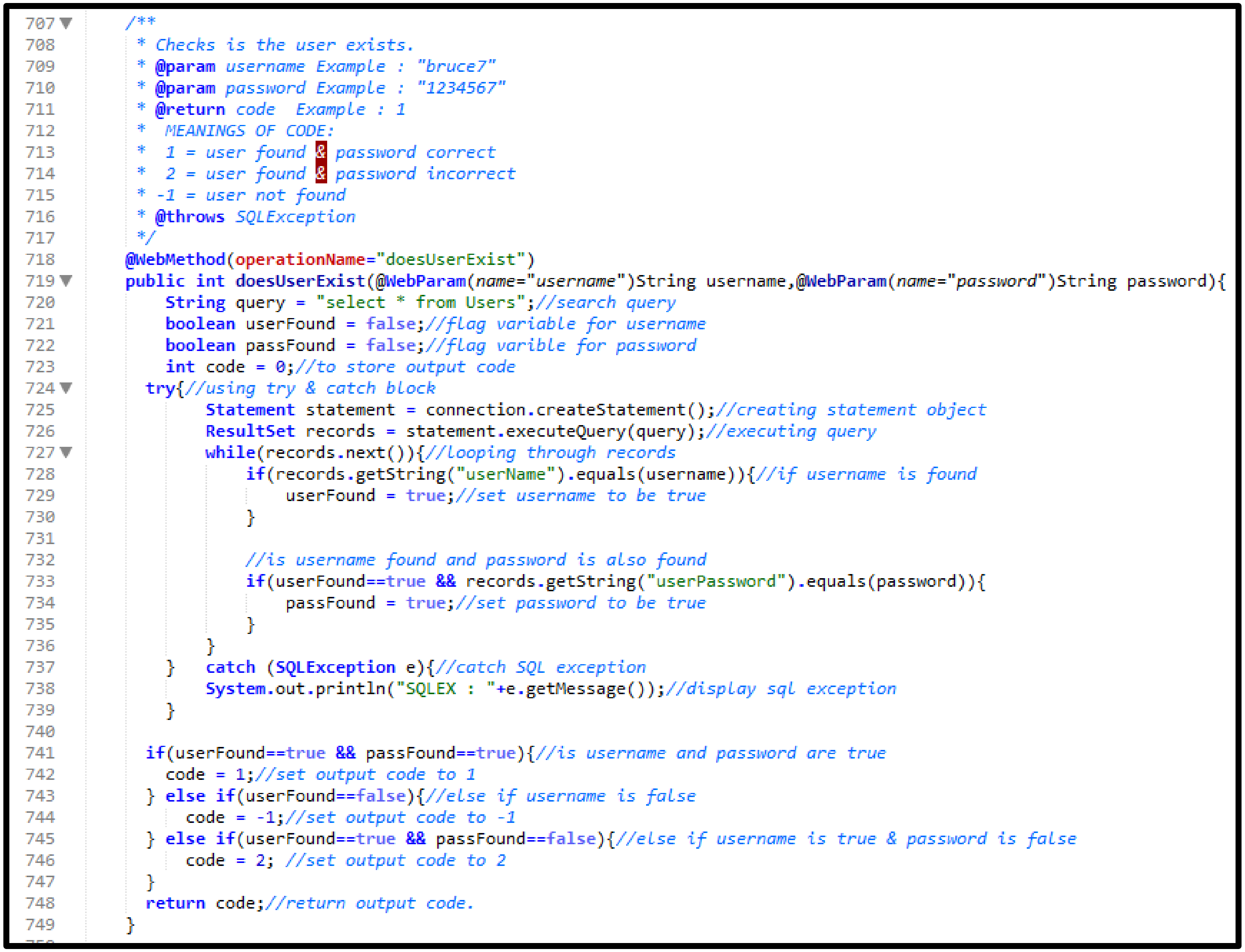
```java
/**
 * Private method, used to current generate time & date.
 * @return Example "2019-01-01 12:32:12"
 */
private String generateTime(){
    SimpleDateFormat formatDate = new SimpleDateFormat("yyyy-MM-dd");//date format
    SimpleDateFormat formatTime = new SimpleDateFormat("HH:mm:ss");//time format
    String currentDate = formatDate.format(new Date());//storing new current date
    String currentTime = formatTime.format(new Date());//storing new current time
    return currentDate+" "+currentTime;//returning date & time
}


/**
 * Used to add a reply.
 * @param text Example : "This is a reply"
 * @param author Example : "bruce7"
 * @param commentIdentity Example : 1
 * @throws SQLException if there is a query or connection  invalid.
 */
@WebMethod(operationName="addReply")
public void addReply(@WebParam(name="text")String text,@WebParam(name="author")String author,@WebParam(name="commentIdentity")int commentIdentity){
    String replyTime = generateTime();//current reply time.

    try{//try & catch block

        String query = "insert into Replies (replyDescription, replyTime, replyAuthor, commentIdentity) "
                + "values(?,?,?,?)";//insert query
        //using prepared statement
        PreparedStatement pst = connection.prepareStatement(query);
        pst.setString(1, text);//text(description)
        pst.setString(2, replyTime);//time
        pst.setString(3, author);//author
        pst.setInt(4, commentIdentity);//comment identity
        pst.executeUpdate();

        String log_out_REPLY = "Reply added by : "+author+" at : "+replyTime;//building string for log
        writeToLog(log_out_REPLY);//writing to log
    } catch (SQLException e){//catching SQL exception
        System.out.println("SQLE : "+e.getMessage());//displaying exception message.
    }

}
```



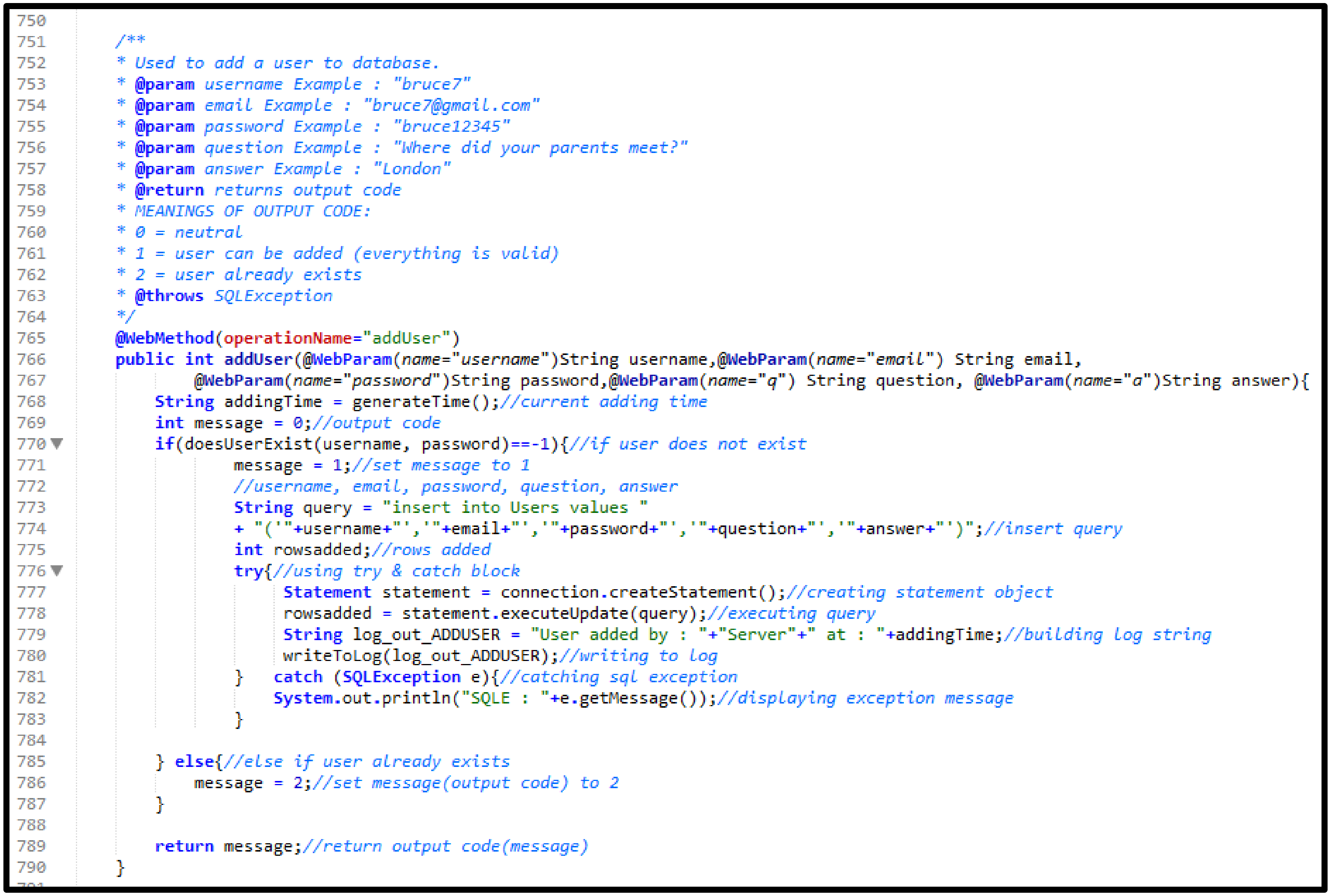
```java
    /**
     * Used to add a post.
     * @param subject Example : "How much is an iPhone?"
     * @param text Example : "Can someone please tell me the price of an iPhone....."
     * @param author Example : "peter1"
     * @param topicID Example : 2 (topicName = Apple)
     * @throws SQLException if there is a query or connection  invalid.
     */
    @WebMethod(operationName="addPost")
    public void addPost(@WebParam(name="subject")String subject,@WebParam(name="text") String text,@WebParam(name="author") String author,@WebParam(name="topicID") int topicID){
        //postID, postSubject, postDescription, postTime, postAuthor, topicID
        String postTime = generateTime();//current post time
        try{//try & catch block
            String query = "insert into Posts (postSubject,postDescription,postTime,postAuthor,topicID)"
                    + "values(?,?,?,?,?)";    //query

            //Using prepareed statement to insert into table
            PreparedStatement pst = connection.prepareStatement(query);
            pst.setString(1, subject);//subject
            pst.setString(2, text);//text
            pst.setString(3, postTime);//time
            pst.setString(4, author);//author
            pst.setInt(5, topicID);//topic id
            pst.executeUpdate();//exeucting update

            String log_out_POST = "Post added by : "+author+" at : "+postTime;//building string for LOG
            writeToLog(log_out_POST);//writing to LOG
        }  catch (SQLException e){//catching SQL Exception
            System.out.println("SQLE : "+e.getMessage());//displaying exception message

        }


    }

    /**
     * Private method used to write to LOG. (Stores on desktop)
     * @param action Example : "Post added by : peter1 at : 2019-01-01 12:32:12"
     * @throws IOException
     */
    private void writeToLog(String action){
        String p = System.getProperty("user.home");//gets user path.
        p = p.replace("\\", "/")+"/Desktop/log.txt";//replaces slashes to append to path.
        try {//usign try & catch block
            PrintWriter out = new PrintWriter(new BufferedWriter(new FileWriter(p, true)), true);//creating file.
            out.println(action);//writing to file.
        } catch (IOException e) {//catching IOException.
            System.out.println("Writing Error : " + e.toString());//displaying exception message.
        }
        System.out.println(action);//displaying to console the string written to LOG
        System.out.println(p);//displaing to console path of LOG
    }
```



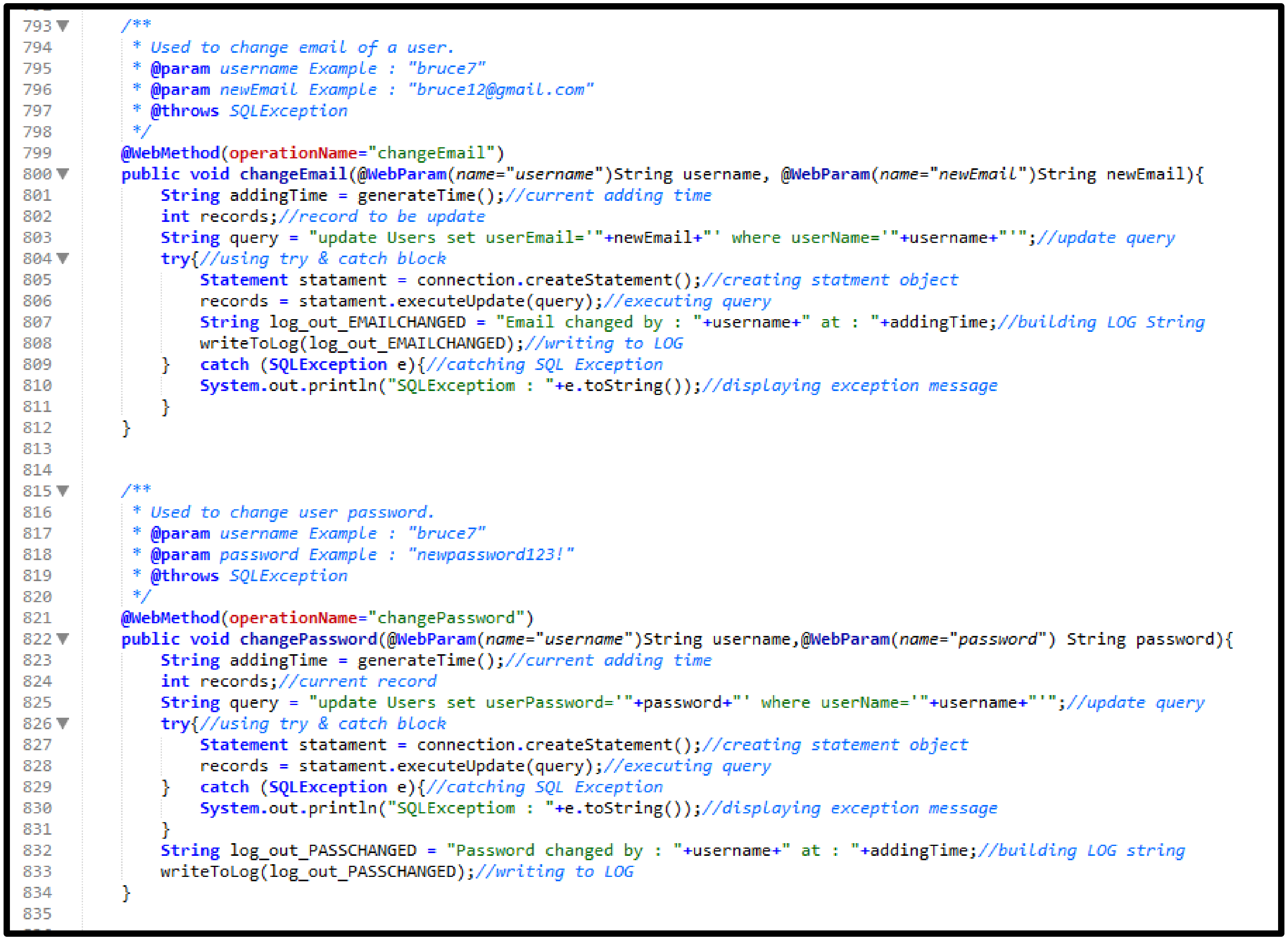
```java

    /**
     * Used to add a comment.
     * @param commentAuthor Example : "bruce7"
     * @param commentDesc Example : "Thank you!"
     * @param postID Example : 2
     */
    @WebMethod(operationName="addComment")
    public void addComment(@WebParam(name="commentAuthor")String commentAuthor,@WebParam(name="commentDesc") String commentDesc,@WebParam(name="postID") int postID){
        int commentID =getTotalCommentCount()+1;//gets the comment id from previous count.
        String commentTime = generateTime();//current comment time

        try{//using try & catch block
            String query = "insert into Comments (commentID, commentAuthor, commentTime, commentDesc, postID) "
                    + "values (?,?,?,?,?)";//insert query
            //using prepared statament
            PreparedStatement pst = connection.prepareStatement(query);
            pst.setInt(1, commentID);//comment id
            pst.setString(2, commentAuthor);//comment author
            pst.setString(3, commentTime);//comment time
            pst.setString(4, commentDesc);//comment description/text
            pst.setInt(5, postID);//post id
            pst.executeUpdate();//executing insert statement

            String log_out_COMMENT = "Comment added by : "+commentAuthor+" at : "+commentTime;//building string for LOG
            writeToLog(log_out_COMMENT);//writing to LOG

        }   catch (SQLException e){//catching SQL Exception
            System.out.println("SQLE : "+e.getMessage());//displaying exception message
        }
    }
```



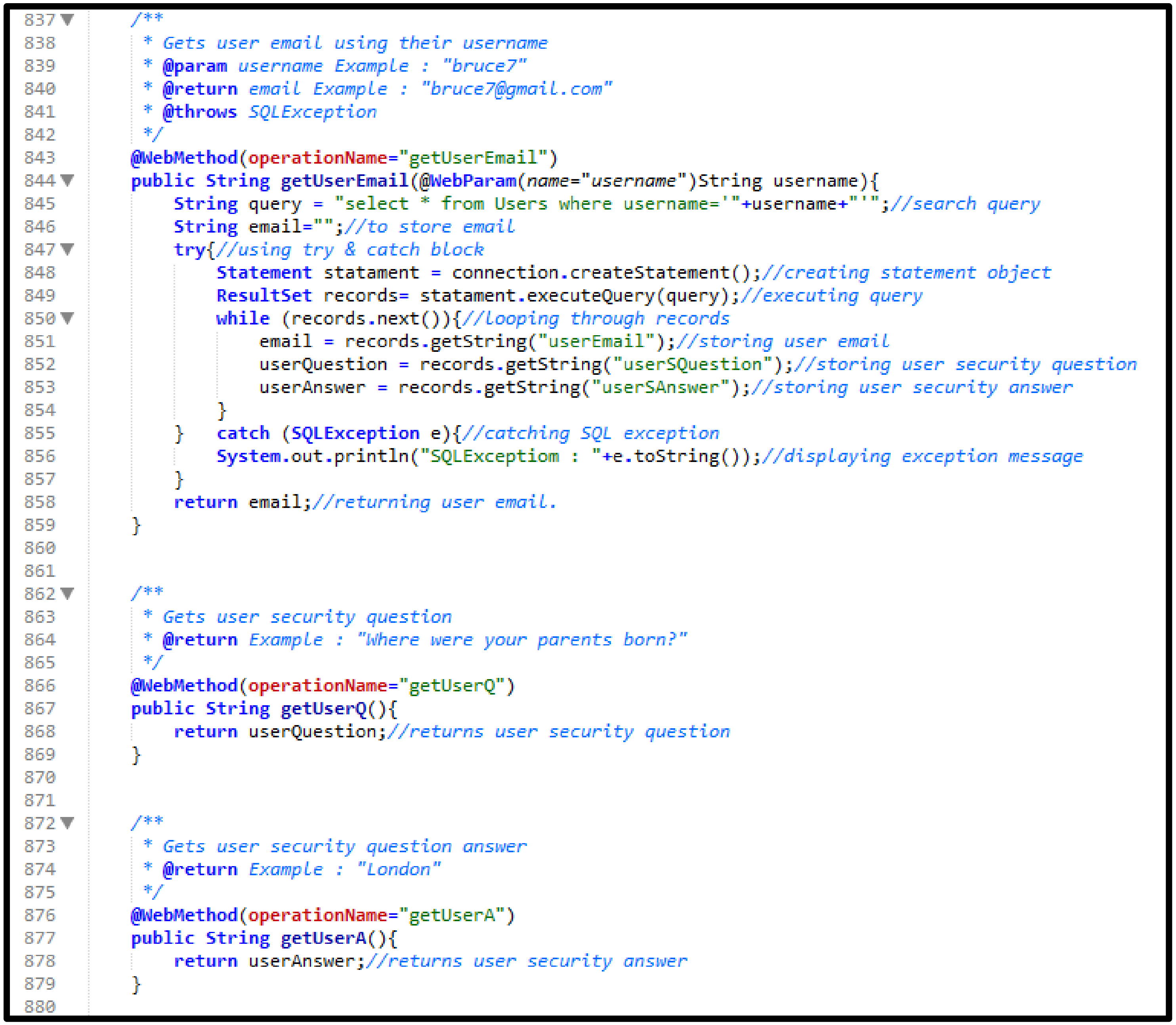
```
/**
 * Checks is the user exists.
 * @param username Example : "bruce7"
 * @param password Example : "1234567"
 * @return code  Example : 1
 *  MEANINGS OF CODE:
 *  1 = user found & password correct
 *  2 = user found & password incorrect
 * -1 = user not found
 * @throws SQLException
 */
@WebMethod(operationName="doesUserExist")
public int doesUserExist(@WebParam(name="username")String username,@WebParam(name="password")String password){
  String query = "select * from Users";//search query
  boolean userFound = false;//flag variable for username
  boolean passFound = false;//flag varible for password
  int code = 0;//to store output code
 try{//using try & catch block
    Statement statement = connection.createStatement();//creating statement object
    ResultSet records = statement.executeQuery(query);//executing query
    while(records.next()){//looping through records
      if(records.getString("userName").equals(username)){//if username is found
        userFound = true;//set username to be true
      }

      //is username found and password is also found
      if(userFound==true && records.getString("userPassword").equals(password)){
        passFound = true;//set password to be true
      }
    }
  }  catch (SQLException e){//catch SQL exception
    System.out.println("SQLEX : "+e.getMessage());//display sql exception
  }

 if(userFound==true && passFound==true){//is username and password are true
  code = 1;//set output code to 1
 } else if(userFound==false){//else if username is false
   code = -1;//set output code to -1
 } else if(userFound==true && passFound==false){//else if username is true & password is false
   code = 2; //set output code to 2
 }
 return code;//return output code.
}
```

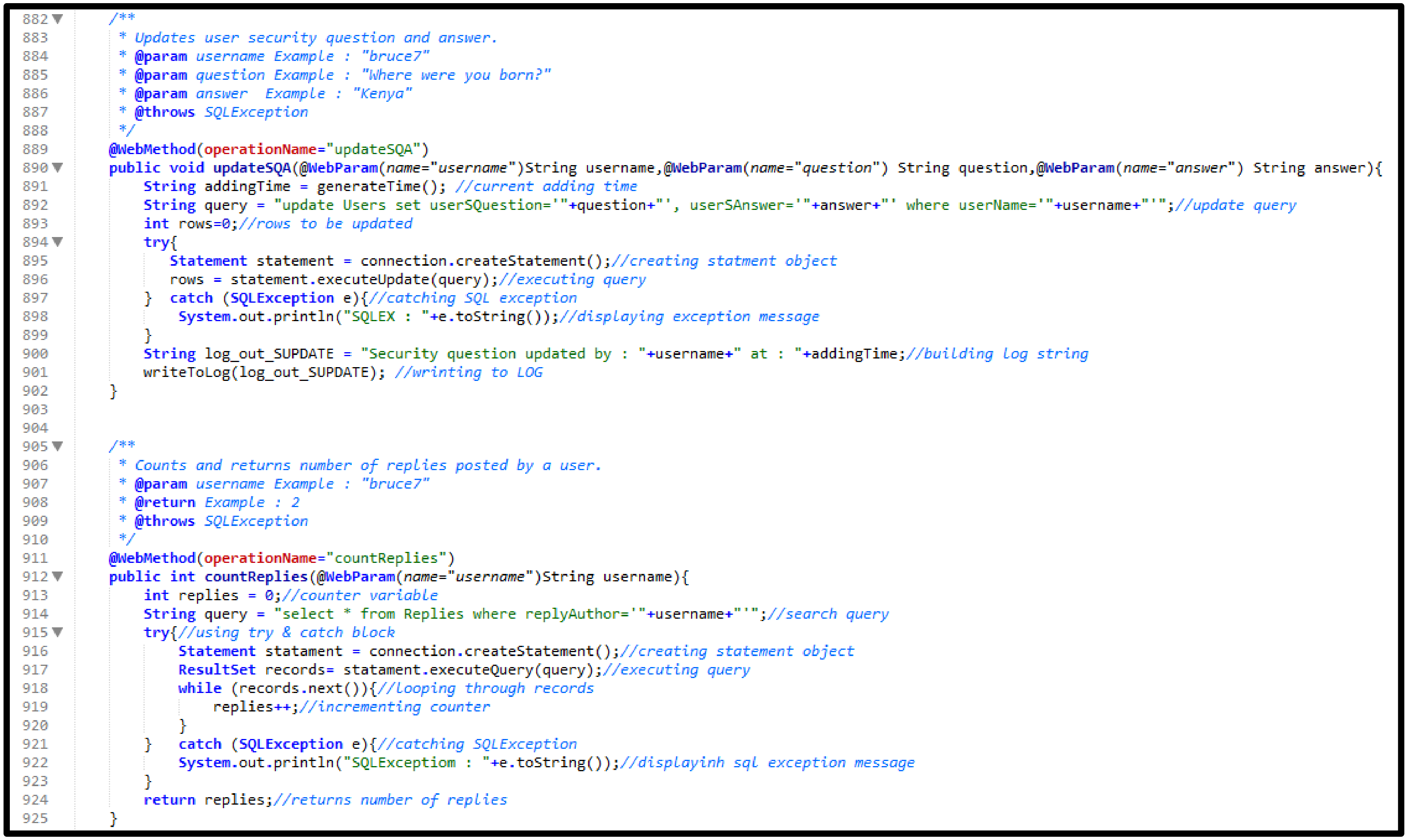


```
    /**
    * Used to add a user to database.
    * @param username Example : "bruce7"
    * @param email Example : "bruce7@gmail.com"
    * @param password Example : "bruce12345"
    * @param question Example : "Where did your parents meet?"
    * @param answer Example : "London"
    * @return returns output code
    * MEANINGS OF OUTPUT CODE:
    * 0 = neutral
    * 1 = user can be added (everything is valid)
    * 2 = user already exists
    * @throws SQLException
    */
    @WebMethod(operationName="addUser")
    public int addUser(@WebParam(name="username")String username,@WebParam(name="email") String email,
            @WebParam(name="password")String password,@WebParam(name="q") String question, @WebParam(name="a")String answer){
        String addingTime = generateTime();//current adding time
        int message = 0;//output code
        if(doesUserExist(username, password)==-1){//if user does not exist
                message = 1;//set message to 1
                //username, email, password, question, answer
                String query = "insert into Users values "
                + "('"+username+"','"+email+"','"+password+"','"+question+"','"+answer+"')";//insert query
                int rowsadded;//rows added
                try{//using try & catch block
                     Statement statement = connection.createStatement();//creating statement object
                     rowsadded = statement.executeUpdate(query);//executing query
                     String log_out_ADDUSER = "User added by : "+"Server"+" at : "+addingTime;//building log string
                     writeToLog(log_out_ADDUSER);//writing to log
                }   catch (SQLException e){//catching sql exception
                    System.out.println("SQLE : "+e.getMessage());//displaying exception message
                }

        } else{//else if user already exists
            message = 2;//set message(output code) to 2
        }

        return message;//return output code(message)
    }
```

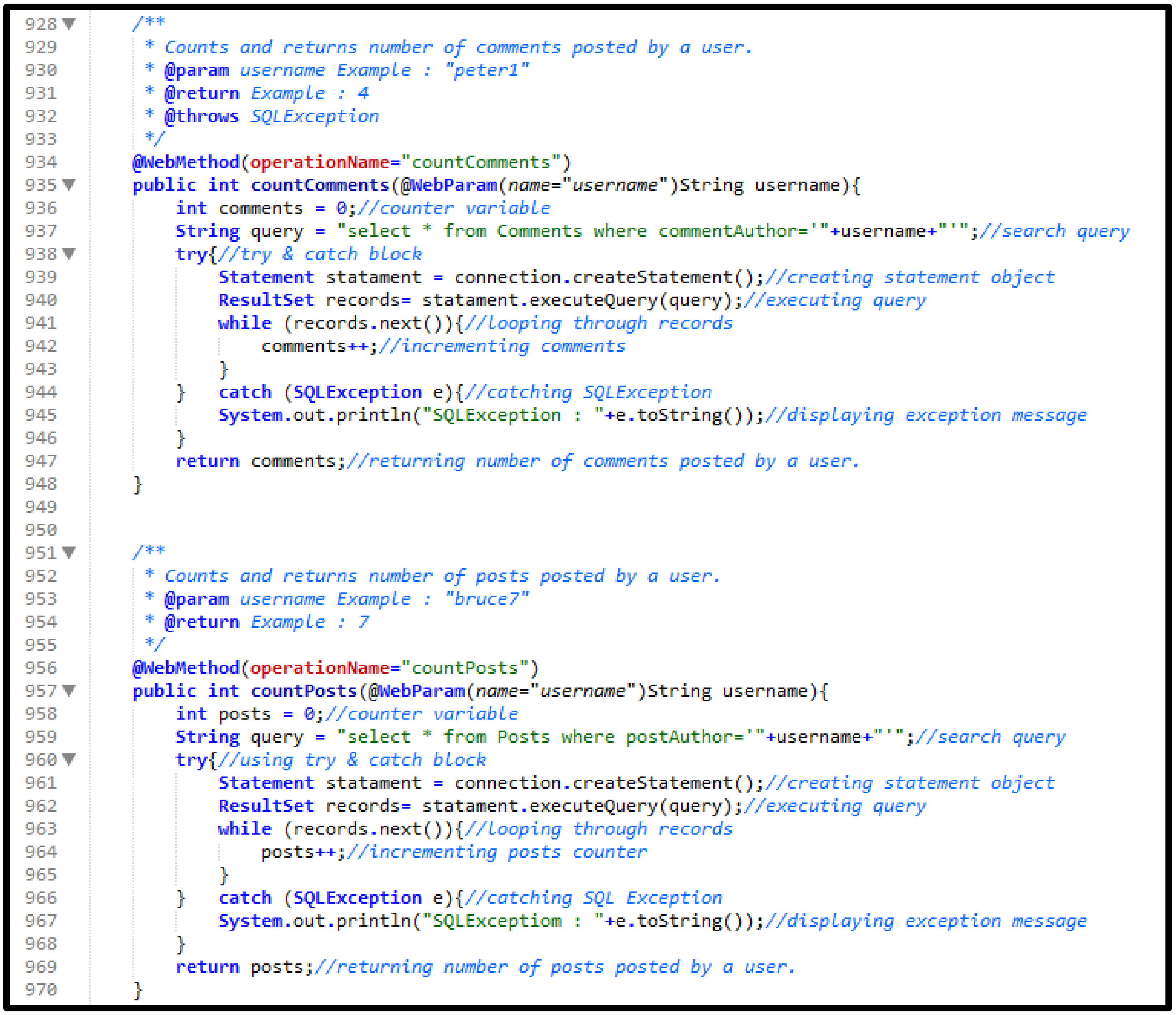


```
    /**
     * Used to change email of a user.
     * @param username Example : "bruce7"
     * @param newEmail Example : "bruce12@gmail.com"
     * @throws SQLException
     */
    @WebMethod(operationName="changeEmail")
    public void changeEmail(@WebParam(name="username")String username, @WebParam(name="newEmail")String newEmail){
        String addingTime = generateTime();//current adding time
        int records;//record to be update
        String query = "update Users set userEmail='"+newEmail+"' where userName='"+username+"'";//update query
        try{//using try & catch block
            Statement statament = connection.createStatement();//creating statment object
            records = statament.executeUpdate(query);//executing query
            String log_out_EMAILCHANGED = "Email changed by : "+username+" at : "+addingTime;//building LOG String
            writeToLog(log_out_EMAILCHANGED);//writing to LOG
        }   catch (SQLException e){//catching SQL Exception
            System.out.println("SQLExceptiom : "+e.toString());//displaying exception message
        }
    }


    /**
     * Used to change user password.
     * @param username Example : "bruce7"
     * @param password Example : "newpassword123!"
     * @throws SQLException
     */
    @WebMethod(operationName="changePassword")
    public void changePassword(@WebParam(name="username")String username,@WebParam(name="password") String password){
        String addingTime = generateTime();//current adding time
        int records;//current record
        String query = "update Users set userPassword='"+password+"' where userName='"+username+"'";//update query
        try{//using try & catch block
            Statement statament = connection.createStatement();//creating statement object
            records = statament.executeUpdate(query);//executing query
        }   catch (SQLException e){//catching SQL Exception
            System.out.println("SQLExceptiom : "+e.toString());//displaying exception message
        }
        String log_out_PASSCHANGED = "Password changed by : "+username+" at : "+addingTime;//building LOG string
        writeToLog(log_out_PASSCHANGED);//writing to LOG
    }
```

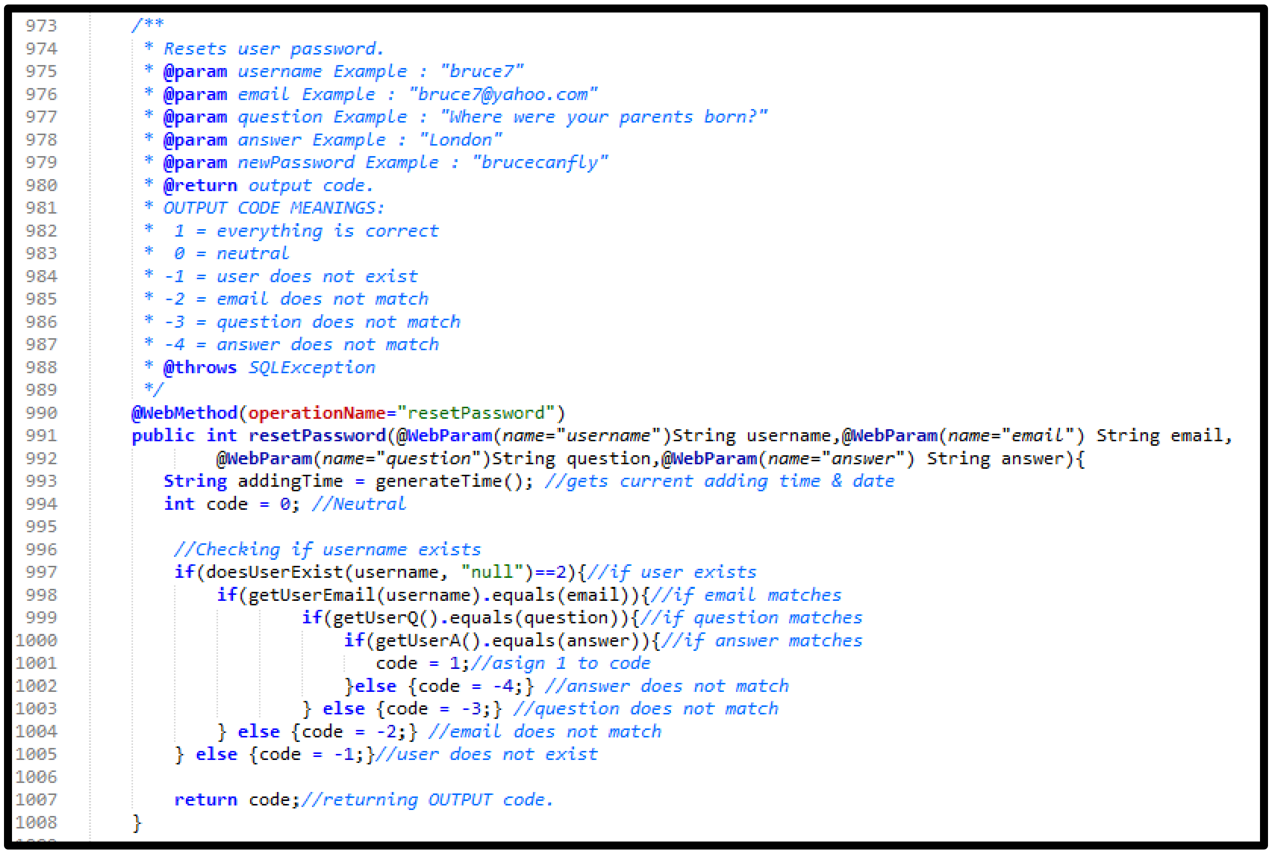

```java
    /**
     * Gets user email using their username
     * @param username Example : "bruce7"
     * @return email Example : "bruce7@gmail.com"
     * @throws SQLException
     */
    @WebMethod(operationName="getUserEmail")
    public String getUserEmail(@WebParam(name="username")String username){
        String query = "select * from Users where username='"+username+"'";//search query
        String email="";//to store email
        try{//using try & catch block
            Statement statament = connection.createStatement();//creating statement object
            ResultSet records= statament.executeQuery(query);//executing query
            while (records.next()){//looping through records
                email = records.getString("userEmail");//storing user email
                userQuestion = records.getString("userSQuestion");//storing user security question
                userAnswer = records.getString("userSAnswer");//storing user security answer
            }
        }   catch (SQLException e){//catching SQL exception
            System.out.println("SQLExceptiom : "+e.toString());//displaying exception message
        }
        return email;//returning user email.
    }


    /**
     * Gets user security question
     * @return Example : "Where were your parents born?"
     */
    @WebMethod(operationName="getUserQ")
    public String getUserQ(){
        return userQuestion;//returns user security question
    }


    /**
     * Gets user security question answer
     * @return Example : "London"
     */
    @WebMethod(operationName="getUserA")
    public String getUserA(){
        return userAnswer;//returns user security answer
    }

```



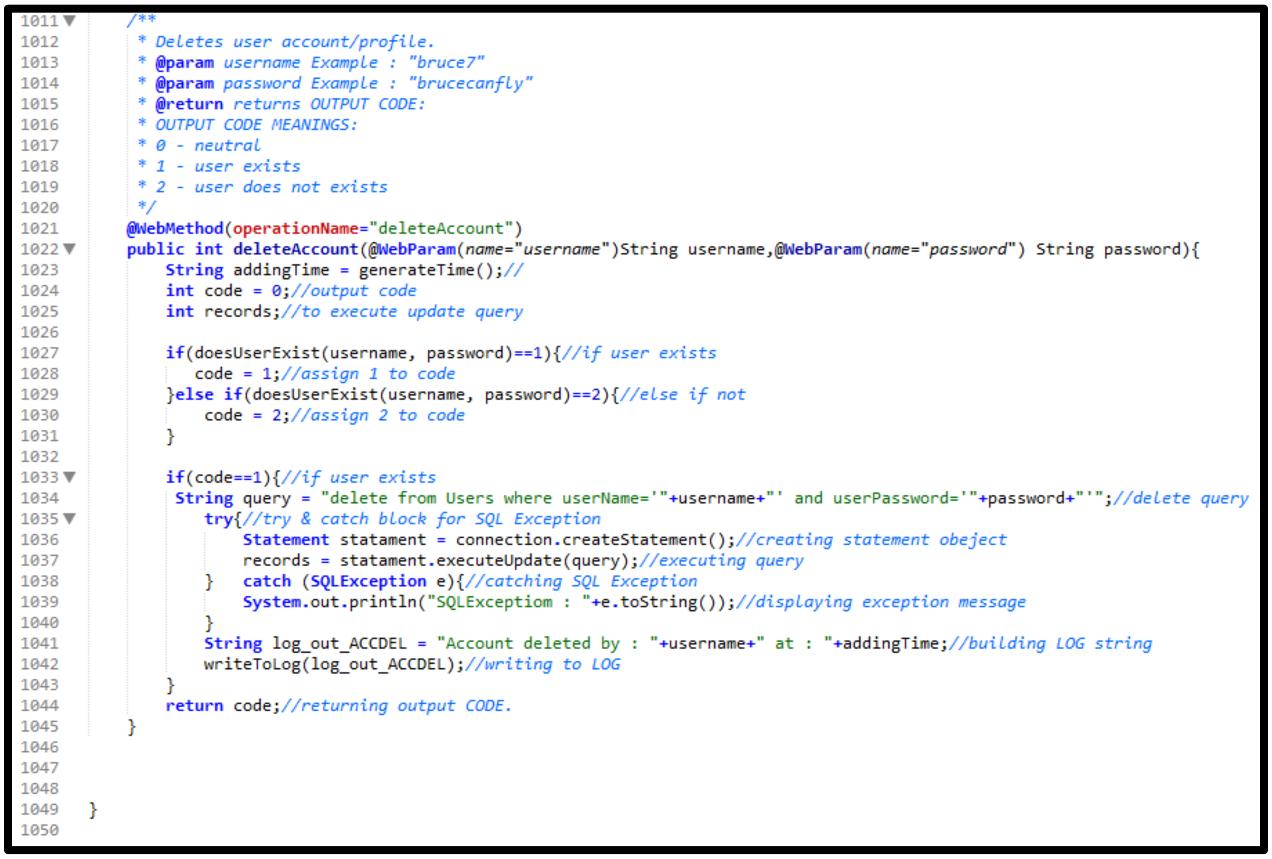
```java
    /**
     * Updates user security question and answer.
     * @param username Example : "bruce7"
     * @param question Example : "Where were you born?"
     * @param answer  Example : "Kenya"
     * @throws SQLException
     */
    @WebMethod(operationName="updateSQA")
    public void updateSQA(@WebParam(name="username")String username,@WebParam(name="question") String question,@WebParam(name="answer") String answer){
        String addingTime = generateTime(); //current adding time
        String query = "update Users set userSQuestion='"+question+"', userSAnswer='"+answer+"' where userName='"+username+"'";//update query
        int rows=0;//rows to be updated
        try{
           Statement statement = connection.createStatement();//creating statment object
           rows = statement.executeUpdate(query);//executing query
        }  catch (SQLException e){//catching SQL exception
            System.out.println("SQLEX : "+e.toString());//displaying exception message
        }
        String log_out_SUPDATE = "Security question updated by : "+username+" at : "+addingTime;//building log string
        writeToLog(log_out_SUPDATE); //wrinting to LOG
    }


    /**
     * Counts and returns number of replies posted by a user.
     * @param username Example : "bruce7"
     * @return Example : 2
     * @throws SQLException
     */
    @WebMethod(operationName="countReplies")
    public int countReplies(@WebParam(name="username")String username){
        int replies = 0;//counter variable
        String query = "select * from Replies where replyAuthor='"+username+"'";//search query
        try{//using try & catch block
            Statement statament = connection.createStatement();//creating statement object
            ResultSet records= statament.executeQuery(query);//executing query
            while (records.next()){//looping through records
                replies++;//incrementing counter
            }
        }   catch (SQLException e){//catching SQLException
            System.out.println("SQLExceptiom : "+e.toString());//displayinh sql exception message
        }
        return replies;//returns number of replies
    }
```



```
    /**
     * Counts and returns number of comments posted by a user.
     * @param username Example : "peter1"
     * @return Example : 4
     * @throws SQLException
     */
    @WebMethod(operationName="countComments")
    public int countComments(@WebParam(name="username")String username){
        int comments = 0;//counter variable
        String query = "select * from Comments where commentAuthor='"+username+"'";//search query
        try{//try & catch block
            Statement statament = connection.createStatement();//creating statement object
            ResultSet records= statament.executeQuery(query);//executing query
            while (records.next()){//looping through records
                comments++;//incrementing comments
            }
        }   catch (SQLException e){//catching SQLException
            System.out.println("SQLException : "+e.toString());//displaying exception message
        }
        return comments;//returning number of comments posted by a user.
    }


    /**
     * Counts and returns number of posts posted by a user.
     * @param username Example : "bruce7"
     * @return Example : 7
     */
    @WebMethod(operationName="countPosts")
    public int countPosts(@WebParam(name="username")String username){
        int posts = 0;//counter variable
        String query = "select * from Posts where postAuthor='"+username+"'";//search query
        try{//using try & catch block
            Statement statament = connection.createStatement();//creating statement object
            ResultSet records= statament.executeQuery(query);//executing query
            while (records.next()){//looping through records
                posts++;//incrementing posts counter
            }
        }   catch (SQLException e){//catching SQL Exception
            System.out.println("SQLExceptiom : "+e.toString());//displaying exception message
        }
        return posts;//returning number of posts posted by a user.
    }
```



```java
    /**
     * Resets user password.
     * @param username Example : "bruce7"
     * @param email Example : "bruce7@yahoo.com"
     * @param question Example : "Where were your parents born?"
     * @param answer Example : "London"
     * @param newPassword Example : "brucecanfly"
     * @return output code.
     * OUTPUT CODE MEANINGS:
     *  1 = everything is correct
     *  0 = neutral
     * -1 = user does not exist
     * -2 = email does not match
     * -3 = question does not match
     * -4 = answer does not match
     * @throws SQLException
     */
    @WebMethod(operationName="resetPassword")
    public int resetPassword(@WebParam(name="username")String username,@WebParam(name="email") String email,
            @WebParam(name="question")String question,@WebParam(name="answer") String answer){
        String addingTime = generateTime(); //gets current adding time & date
        int code = 0; //Neutral

         //Checking if username exists
         if(doesUserExist(username, "null")==2){//if user exists
             if(getUserEmail(username).equals(email)){//if email matches
                     if(getUserQ().equals(question)){//if question matches
                         if(getUserA().equals(answer)){//if answer matches
                            code = 1;//asign 1 to code
                         }else {code = -4;} //answer does not match
                     } else {code = -3;} //question does not match
             } else {code = -2;} //email does not match
         } else {code = -1;}//user does not exist

         return code;//returning OUTPUT code.
    }
```

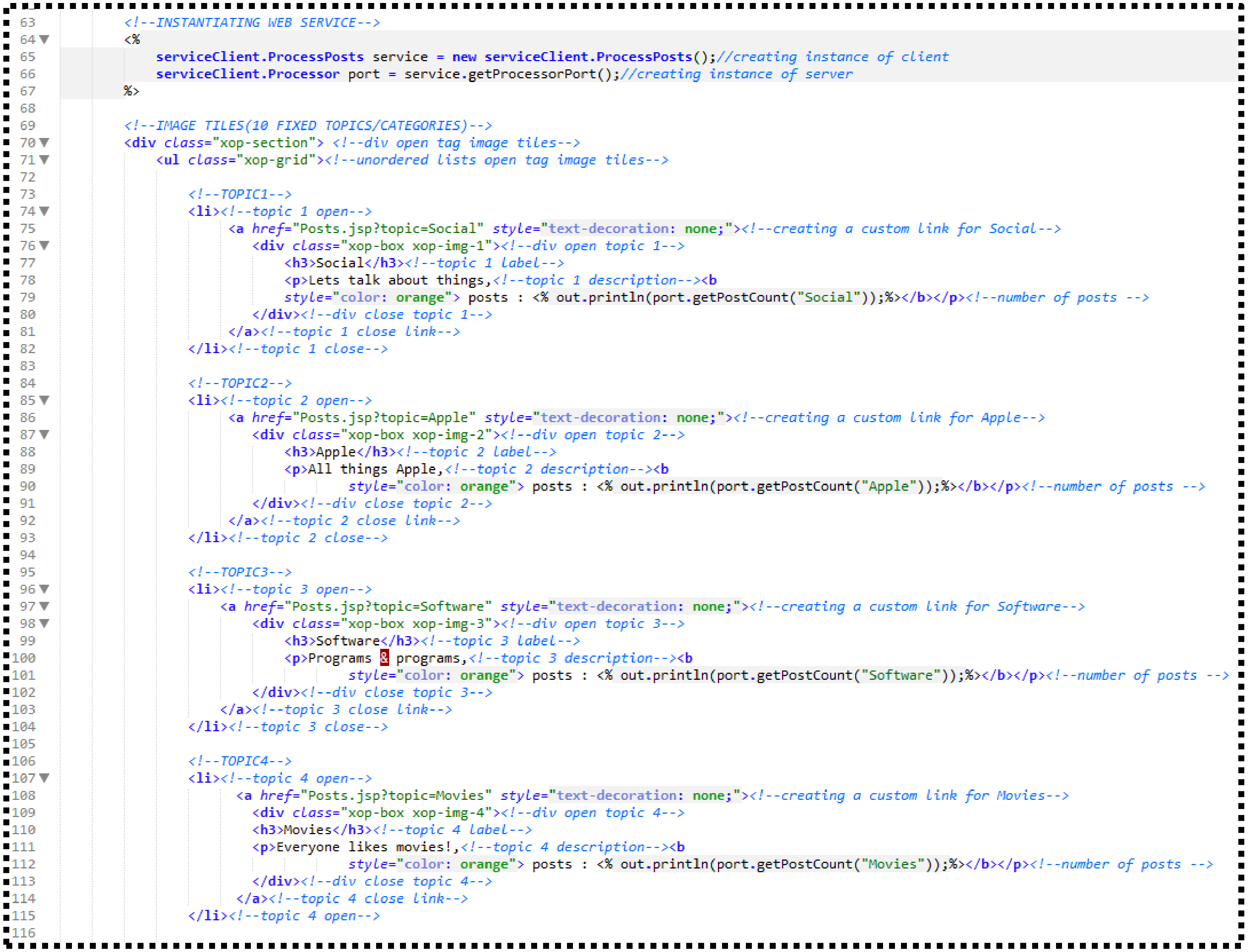


```java
    /**
     * Deletes user account/profile.
     * @param username Example : "bruce7"
     * @param password Example : "brucecanfly"
     * @return returns OUTPUT CODE:
     * OUTPUT CODE MEANINGS:
     * 0 - neutral
     * 1 - user exists
     * 2 - user does not exists
     */
    @WebMethod(operationName="deleteAccount")
    public int deleteAccount(@WebParam(name="username")String username,@WebParam(name="password") String password){
        String addingTime = generateTime();//
        int code = 0;//output code
        int records;//to execute update query

        if(doesUserExist(username, password)==1){//if user exists
           code = 1;//assign 1 to code
        }else if(doesUserExist(username, password)==2){//else if not
            code = 2;//assign 2 to code
        }

        if(code==1){//if user exists
         String query = "delete from Users where userName='"+username+"' and userPassword='"+password+"'";//delete query
            try{//try & catch block for SQL Exception
                Statement statament = connection.createStatement();//creating statement obeject
                records = statament.executeUpdate(query);//executing query
            }   catch (SQLException e){//catching SQL Exception
                System.out.println("SQLException : "+e.toString());//displaying exception message
            }
            String log_out_ACCDEL = "Account deleted by : "+username+" at : "+addingTime;//building LOG string
            writeToLog(log_out_ACCDEL);//writing to LOG
        }
        return code;//returning output CODE.
    }



}
```

Topics.jsp

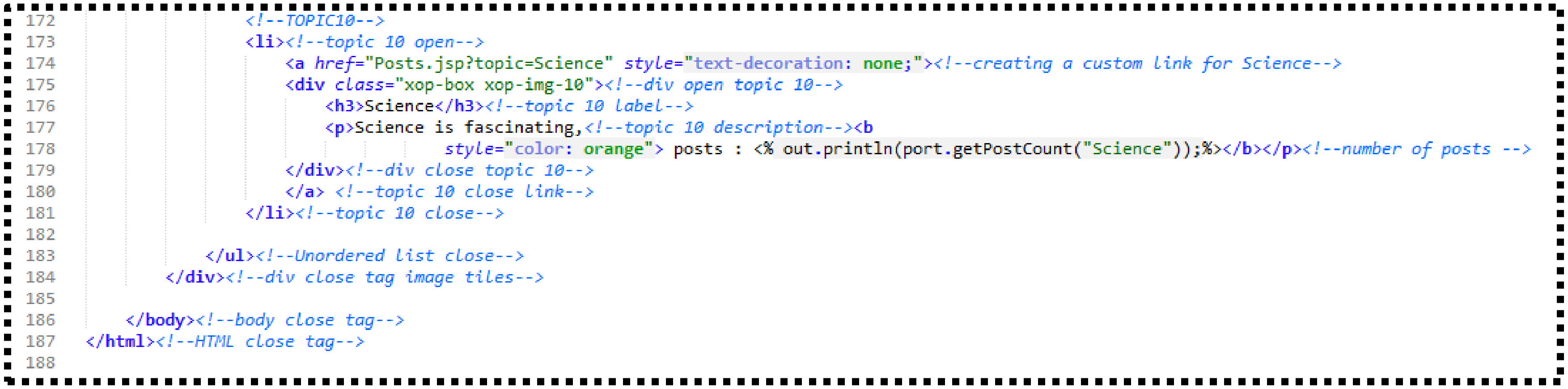
Used to display the topics.

```
<%--
    Filename                : Topics
    Author                  : Aysham Hameed
    Created                 : Dec 7, 2018, 10:57:16 PM
    Operating System        : Windows 10
    Version                 : NetBeans IDE 8.2
    Description             : Used to diplay the topics.
--%>

<html><!--HTML open tag-->
    <head><!--head open tag-->
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"> <!--specify the type of document it is-->
        <title>Topics</title><!--Topics as title of page-->
        <link rel="stylesheet" href="CSS/headerStyle.css"><!--using headerStyle sheet for customizations-->
        <link rel="stylesheet" href="CSS/topicsStyle.css"><!--using topicsStyle sheet for customizations-->
    </head><!--head close tag-->

    <body><!--body open tag-->

        <!--HEADER & NAVIGATION BAR-->
        <header><!--header open tag-->
            <div class="container"><!--div tag-->
                <a href="Topics.jsp"><!--link to home page-->
                   <img src="CSS/logo.png" alt="logo" class="logo"/><!--displaying logo-->
                </a><!--link home page close tag-->

                <nav><!--nav open tag-->
                    <ul><!--unordered list open tag-->
                        <li><a href="Topics.jsp">Home</a></li><!--Home button-->
                        <%
                        if(session.getAttribute("usernameSession")!=null){//If session is created
                            out.println("<li><a href=\"Topics.jsp?logout=true\">");//create hyperlink to homepage
                            String user =  (String)session.getAttribute("usernameSession");//get username
                            out.println("Logout, "+user+"</a></li>");//display Logout button in navigation bar
                            out.println("<li><a href=\"Profile.jsp\">Profile</a></li>");//display Profile button in navigation bar
                        }else{//else if user session is not created then
                            session.setAttribute("pageName", "Topics");//create a session for page name(see login page)
                            out.println("<li><a href=\"Login.jsp\">Login</a></li>");//display Login button
                        }
                        %>
                    </ul><!--unordered list close tag-->
                </nav><!--nav close tag-->
            </div><!--div close tag-->
        </header><!--header close tag-->

        <!--MAIN HEADING-->
        <div style="overflow-x: hidden;"><!--div open tag-->
        <h1 id="mainHeading"><!--assigning id to main heading-->
            Welcome to Fox forums!<!--main heading-->
        </h1><!--main h1 heading close tag-->
        </div><!--main heading div close tag-->

        <!--LOGOUT BUTTON FUNCTIONALITY-->
        <%
            String logOUT = request.getParameter("logout")==null?"0":
            request.getParameter("logout");//getting logout parameter from page url
            if(logOUT.equals("true")){//if logout is true then
                session.invalidate();//invalidate user sesssion
                response.sendRedirect("Topics.jsp");//and go to homepage
            }
        %>
```



```
<!--INSTANTIATING WEB SERVICE-->
<%
    serviceClient.ProcessPosts service = new serviceClient.ProcessPosts();//creating instance of client
    serviceClient.Processor port = service.getProcessorPort();//creating instance of server
%>

<!--IMAGE TILES(10 FIXED TOPICS/CATEGORIES)-->
<div class="xop-section"> <!--div open tag image tiles-->
    <ul class="xop-grid"><!--unordered lists open tag image tiles-->

        <!--TOPIC1-->
        <li><!--topic 1 open-->
             <a href="Posts.jsp?topic=Social" style="text-decoration: none;"><!--creating a custom link for Social-->
                <div class="xop-box xop-img-1"><!--div open topic 1-->
                    <h3>Social</h3><!--topic 1 label-->
                    <p>Lets talk about things,<!--topic 1 description--><b
                    style="color: orange"> posts : <% out.println(port.getPostCount("Social"));%></b></p><!--number of posts -->
                </div><!--div close topic 1-->
             </a><!--topic 1 close link-->
        </li><!--topic 1 close-->

        <!--TOPIC2-->
        <li><!--topic 2 open-->
             <a href="Posts.jsp?topic=Apple" style="text-decoration: none;"><!--creating a custom link for Apple-->
                <div class="xop-box xop-img-2"><!--div open topic 2-->
                    <h3>Apple</h3><!--topic 2 label-->
                    <p>All things Apple,<!--topic 2 description--><b
                            style="color: orange"> posts : <% out.println(port.getPostCount("Apple"));%></b></p><!--number of posts -->
                </div><!--div close topic 2-->
             </a><!--topic 2 close link-->
        </li><!--topic 2 close-->

        <!--TOPIC3-->
        <li><!--topic 3 open-->
            <a href="Posts.jsp?topic=Software" style="text-decoration: none;"><!--creating a custom link for Software-->
                <div class="xop-box xop-img-3"><!--div open topic 3-->
                    <h3>Software</h3><!--topic 3 label-->
                    <p>Programs & programs,<!--topic 3 description--><b
                            style="color: orange"> posts : <% out.println(port.getPostCount("Software"));%></b></p><!--number of posts -->
                </div><!--div close topic 3-->
            </a><!--topic 3 close link-->
        </li><!--topic 3 close-->

        <!--TOPIC4-->
        <li><!--topic 4 open-->
              <a href="Posts.jsp?topic=Movies" style="text-decoration: none;"><!--creating a custom link for Movies-->
                <div class="xop-box xop-img-4"><!--div open topic 4-->
                <h3>Movies</h3><!--topic 4 label-->
                <p>Everyone likes movies!,<!--topic 4 description--><b
                            style="color: orange"> posts : <% out.println(port.getPostCount("Movies"));%></b></p><!--number of posts -->
                </div><!--div close topic 4-->
              </a><!--topic 4 close link-->
        </li><!--topic 4 open-->

```

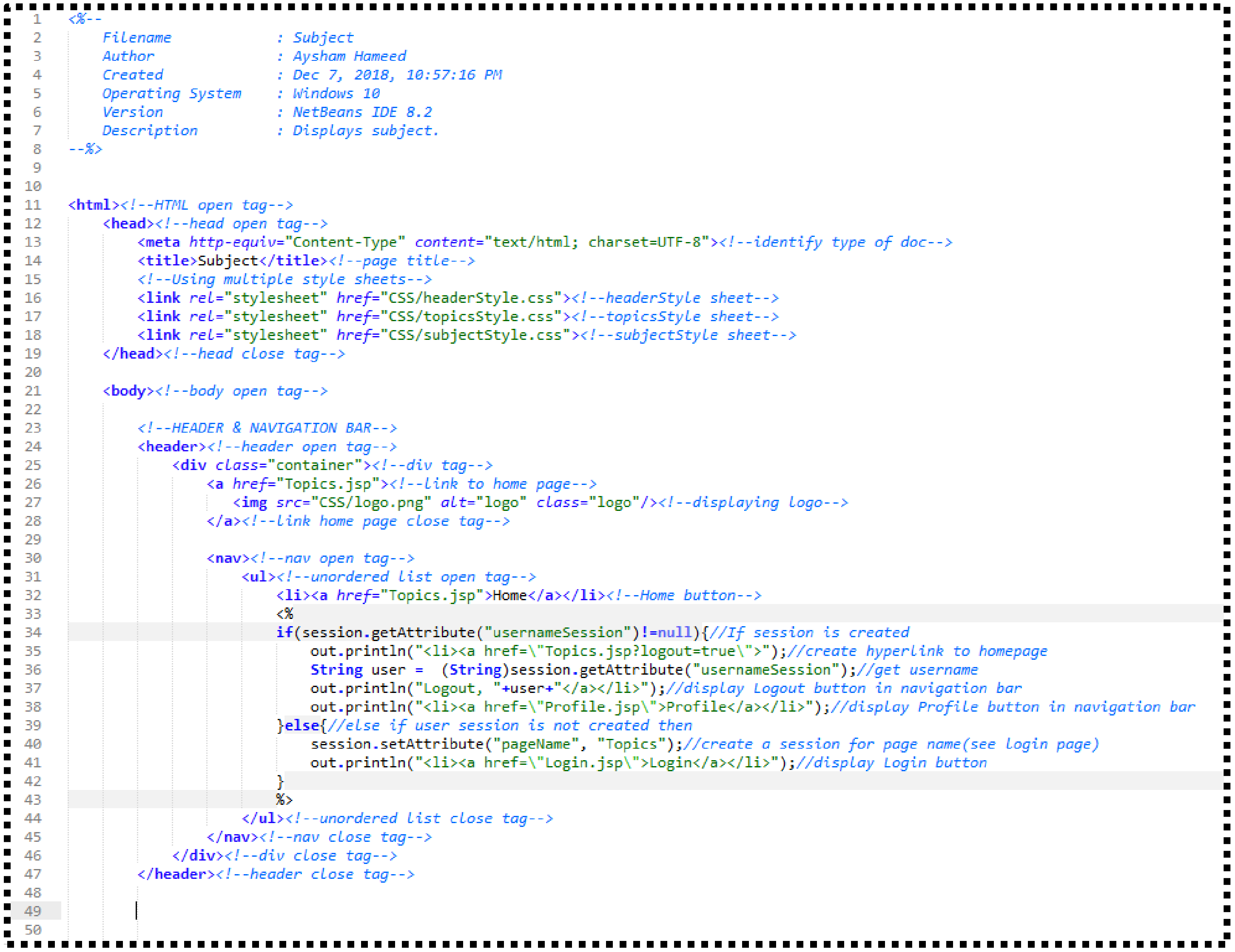
```
<!--TOPIC5-->
<li><!--topic 5 open-->
    <a href="Posts.jsp?topic=Books" style="text-decoration: none;"><!--creating a custom link for Books-->
    <div class="xop-box xop-img-5"><!--div open topic 5-->
        <h3>Books</h3><!--topic 5 label-->
        <p>Lets read together,<!--topic 5 description--><b
                    style="color: orange"> posts : <% out.println(port.getPostCount("Books"));%></b></p><!--number of posts -->
    </div><!--div close topic 5-->
    </a><!--topic 5 close link-->
</li><!--topic 5 close-->

<!--TOPIC6-->
<li><!--topic 6 open-->
    <a href="Posts.jsp?topic=Events" style="text-decoration: none;"><!--creating a custom link for Events-->
    <div class="xop-box xop-img-6"><!--div open topic 6-->
        <h3>Events</h3><!--topic 6 label-->
        <p>For all events discussion,<!--topic 6 description--><b
                    style="color: orange"> posts : <% out.println(port.getPostCount("Events"));%></b></p><!--number of posts -->
    </div><!--div close topic 6-->
    </a><!--topic 6 close link-->
</li><!--topic 6 close-->

<!--TOPIC7-->
<li><!--topic 7 open-->
    <a href="Posts.jsp?topic=Gaming" style="text-decoration: none;"><!--creating a custom link for Gaming-->
    <div class="xop-box xop-img-7"><!--div open topic 7-->
        <h3>Gaming</h3><!--topic 7 label-->
        <p>Achievement unlocked?,<!--topic 7 description--><b
                    style="color: orange"> posts : <% out.println(port.getPostCount("Gaming"));%></b></p><!--number of posts -->
    </div><!--div close topic 7-->
    </a><!--topic 7 close link-->
</li><!--topic 7 close-->

<!--TOPIC8-->
<li><!--topic 8 open-->
    <a href="Posts.jsp?topic=Hacking" style="text-decoration: none;"><!--creating a custom link for Hacking-->
    <div class="xop-box xop-img-8"><!--div open topic 8-->
        <h3>Hacking</h3><!--topic 8 label-->
        <p>Hacking make you feel cool,<!--topic 8 description--><b
                    style="color: orange"> posts : <% out.println(port.getPostCount("Hacking"));%></b></p><!--number of posts -->
    </div><!--div close topic 8-->
    </a><!--topic 8 close link-->
</li><!--topic 8 close-->

<!--TOPIC9-->
<li><!--topic 9 open-->
    <a href="Posts.jsp?topic=Coding" style="text-decoration: none;"><!--creating a custom link for Coding-->
    <div class="xop-box xop-img-9"><!--div open topic 9-->
        <h3>Coding</h3><!--topic 9 label-->
        <p>Source code, problems & more,<!--topic 9 description--><b
                    style="color: orange"> posts : <% out.println(port.getPostCount("Coding"));%></b></p><!--number of posts -->
    </div><!--div close topic 9-->
    </a> <!--topic 9 close link-->
</li><!--topic 9 close-->

```



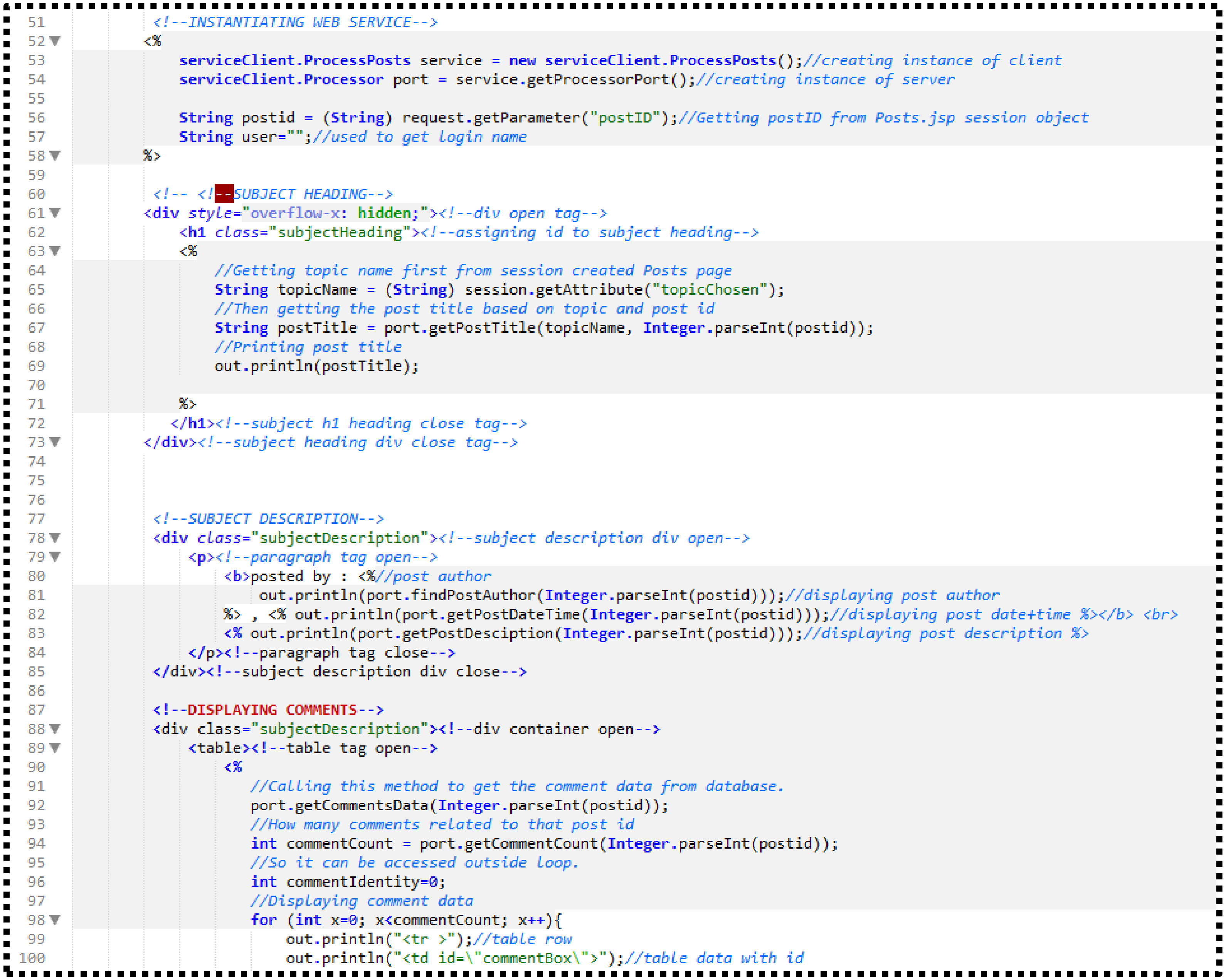
```
                <!--TOPIC10-->
                <li><!--topic 10 open-->
                    <a href="Posts.jsp?topic=Science" style="text-decoration: none;"><!--creating a custom link for Science-->
                    <div class="xop-box xop-img-10"><!--div open topic 10-->
                        <h3>Science</h3><!--topic 10 label-->
                        <p>Science is fascinating,<!--topic 10 description--><b
                                    style="color: orange"> posts : <% out.println(port.getPostCount("Science"));%></b></p><!--number of posts -->
                    </div><!--div close topic 10-->
                    </a> <!--topic 10 close link-->
                </li><!--topic 10 close-->

            </ul><!--Unordered list close-->
        </div><!--div close tag image tiles-->

    </body><!--body close tag-->
</html><!--HTML close tag-->

```

Subject.jsp

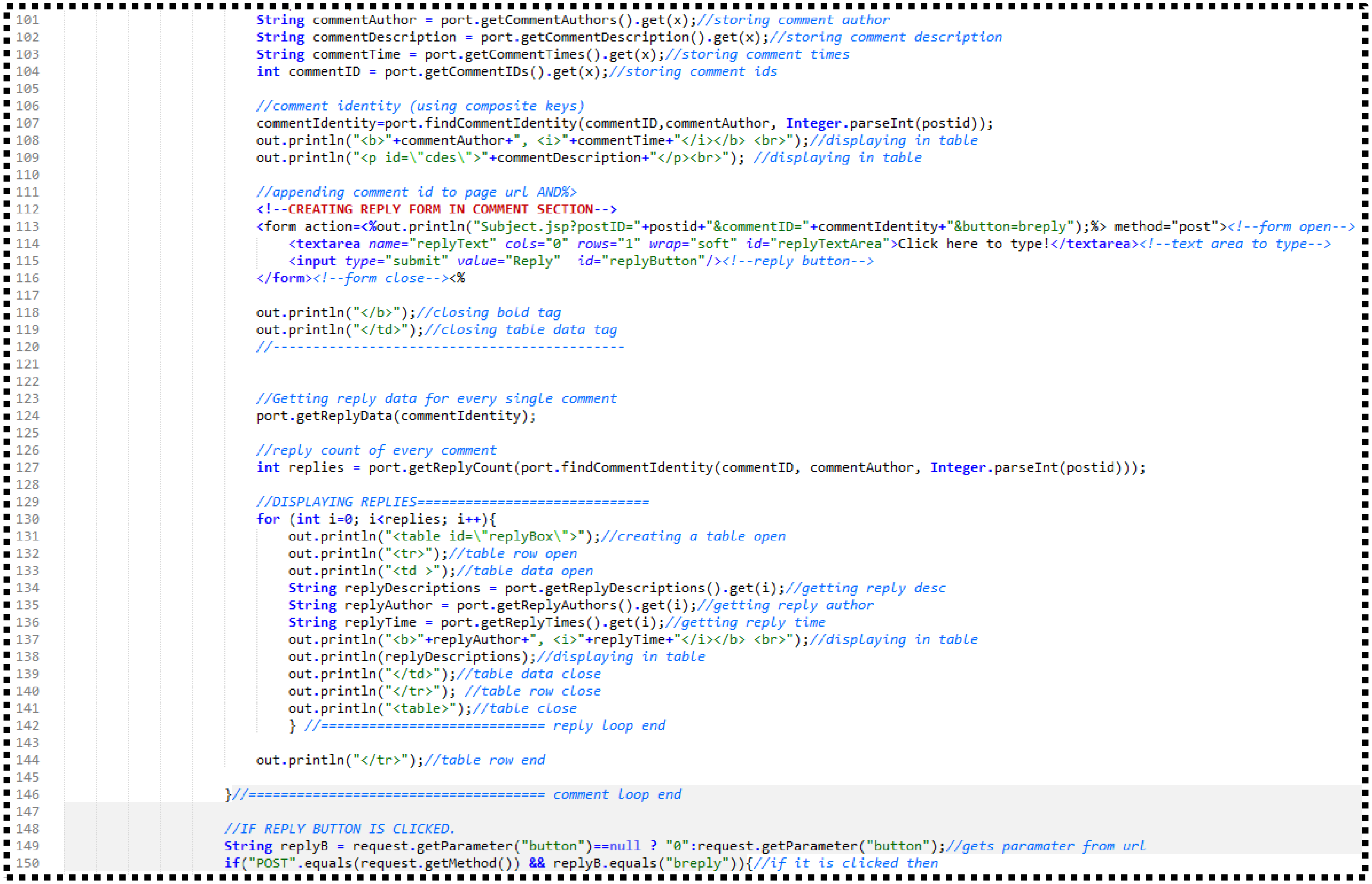
Displays subject (a specific post & its, comments & replies).



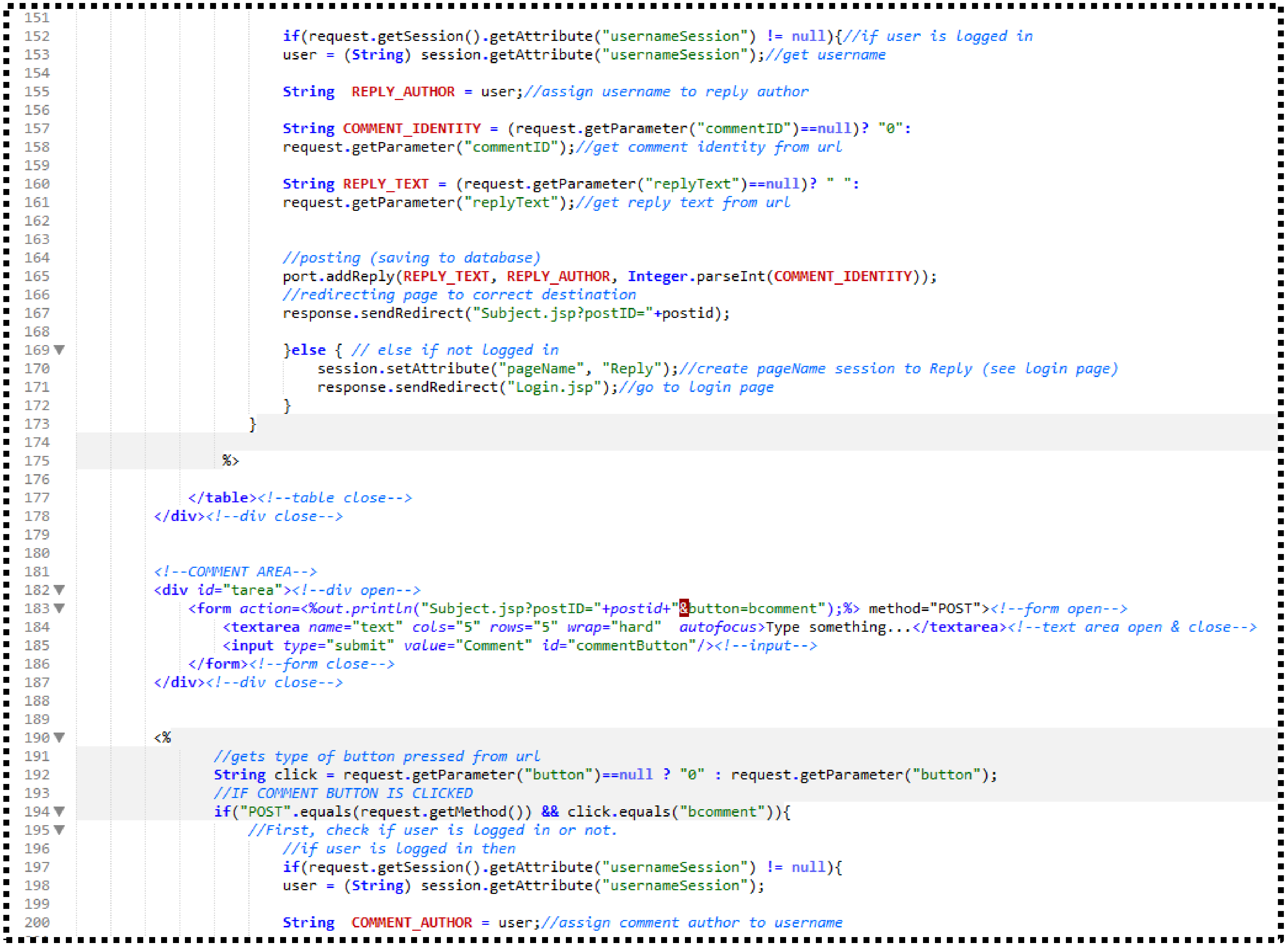
```
<%--
    Filename            : Subject
    Author              : Aysham Hameed
    Created             : Dec 7, 2018, 10:57:16 PM
    Operating System    : Windows 10
    Version             : NetBeans IDE 8.2
    Description         : Displays subject.
--%>


<html><!--HTML open tag-->
    <head><!--head open tag-->
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"><!--identify type of doc-->
        <title>Subject</title><!--page title-->
        <!--Using multiple style sheets-->
        <link rel="stylesheet" href="CSS/headerStyle.css"><!--headerStyle sheet-->
        <link rel="stylesheet" href="CSS/topicsStyle.css"><!--topicsStyle sheet-->
        <link rel="stylesheet" href="CSS/subjectStyle.css"><!--subjectStyle sheet-->
    </head><!--head close tag-->

    <body><!--body open tag-->

        <!--HEADER & NAVIGATION BAR-->
        <header><!--header open tag-->
            <div class="container"><!--div tag-->
                <a href="Topics.jsp"><!--Link to home page-->
                   <img src="CSS/logo.png" alt="logo" class="logo"/><!--displaying logo-->
                </a><!--Link home page close tag-->

                <nav><!--nav open tag-->
                    <ul><!--unordered list open tag-->
                        <li><a href="Topics.jsp">Home</a></li><!--Home button-->
                        <%
                        if(session.getAttribute("usernameSession")!=null){//If session is created
                            out.println("<li><a href=\"Topics.jsp?logout=true\">");//create hyperlink to homepage
                            String user =  (String)session.getAttribute("usernameSession");//get username
                            out.println("Logout, "+user+"</a></li>");//display Logout button in navigation bar
                            out.println("<li><a href=\"Profile.jsp\">Profile</a></li>");//display Profile button in navigation bar
                        }else{//else if user session is not created then
                            session.setAttribute("pageName", "Topics");//create a session for page name(see login page)
                            out.println("<li><a href=\"Login.jsp\">Login</a></li>");//display Login button
                        }
                        %>
                    </ul><!--unordered list close tag-->
                </nav><!--nav close tag-->
            </div><!--div close tag-->
        </header><!--header close tag-->


```



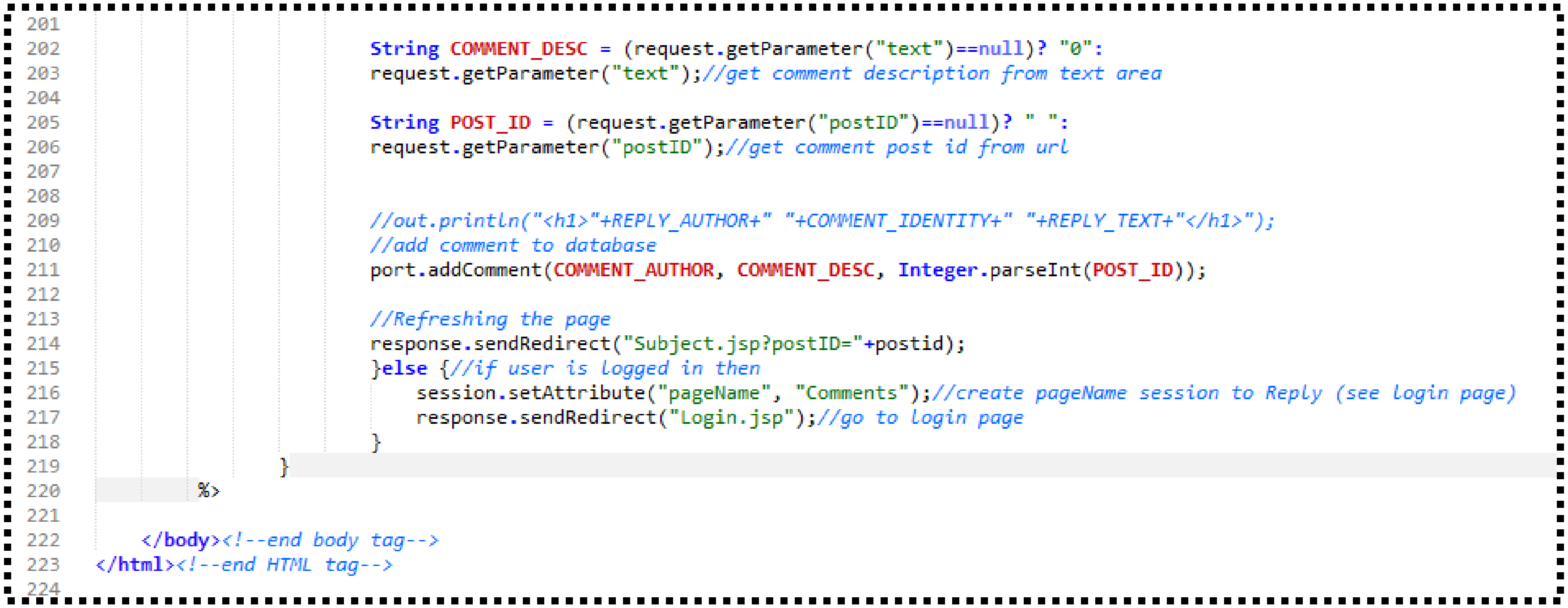
```
     <!--INSTANTIATING WEB SERVICE-->
    <%
        serviceClient.ProcessPosts service = new serviceClient.ProcessPosts();//creating instance of client
        serviceClient.Processor port = service.getProcessorPort();//creating instance of server

        String postid = (String) request.getParameter("postID");//Getting postID from Posts.jsp session object
        String user="";//used to get login name
    %>

     <!-- <!--SUBJECT HEADING-->
    <div style="overflow-x: hidden;"><!--div open tag-->
        <h1 class="subjectHeading"><!--assigning id to subject heading-->
        <%
            //Getting topic name first from session created Posts page
            String topicName = (String) session.getAttribute("topicChosen");
            //Then getting the post title based on topic and post id
            String postTitle = port.getPostTitle(topicName, Integer.parseInt(postid));
            //Printing post title
            out.println(postTitle);

        %>
       </h1><!--subject h1 heading close tag-->
    </div><!--subject heading div close tag-->



     <!--SUBJECT DESCRIPTION-->
     <div class="subjectDescription"><!--subject description div open-->
         <p><!--paragraph tag open-->
             <b>posted by : <%//post author
                 out.println(port.findPostAuthor(Integer.parseInt(postid)));//displaying post author
             %> , <% out.println(port.getPostDateTime(Integer.parseInt(postid)));//displaying post date+time %></b> <br>
             <% out.println(port.getPostDesciption(Integer.parseInt(postid)));//displaying post description %>
         </p><!--paragraph tag close-->
     </div><!--subject description div close-->

     <!--DISPLAYING COMMENTS-->
     <div class="subjectDescription"><!--div container open-->
         <table><!--table tag open-->
             <%
                //Calling this method to get the comment data from database.
                port.getCommentsData(Integer.parseInt(postid));
                //How many comments related to that post id
                int commentCount = port.getCommentCount(Integer.parseInt(postid));
                //So it can be accessed outside loop.
                int commentIdentity=0;
                //Displaying comment data
                for (int x=0; x<commentCount; x++){
                    out.println("<tr >");//table row
                    out.println("<td id=\"commentBox\">");//table data with id
```



```
                        String commentAuthor = port.getCommentAuthors().get(x);//storing comment author
                        String commentDescription = port.getCommentDescription().get(x);//storing comment description
                        String commentTime = port.getCommentTimes().get(x);//storing comment times
                        int commentID = port.getCommentIDs().get(x);//storing comment ids

                        //comment identity (using composite keys)
                        commentIdentity=port.findCommentIdentity(commentID,commentAuthor, Integer.parseInt(postid));
                        out.println("<b>"+commentAuthor+", <i>"+commentTime+"</i></b> <br>");//displaying in table
                        out.println("<p id=\"cdes\">"+commentDescription+"</p><br>"); //displaying in table

                        //appending comment id to page url AND%>
                        <!--CREATING REPLY FORM IN COMMENT SECTION-->
                        <form action=<%out.println("Subject.jsp?postID="+postid+"&commentID="+commentIdentity+"&button=breply");%> method="post"><!--form open-->
                            <textarea name="replyText" cols="0" rows="1" wrap="soft" id="replyTextArea">Click here to type!</textarea><!--text area to type-->
                            <input type="submit" value="Reply"  id="replyButton"/><!--reply button-->
                        </form><!--form close--><%

                        out.println("</b>");//closing bold tag
                        out.println("</td>");//closing table data tag
                        //---------------------------------------------


                        //Getting reply data for every single comment
                        port.getReplyData(commentIdentity);

                        //reply count of every comment
                        int replies = port.getReplyCount(port.findCommentIdentity(commentID, commentAuthor, Integer.parseInt(postid)));

                        //DISPLAYING REPLIES============================
                        for (int i=0; i<replies; i++){
                            out.println("<table id=\"replyBox\">");//creating a table open
                            out.println("<tr>");//table row open
                            out.println("<td >");//table data open
                            String replyDescriptions = port.getReplyDescriptions().get(i);//getting reply desc
                            String replyAuthor = port.getReplyAuthors().get(i);//getting reply author
                            String replyTime = port.getReplyTimes().get(i);//getting reply time
                            out.println("<b>"+replyAuthor+", <i>"+replyTime+"</i></b> <br>");//displaying in table
                            out.println(replyDescriptions);//displaying in table
                            out.println("</td>");//table data close
                            out.println("</tr>"); //table row close
                            out.println("<table>");//table close
                            } //============================ reply loop end

                        out.println("</tr>");//table row end

                    }//==================================== comment loop end

                    //IF REPLY BUTTON IS CLICKED.
                    String replyB = request.getParameter("button")==null ? "0":request.getParameter("button");//gets paramater from url
                    if("POST".equals(request.getMethod()) && replyB.equals("breply")){//if it is clicked then
```



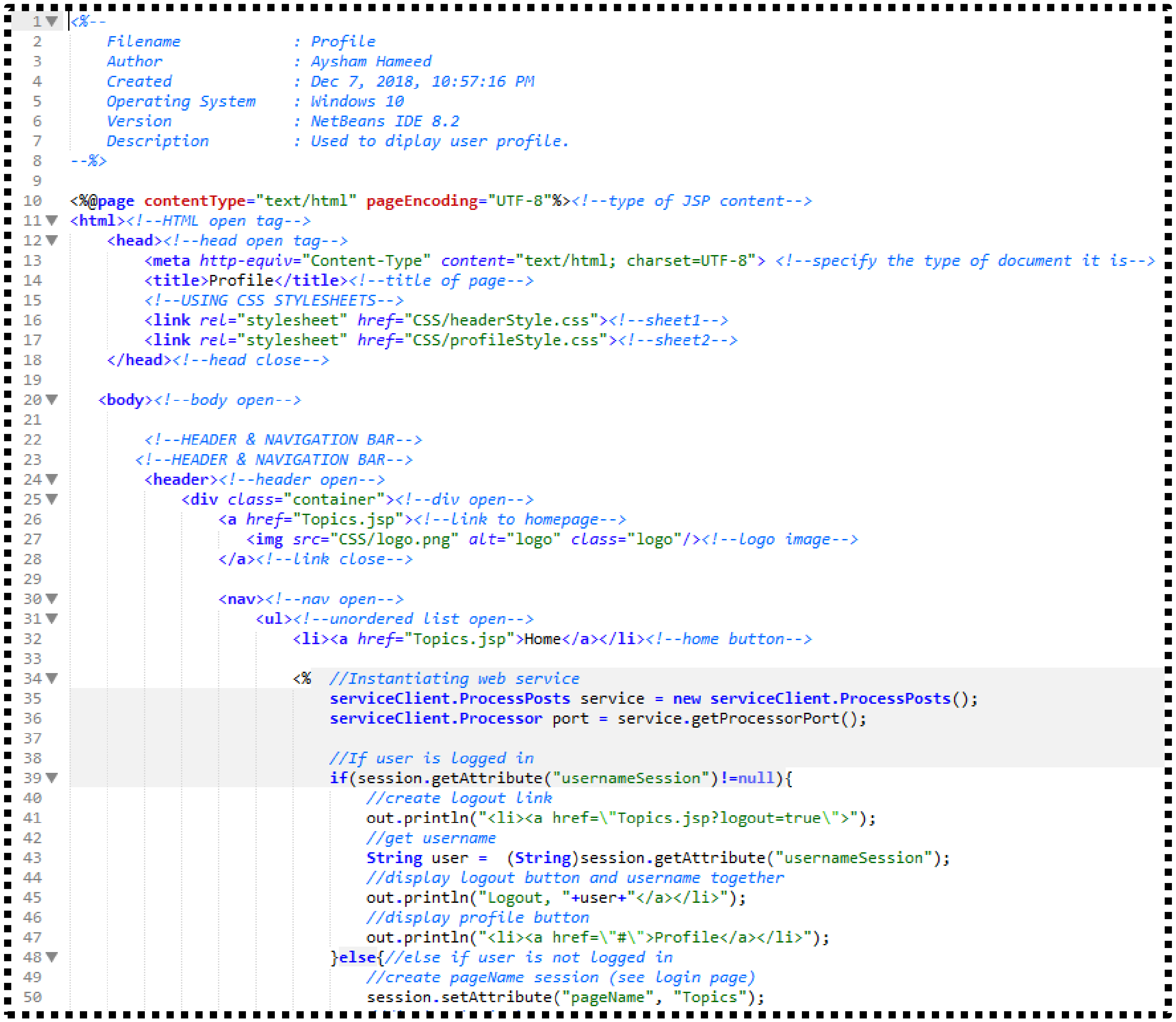
```
                        if(request.getSession().getAttribute("usernameSession") != null){//if user is logged in
                        user = (String) session.getAttribute("usernameSession");//get username

                        String  REPLY_AUTHOR = user;//assign username to reply author

                        String COMMENT_IDENTITY = (request.getParameter("commentID")==null)? "0":
                        request.getParameter("commentID");//get comment identity from url

                        String REPLY_TEXT = (request.getParameter("replyText")==null)? " ":
                        request.getParameter("replyText");//get reply text from url


                        //posting (saving to database)
                        port.addReply(REPLY_TEXT, REPLY_AUTHOR, Integer.parseInt(COMMENT_IDENTITY));
                        //redirecting page to correct destination
                        response.sendRedirect("Subject.jsp?postID="+postid);

                        }else { // else if not logged in
                            session.setAttribute("pageName", "Reply");//create pageName session to Reply (see login page)
                            response.sendRedirect("Login.jsp");//go to login page
                        }
                    }

                %>

            </table><!--table close-->
        </div><!--div close-->


        <!--COMMENT AREA-->
        <div id="tarea"><!--div open-->
            <form action=<%out.println("Subject.jsp?postID="+postid+"&button=bcomment");%> method="POST"><!--form open-->
                <textarea name="text" cols="5" rows="5" wrap="hard"  autofocus>Type something...</textarea><!--text area open & close-->
                <input type="submit" value="Comment" id="commentButton"/><!--input-->
            </form><!--form close-->
        </div><!--div close-->


        <%
                //gets type of button pressed from url
                String click = request.getParameter("button")==null ? "0" : request.getParameter("button");
                //IF COMMENT BUTTON IS CLICKED
                if("POST".equals(request.getMethod()) && click.equals("bcomment")){
                    //First, check if user is logged in or not.
                        //if user is logged in then
                        if(request.getSession().getAttribute("usernameSession") != null){
                        user = (String) session.getAttribute("usernameSession");

                        String  COMMENT_AUTHOR = user;//assign comment author to username
```



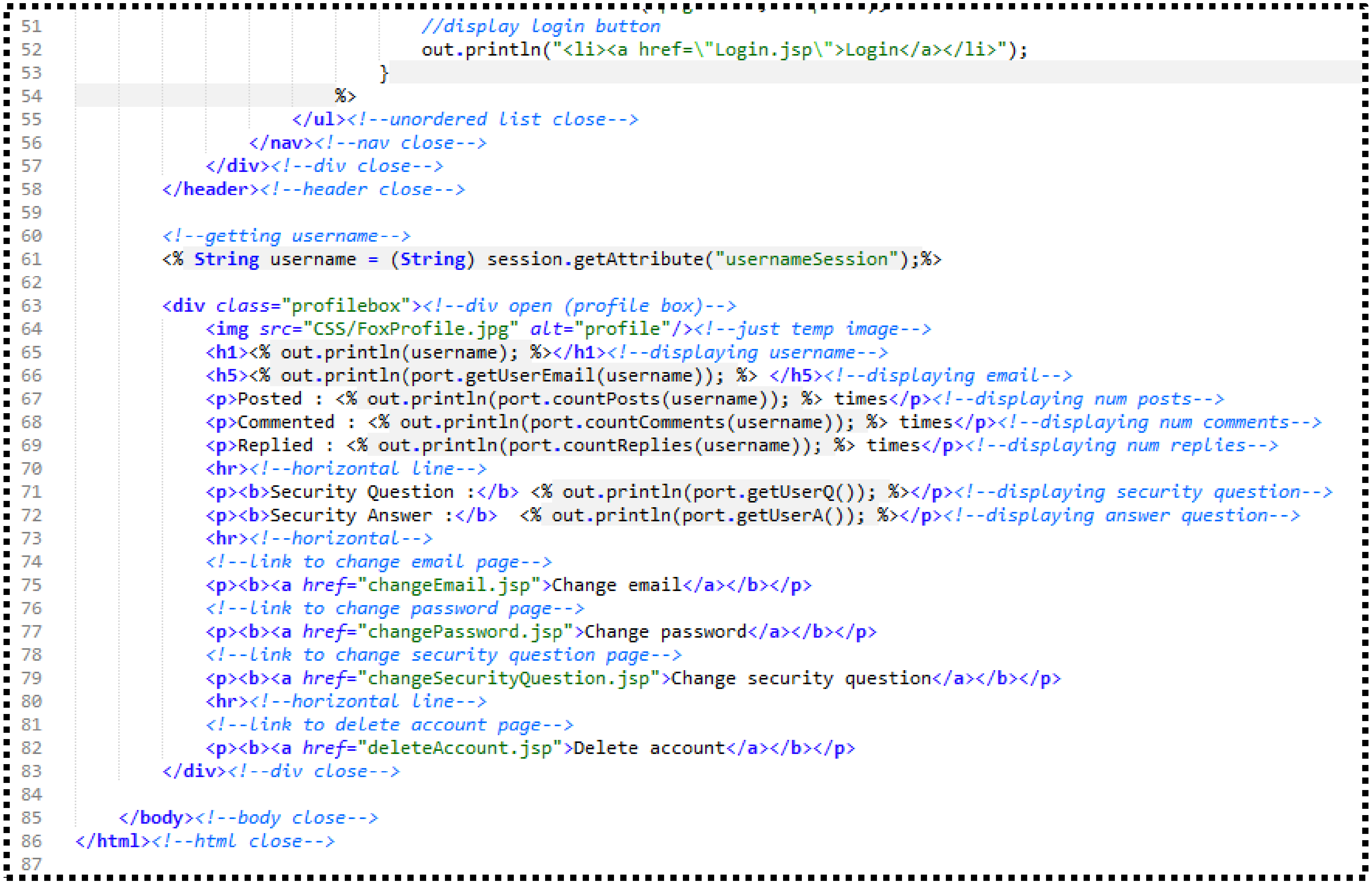
```
                        String COMMENT_DESC = (request.getParameter("text")==null)? "0":
                        request.getParameter("text");//get comment description from text area

                        String POST_ID = (request.getParameter("postID")==null)? " ":
                        request.getParameter("postID");//get comment post id from url


                        //out.println("<h1>"+REPLY_AUTHOR+" "+COMMENT_IDENTITY+" "+REPLY_TEXT+"</h1>");
                        //add comment to database
                        port.addComment(COMMENT_AUTHOR, COMMENT_DESC, Integer.parseInt(POST_ID));

                        //Refreshing the page
                        response.sendRedirect("Subject.jsp?postID="+postid);
                        }else {//if user is logged in then
                            session.setAttribute("pageName", "Comments");//create pageName session to Reply (see login page)
                            response.sendRedirect("Login.jsp");//go to login page
                        }
                }
         %>

    </body><!--end body tag-->
</html><!--end HTML tag-->

```

Profile.jsp

Used to display user profile.



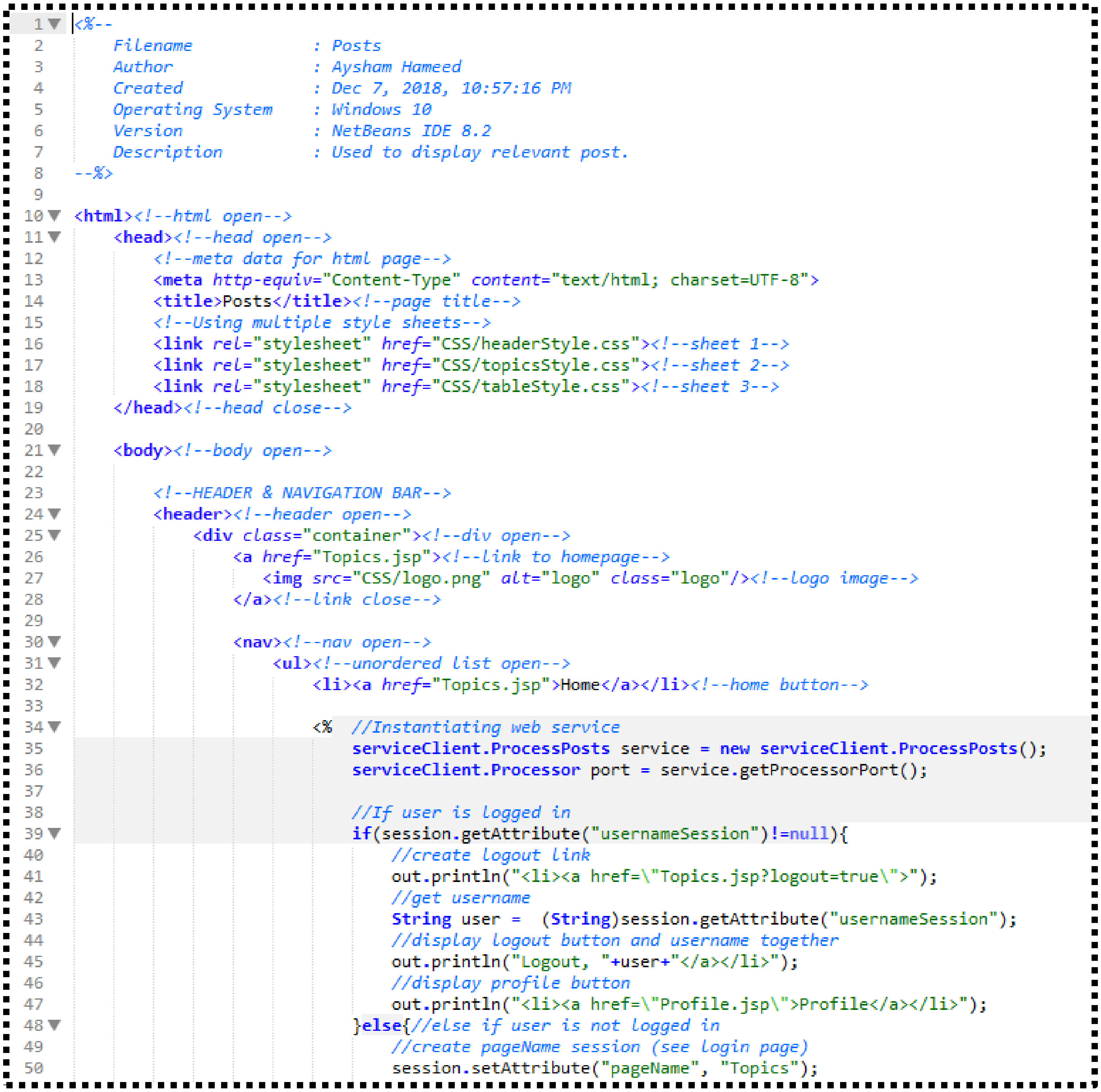
```
<%--
    Filename            : Profile
    Author              : Aysham Hameed
    Created             : Dec 7, 2018, 10:57:16 PM
    Operating System    : Windows 10
    Version             : NetBeans IDE 8.2
    Description         : Used to diplay user profile.
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%><!--type of JSP content-->
<html><!--HTML open tag-->
    <head><!--head open tag-->
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"> <!--specify the type of document it is-->
        <title>Profile</title><!--title of page-->
        <!--USING CSS STYLESHEETS-->
        <link rel="stylesheet" href="CSS/headerStyle.css"><!--sheet1-->
        <link rel="stylesheet" href="CSS/profileStyle.css"><!--sheet2-->
    </head><!--head close-->

   <body><!--body open-->

        <!--HEADER & NAVIGATION BAR-->
       <!--HEADER & NAVIGATION BAR-->
        <header><!--header open-->
            <div class="container"><!--div open-->
                <a href="Topics.jsp"><!--link to homepage-->
                   <img src="CSS/logo.png" alt="logo" class="logo"/><!--logo image-->
                </a><!--link close-->

                <nav><!--nav open-->
                    <ul><!--unordered list open-->
                        <li><a href="Topics.jsp">Home</a></li><!--home button-->

                        <%  //Instantiating web service
                            serviceClient.ProcessPosts service = new serviceClient.ProcessPosts();
                            serviceClient.Processor port = service.getProcessorPort();

                            //If user is logged in
                            if(session.getAttribute("usernameSession")!=null){
                                //create logout link
                                out.println("<li><a href=\"Topics.jsp?logout=true\">");
                                //get username
                                String user =  (String)session.getAttribute("usernameSession");
                                //display logout button and username together
                                out.println("Logout, "+user+"</a></li>");
                                //display profile button
                                out.println("<li><a href=\"#\">Profile</a></li>");
                            }else{//else if user is not logged in
                                //create pageName session (see login page)
                                session.setAttribute("pageName", "Topics");
```



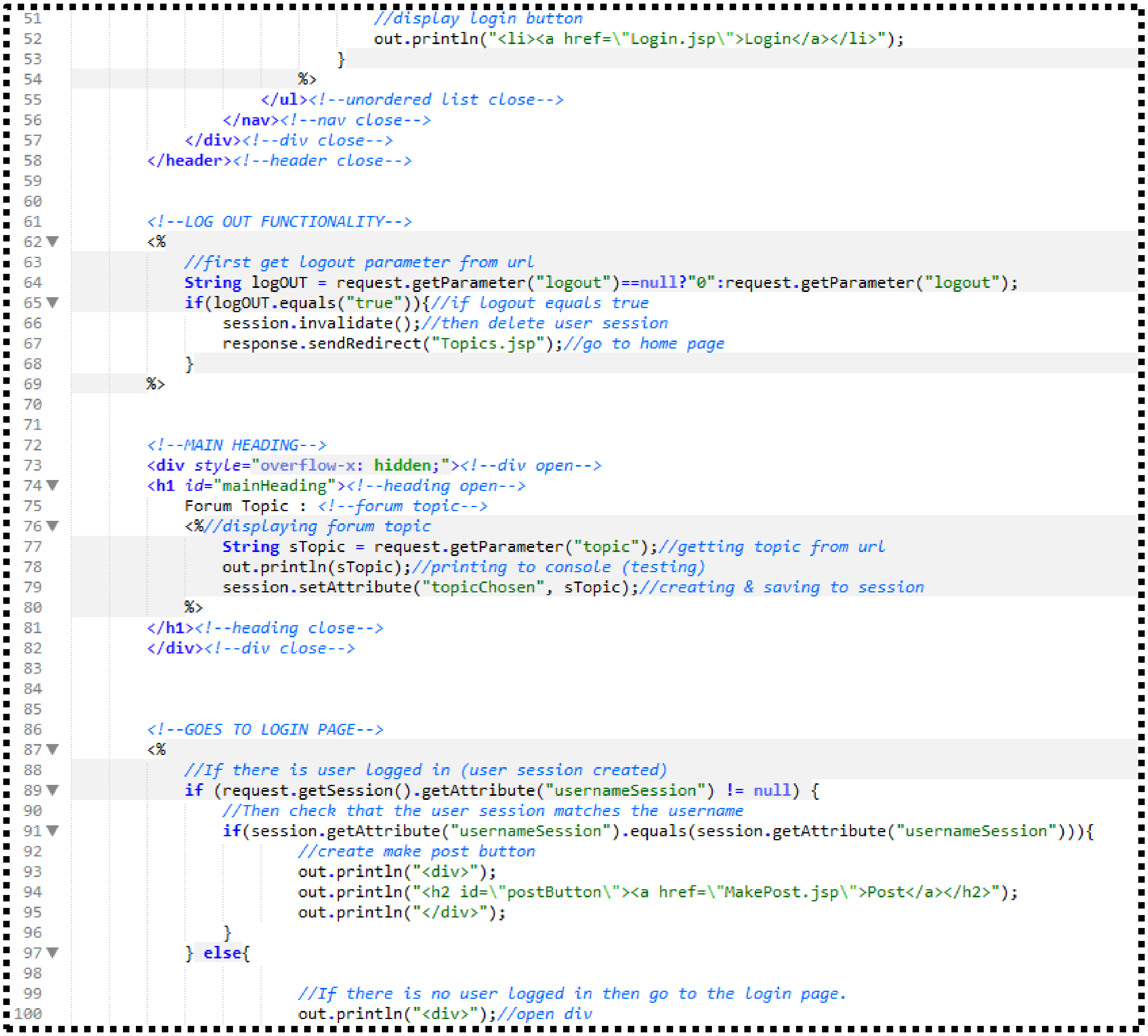
```
                                //display login button
                                out.println("<li><a href=\"Login.jsp\">Login</a></li>");
                            }
                        %>
                    </ul><!--unordered list close-->
                </nav><!--nav close-->
            </div><!--div close-->
        </header><!--header close-->

        <!--getting username-->
        <% String username = (String) session.getAttribute("usernameSession");%>

        <div class="profilebox"><!--div open (profile box)-->
            <img src="CSS/FoxProfile.jpg" alt="profile"/><!--just temp image-->
            <h1><% out.println(username); %></h1><!--displaying username-->
            <h5><% out.println(port.getUserEmail(username)); %> </h5><!--displaying email-->
            <p>Posted : <% out.println(port.countPosts(username)); %> times</p><!--displaying num posts-->
            <p>Commented : <% out.println(port.countComments(username)); %> times</p><!--displaying num comments-->
            <p>Replied : <% out.println(port.countReplies(username)); %> times</p><!--displaying num replies-->
            <hr><!--horizontal line-->
            <p><b>Security Question :</b> <% out.println(port.getUserQ()); %></p><!--displaying security question-->
            <p><b>Security Answer :</b>  <% out.println(port.getUserA()); %></p><!--displaying answer question-->
            <hr><!--horizontal-->
            <!--link to change email page-->
            <p><b><a href="changeEmail.jsp">Change email</a></b></p>
            <!--link to change password page-->
            <p><b><a href="changePassword.jsp">Change password</a></b></p>
            <!--link to change security question page-->
            <p><b><a href="changeSecurityQuestion.jsp">Change security question</a></b></p>
            <hr><!--horizontal line-->
            <!--link to delete account page-->
            <p><b><a href="deleteAccount.jsp">Delete account</a></b></p>
        </div><!--div close-->

    </body><!--body close-->
</html><!--html close-->

```

Posts.jsp

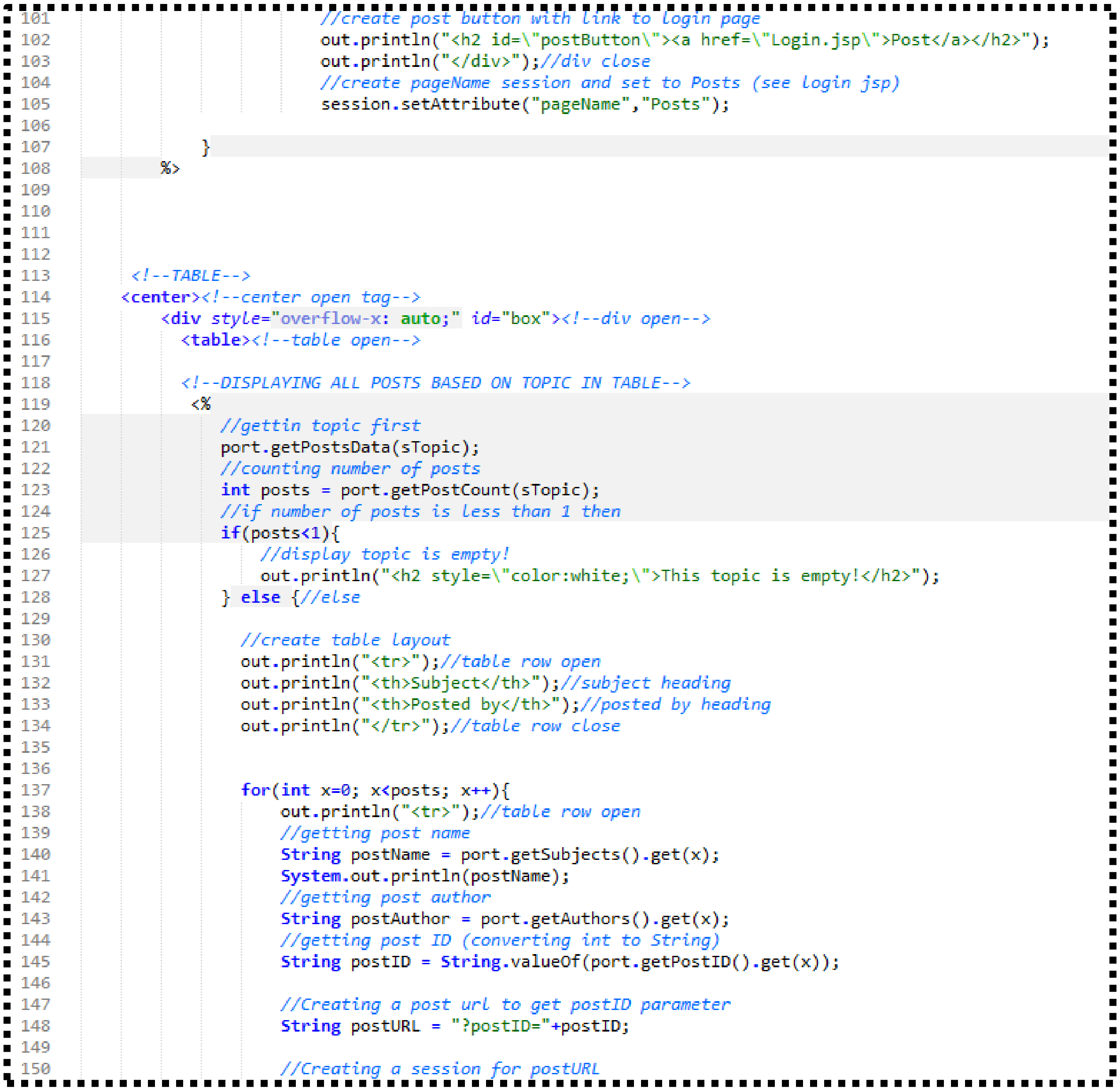
Used to display relevant posts.



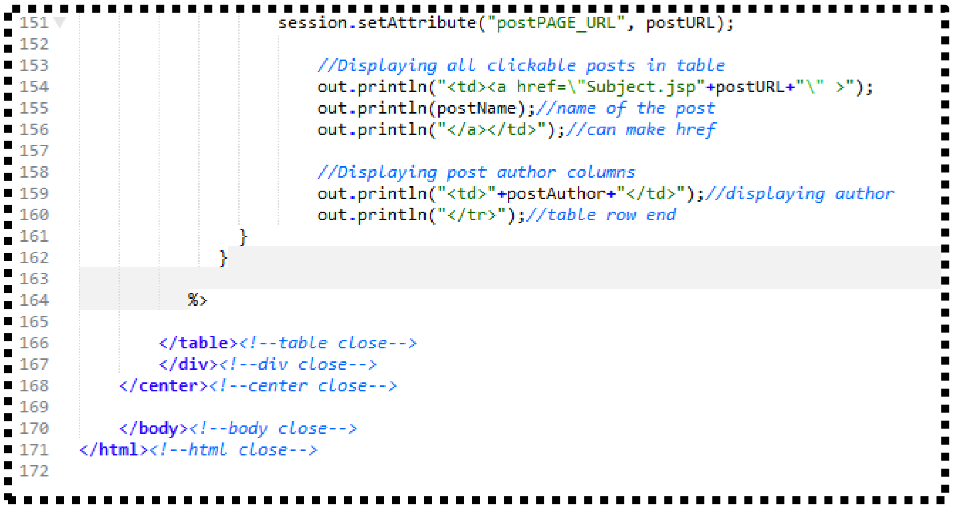
```
<%--
    Filename              : Posts
    Author                : Aysham Hameed
    Created               : Dec 7, 2018, 10:57:16 PM
    Operating System      : Windows 10
    Version               : NetBeans IDE 8.2
    Description           : Used to display relevant post.
--%>

<html><!--html open-->
    <head><!--head open-->
        <!--meta data for html page-->
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Posts</title><!--page title-->
        <!--Using multiple style sheets-->
        <link rel="stylesheet" href="CSS/headerStyle.css"><!--sheet 1-->
        <link rel="stylesheet" href="CSS/topicsStyle.css"><!--sheet 2-->
        <link rel="stylesheet" href="CSS/tableStyle.css"><!--sheet 3-->
    </head><!--head close-->

    <body><!--body open-->

        <!--HEADER & NAVIGATION BAR-->
        <header><!--header open-->
            <div class="container"><!--div open-->
                <a href="Topics.jsp"><!--link to homepage-->
                   <img src="CSS/logo.png" alt="logo" class="logo"/><!--logo image-->
                </a><!--link close-->

                <nav><!--nav open-->
                    <ul><!--unordered list open-->
                        <li><a href="Topics.jsp">Home</a></li><!--home button-->

                        <%  //Instantiating web service
                            serviceClient.ProcessPosts service = new serviceClient.ProcessPosts();
                            serviceClient.Processor port = service.getProcessorPort();

                            //If user is logged in
                            if(session.getAttribute("usernameSession")!=null){
                                //create logout link
                                out.println("<li><a href=\"Topics.jsp?logout=true\">");
                                //get username
                                String user =  (String)session.getAttribute("usernameSession");
                                //display logout button and username together
                                out.println("Logout, "+user+"</a></li>");
                                //display profile button
                                out.println("<li><a href=\"Profile.jsp\">Profile</a></li>");
                            }else{//else if user is not logged in
                                //create pageName session (see login page)
                                session.setAttribute("pageName", "Topics");
```



```
                                //display login button
                                out.println("<li><a href=\"Login.jsp\">Login</a></li>");
                            }
                        %>
                    </ul><!--unordered list close-->
                </nav><!--nav close-->
            </div><!--div close-->
        </header><!--header close-->


        <!--LOG OUT FUNCTIONALITY-->
        <%
            //first get logout parameter from url
            String logOUT = request.getParameter("logout")==null?"0":request.getParameter("logout");
            if(logOUT.equals("true")){//if logout equals true
                session.invalidate();//then delete user session
                response.sendRedirect("Topics.jsp");//go to home page
            }
        %>


        <!--MAIN HEADING-->
        <div style="overflow-x: hidden;"><!--div open-->
        <h1 id="mainHeading"><!--heading open-->
            Forum Topic : <!--forum topic-->
            <%//displaying forum topic
                String sTopic = request.getParameter("topic");//getting topic from url
                out.println(sTopic);//printing to console (testing)
                session.setAttribute("topicChosen", sTopic);//creating & saving to session
            %>
        </h1><!--heading close-->
        </div><!--div close-->



        <!--GOES TO LOGIN PAGE-->
        <%
            //If there is user logged in (user session created)
            if (request.getSession().getAttribute("usernameSession") != null) {
                //Then check that the user session matches the username
                if(session.getAttribute("usernameSession").equals(session.getAttribute("usernameSession"))){
                        //create make post button
                        out.println("<div>");
                        out.println("<h2 id=\"postButton\"><a href=\"MakePost.jsp\">Post</a></h2>");
                        out.println("</div>");
                }
            } else{

                        //If there is no user logged in then go to the login page.
                        out.println("<div>");//open div
```



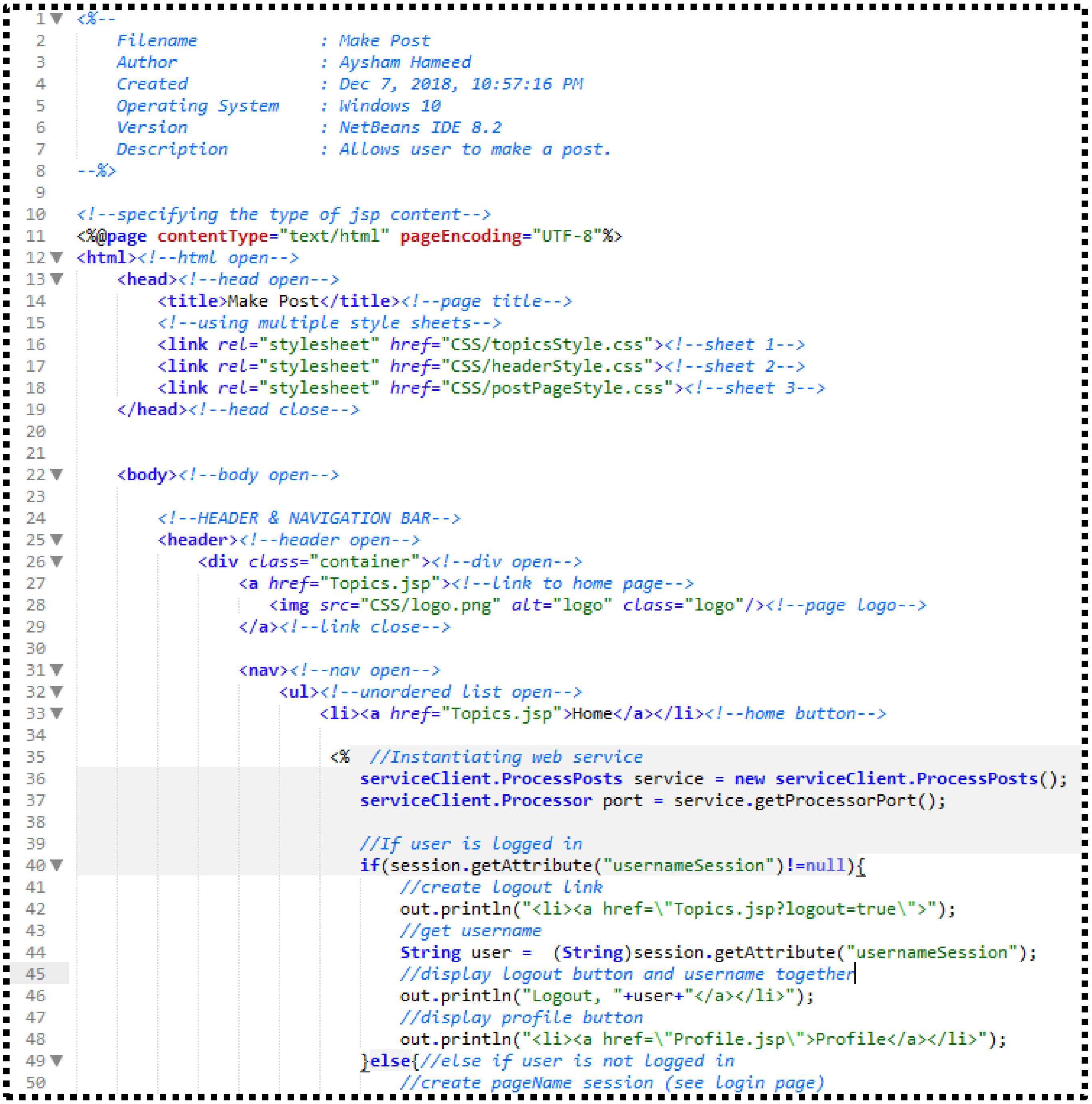
```
                    //create post button with link to login page
                    out.println("<h2 id=\"postButton\"><a href=\"Login.jsp\">Post</a></h2>");
                    out.println("</div>");//div close
                    //create pageName session and set to Posts (see login jsp)
                    session.setAttribute("pageName","Posts");

            }
        %>




 <!--TABLE-->
<center><!--center open tag-->
    <div style="overflow-x: auto;" id="box"><!--div open-->
      <table><!--table open-->

      <!--DISPLAYING ALL POSTS BASED ON TOPIC IN TABLE-->
       <%
          //gettin topic first
          port.getPostsData(sTopic);
          //counting number of posts
          int posts = port.getPostCount(sTopic);
          //if number of posts is less than 1 then
          if(posts<1){
              //display topic is empty!
              out.println("<h2 style=\"color:white;\">This topic is empty!</h2>");
          } else {//else

            //create table layout
            out.println("<tr>");//table row open
            out.println("<th>Subject</th>");//subject heading
            out.println("<th>Posted by</th>");//posted by heading
            out.println("</tr>");//table row close


            for(int x=0; x<posts; x++){
                out.println("<tr>");//table row open
                //getting post name
                String postName = port.getSubjects().get(x);
                System.out.println(postName);
                //getting post author
                String postAuthor = port.getAuthors().get(x);
                //getting post ID (converting int to String)
                String postID = String.valueOf(port.getPostID().get(x));

                //Creating a post url to get postID parameter
                String postURL = "?postID="+postID;

                //Creating a session for postURL
```



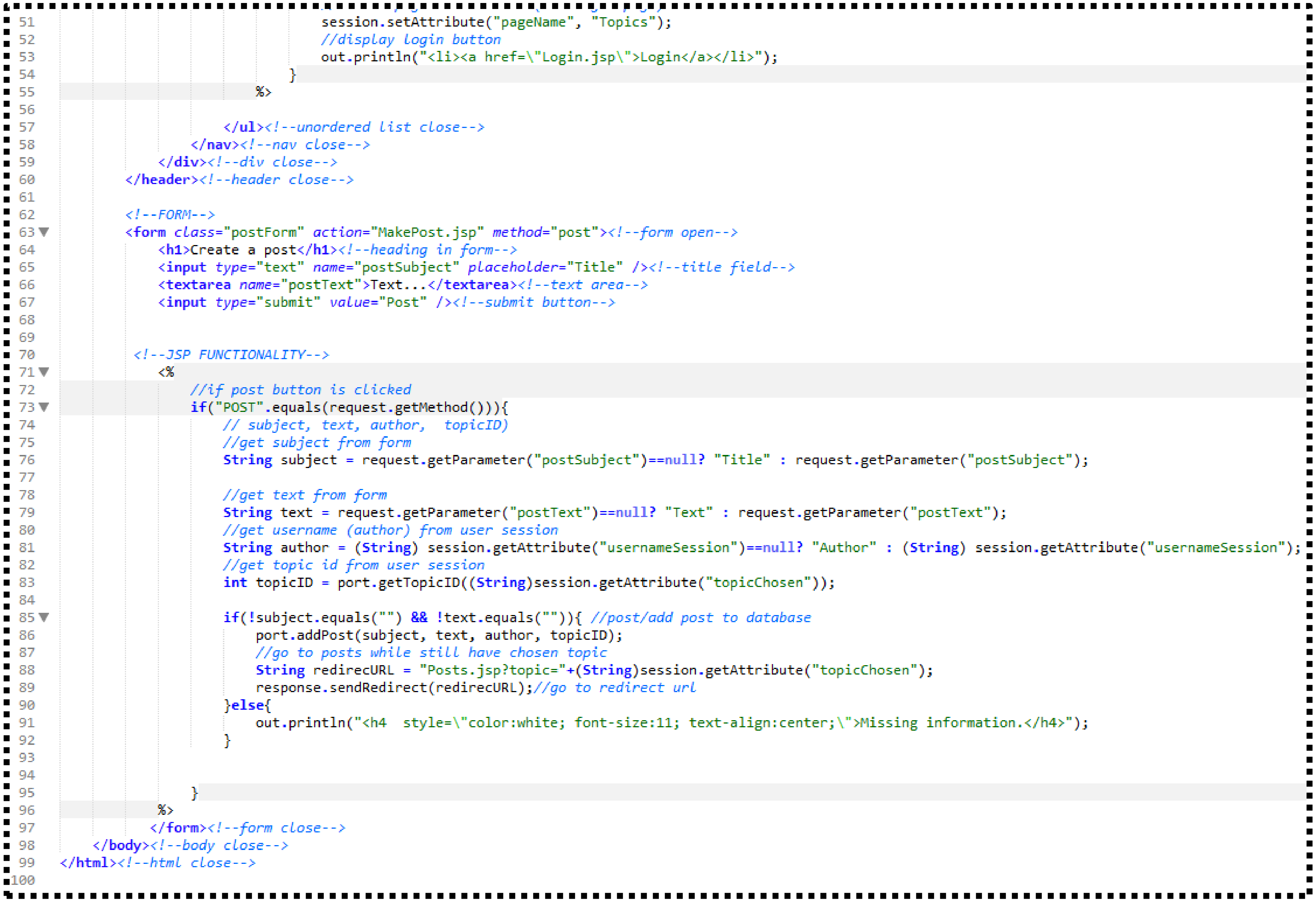
```
                session.setAttribute("postPAGE_URL", postURL);

                    //Displaying all clickable posts in table
                    out.println("<td><a href=\"Subject.jsp"+postURL+"\" >");
                    out.println(postName);//name of the post
                    out.println("</a></td>");//can make href

                    //Displaying post author columns
                    out.println("<td>"+postAuthor+"</td>");//displaying author
                    out.println("</tr>");//table row end
            }
          }

        %>

        </table><!--table close-->
        </div><!--div close-->
    </center><!--center close-->

    </body><!--body close-->
</html><!--html close-->
```

MakePost.jsp

Allows user to make a post.



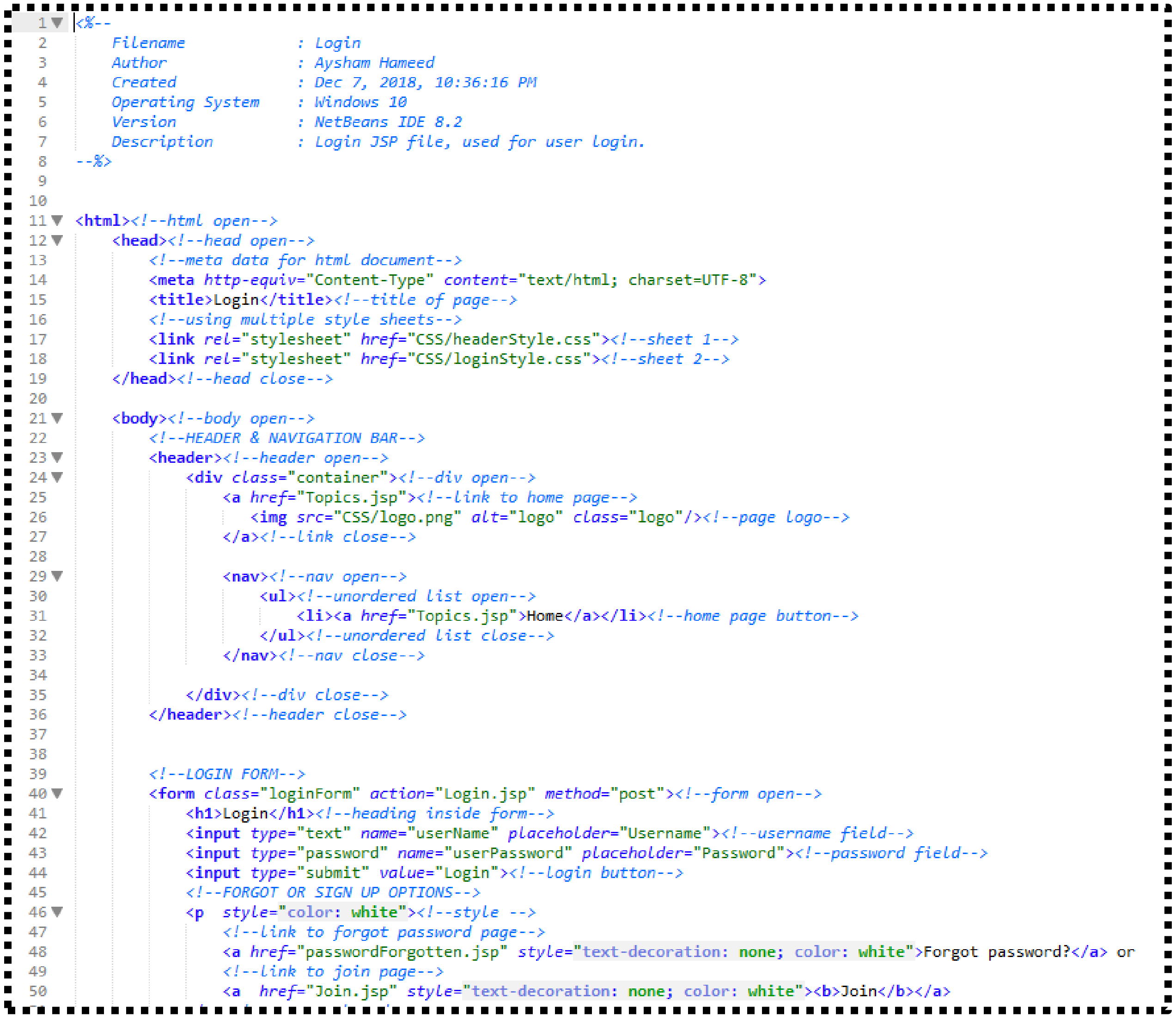
```
<%--
    Filename            : Make Post
    Author              : Aysham Hameed
    Created             : Dec 7, 2018, 10:57:16 PM
    Operating System    : Windows 10
    Version             : NetBeans IDE 8.2
    Description         : Allows user to make a post.
--%>

<!--specifying the type of jsp content-->
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<html><!--html open-->
    <head><!--head open-->
        <title>Make Post</title><!--page title-->
        <!--using multiple style sheets-->
        <link rel="stylesheet" href="CSS/topicsStyle.css"><!--sheet 1-->
        <link rel="stylesheet" href="CSS/headerStyle.css"><!--sheet 2-->
        <link rel="stylesheet" href="CSS/postPageStyle.css"><!--sheet 3-->
    </head><!--head close-->


    <body><!--body open-->

        <!--HEADER & NAVIGATION BAR-->
        <header><!--header open-->
            <div class="container"><!--div open-->
                <a href="Topics.jsp"><!--link to home page-->
                   <img src="CSS/logo.png" alt="logo" class="logo"/><!--page logo-->
                </a><!--link close-->

                <nav><!--nav open-->
                    <ul><!--unordered list open-->
                        <li><a href="Topics.jsp">Home</a></li><!--home button-->

                         <%  //Instantiating web service
                            serviceClient.ProcessPosts service = new serviceClient.ProcessPosts();
                            serviceClient.Processor port = service.getProcessorPort();

                            //If user is logged in
                            if(session.getAttribute("usernameSession")!=null){
                                //create logout link
                                out.println("<li><a href=\"Topics.jsp?logout=true\">");
                                //get username
                                String user =  (String)session.getAttribute("usernameSession");
                                //display logout button and username together
                                out.println("Logout, "+user+"</a></li>");
                                //display profile button
                                out.println("<li><a href=\"Profile.jsp\">Profile</a></li>");
                            }else{//else if user is not logged in
                                //create pageName session (see login page)
```



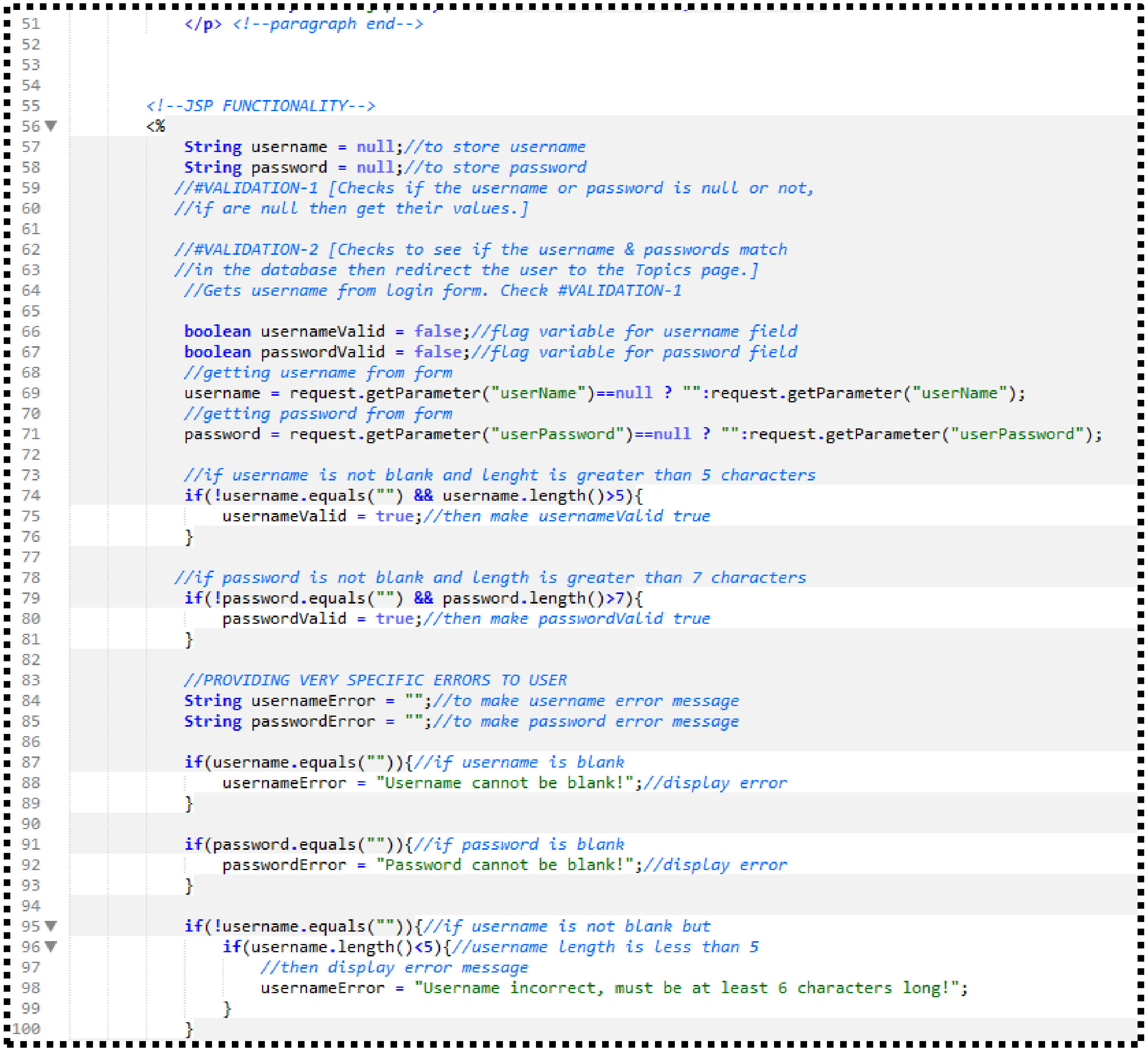
```
                                session.setAttribute("pageName", "Topics");
                                //display login button
                                out.println("<li><a href=\"Login.jsp\">Login</a></li>");
                            }
                        %>

                    </ul><!--unordered list close-->
                </nav><!--nav close-->
            </div><!--div close-->
        </header><!--header close-->

        <!--FORM-->
        <form class="postForm" action="MakePost.jsp" method="post"><!--form open-->
            <h1>Create a post</h1><!--heading in form-->
            <input type="text" name="postSubject" placeholder="Title" /><!--title field-->
            <textarea name="postText">Text...</textarea><!--text area-->
            <input type="submit" value="Post" /><!--submit button-->


         <!--JSP FUNCTIONALITY-->
            <%
                //if post button is clicked
                if("POST".equals(request.getMethod())){
                    // subject, text, author,  topicID)
                    //get subject from form
                    String subject = request.getParameter("postSubject")==null? "Title" : request.getParameter("postSubject");

                    //get text from form
                    String text = request.getParameter("postText")==null? "Text" : request.getParameter("postText");
                    //get username (author) from user session
                    String author = (String) session.getAttribute("usernameSession")==null? "Author" : (String) session.getAttribute("usernameSession");
                    //get topic id from user session
                    int topicID = port.getTopicID((String)session.getAttribute("topicChosen"));

                    if(!subject.equals("") && !text.equals("")){ //post/add post to database
                        port.addPost(subject, text, author, topicID);
                        //go to posts while still have chosen topic
                        String redirecURL = "Posts.jsp?topic="+(String)session.getAttribute("topicChosen");
                        response.sendRedirect(redirecURL);//go to redirect url
                    }else{
                        out.println("<h4  style=\"color:white; font-size:11; text-align:center;\">Missing information.</h4>");
                    }


                }
            %>
           </form><!--form close-->
    </body><!--body close-->
</html><!--html close-->
```

Login.jsp

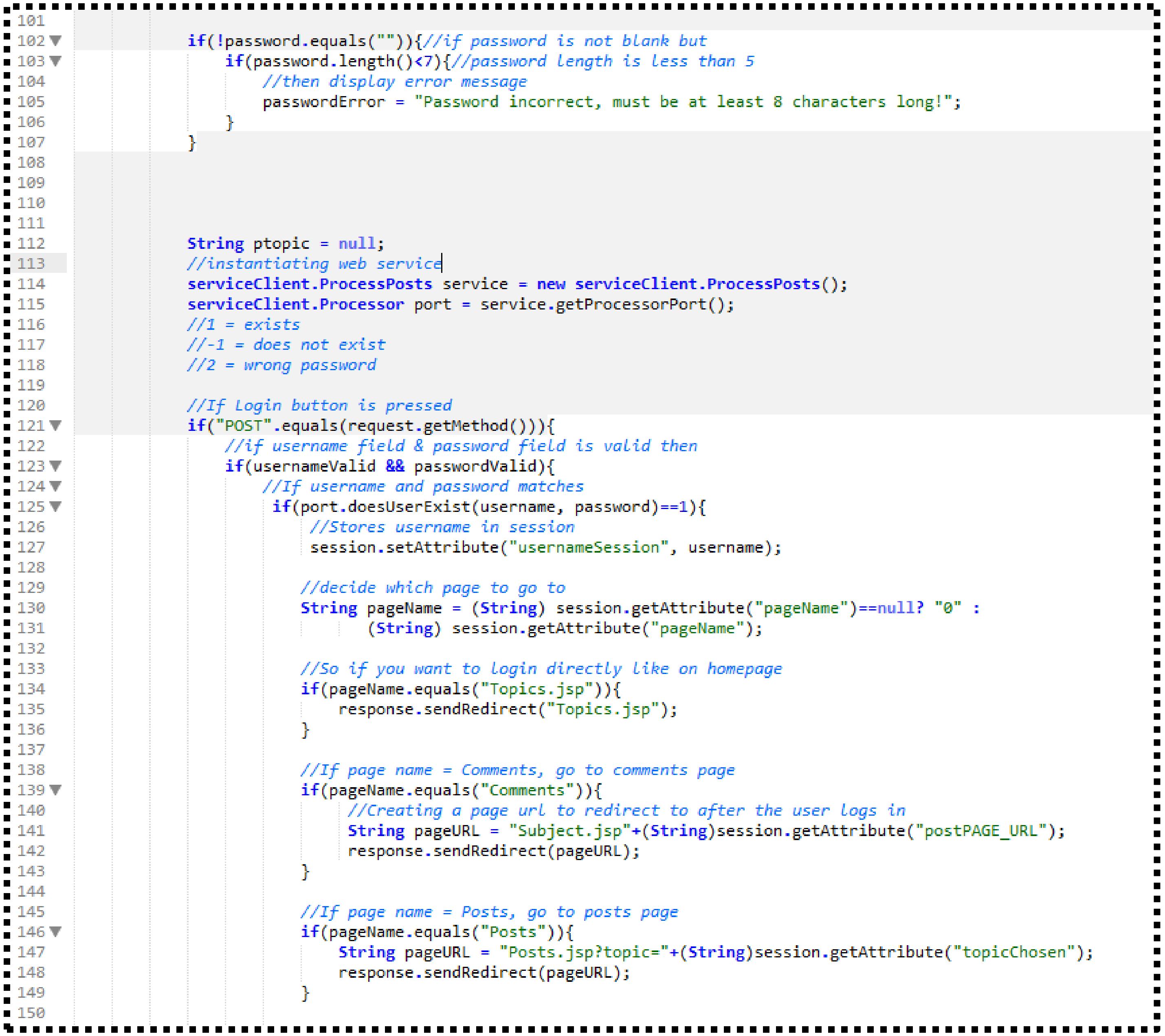
Login JSP file, used for user login.



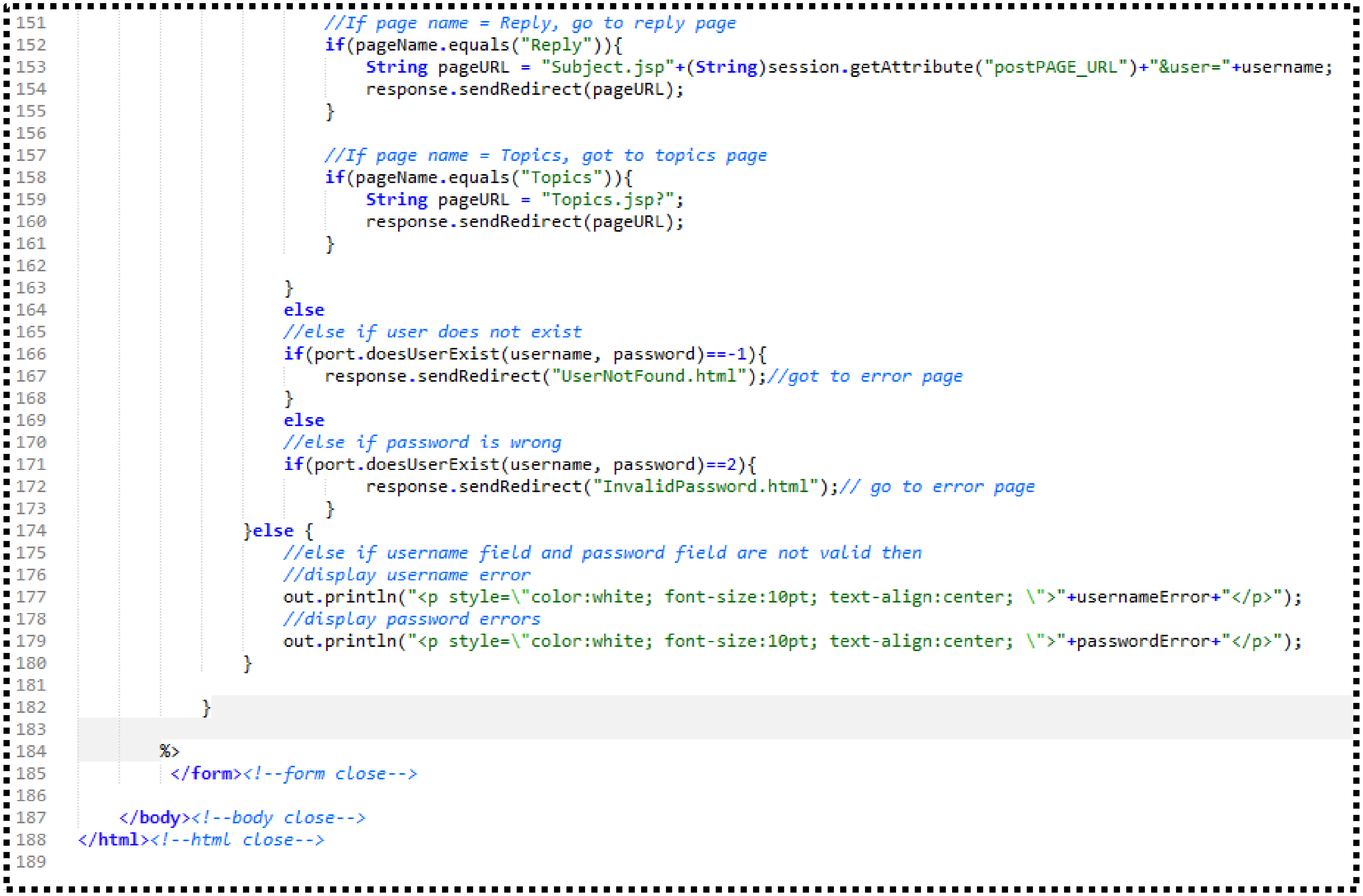
```
<%--
    Filename            : Login
    Author              : Aysham Hameed
    Created             : Dec 7, 2018, 10:36:16 PM
    Operating System    : Windows 10
    Version             : NetBeans IDE 8.2
    Description         : Login JSP file, used for user login.
--%>


<html><!--html open-->
    <head><!--head open-->
        <!--meta data for html document-->
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Login</title><!--title of page-->
        <!--using multiple style sheets-->
        <link rel="stylesheet" href="CSS/headerStyle.css"><!--sheet 1-->
        <link rel="stylesheet" href="CSS/loginStyle.css"><!--sheet 2-->
    </head><!--head close-->

    <body><!--body open-->
        <!--HEADER & NAVIGATION BAR-->
        <header><!--header open-->
            <div class="container"><!--div open-->
                <a href="Topics.jsp"><!--link to home page-->
                   <img src="CSS/logo.png" alt="logo" class="logo"/><!--page logo-->
                </a><!--link close-->

                <nav><!--nav open-->
                    <ul><!--unordered list open-->
                        <li><a href="Topics.jsp">Home</a></li><!--home page button-->
                    </ul><!--unordered list close-->
                </nav><!--nav close-->

            </div><!--div close-->
        </header><!--header close-->


        <!--LOGIN FORM-->
        <form class="loginForm" action="Login.jsp" method="post"><!--form open-->
            <h1>Login</h1><!--heading inside form-->
            <input type="text" name="userName" placeholder="Username"><!--username field-->
            <input type="password" name="userPassword" placeholder="Password"><!--password field-->
            <input type="submit" value="Login"><!--login button-->
            <!--FORGOT OR SIGN UP OPTIONS-->
            <p  style="color: white"><!--style -->
                <!--link to forgot password page-->
                <a href="passwordForgotten.jsp" style="text-decoration: none; color: white">Forgot password?</a> or
                <!--link to join page-->
                <a  href="Join.jsp" style="text-decoration: none; color: white"><b>Join</b></a>
```



```
    </p> <!--paragraph end-->



<!--JSP FUNCTIONALITY-->
<%
    String username = null;//to store username
    String password = null;//to store password
   //#VALIDATION-1 [Checks if the username or password is null or not,
   //if are null then get their values.]

   //#VALIDATION-2 [Checks to see if the username & passwords match
   //in the database then redirect the user to the Topics page.]
    //Gets username from login form. Check #VALIDATION-1

    boolean usernameValid = false;//flag variable for username field
    boolean passwordValid = false;//flag variable for password field
    //getting username from form
    username = request.getParameter("userName")==null ? "":request.getParameter("userName");
    //getting password from form
    password = request.getParameter("userPassword")==null ? "":request.getParameter("userPassword");

    //if username is not blank and lenght is greater than 5 characters
    if(!username.equals("") && username.length()>5){
        usernameValid = true;//then make usernameValid true
    }

   //if password is not blank and length is greater than 7 characters
    if(!password.equals("") && password.length()>7){
        passwordValid = true;//then make passwordValid true
    }

    //PROVIDING VERY SPECIFIC ERRORS TO USER
    String usernameError = "";//to make username error message
    String passwordError = "";//to make password error message

    if(username.equals("")){//if username is blank
        usernameError = "Username cannot be blank!";//display error
    }

    if(password.equals("")){//if password is blank
        passwordError = "Password cannot be blank!";//display error
    }

    if(!username.equals("")){//if username is not blank but
        if(username.length()<5){//username length is less than 5
            //then display error message
            usernameError = "Username incorrect, must be at least 6 characters long!";
        }
    }
```



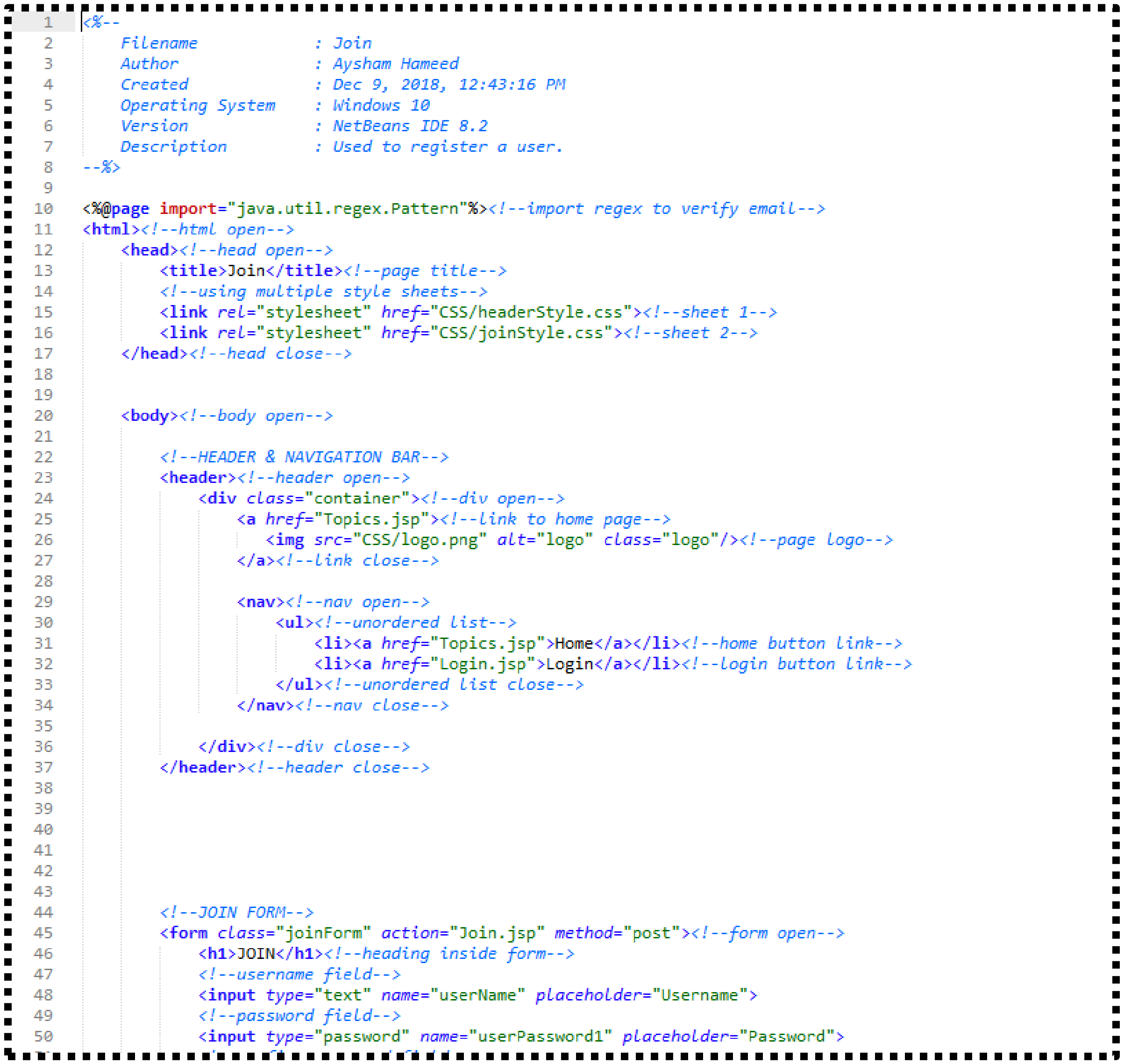
```
if(!password.equals("")){//if password is not blank but
    if(password.length()<7){//password length is less than 5
        //then display error message
        passwordError = "Password incorrect, must be at least 8 characters long!";
    }
}




String ptopic = null;
//instantiating web service
serviceClient.ProcessPosts service = new serviceClient.ProcessPosts();
serviceClient.Processor port = service.getProcessorPort();
//1 = exists
//-1 = does not exist
//2 = wrong password

//If Login button is pressed
if("POST".equals(request.getMethod())){
    //if username field & password field is valid then
    if(usernameValid && passwordValid){
        //If username and password matches
         if(port.doesUserExist(username, password)==1){
             //Stores username in session
             session.setAttribute("usernameSession", username);

            //decide which page to go to
            String pageName = (String) session.getAttribute("pageName")==null? "0" :
                   (String) session.getAttribute("pageName");

            //So if you want to login directly like on homepage
            if(pageName.equals("Topics.jsp")){
                response.sendRedirect("Topics.jsp");
            }

            //If page name = Comments, go to comments page
            if(pageName.equals("Comments")){
                 //Creating a page url to redirect to after the user logs in
                 String pageURL = "Subject.jsp"+(String)session.getAttribute("postPAGE_URL");
                 response.sendRedirect(pageURL);
            }

            //If page name = Posts, go to posts page
            if(pageName.equals("Posts")){
                String pageURL = "Posts.jsp?topic="+(String)session.getAttribute("topicChosen");
                response.sendRedirect(pageURL);
            }
```



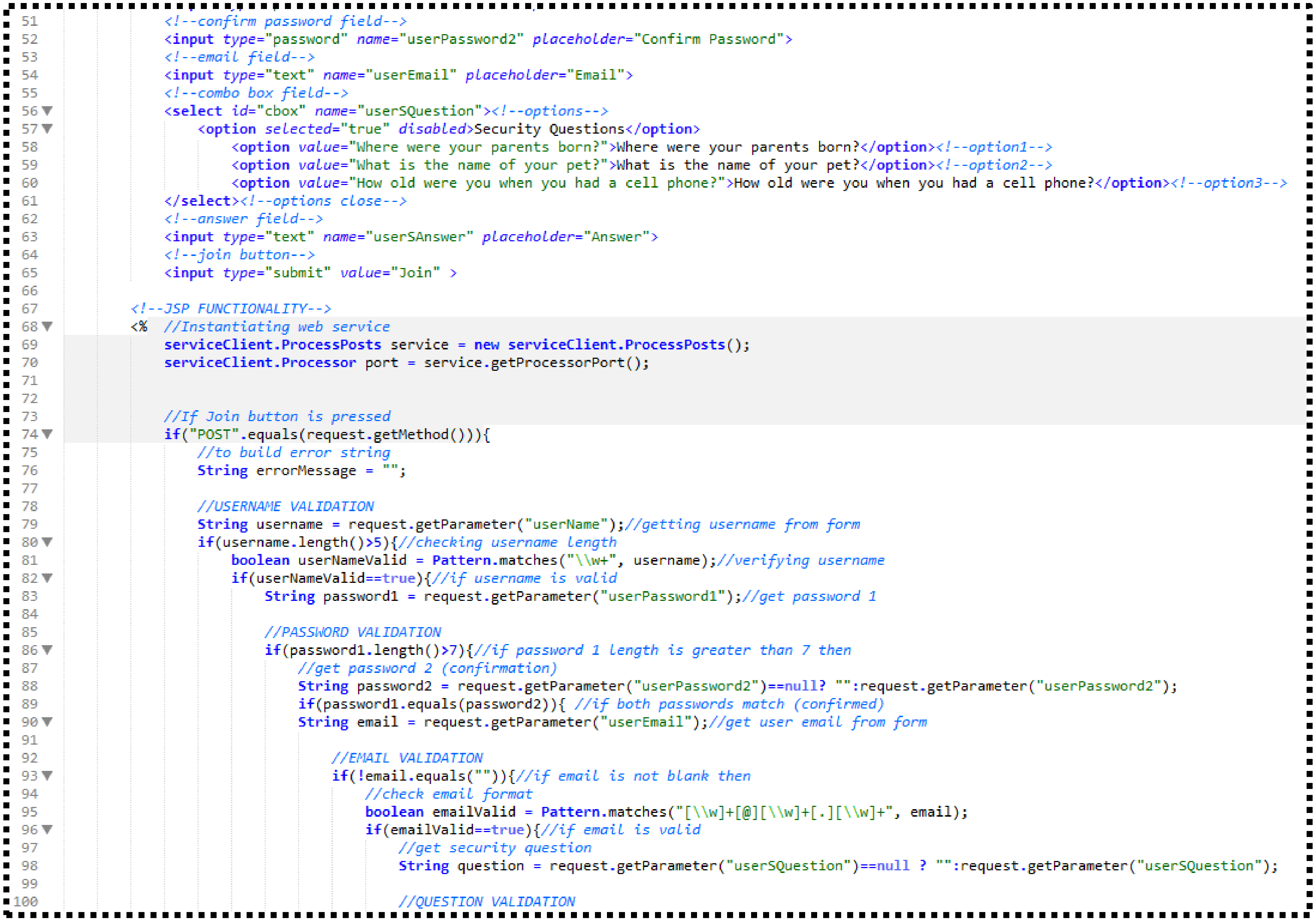
```
                        //If page name = Reply, go to reply page
                        if(pageName.equals("Reply")){
                            String pageURL = "Subject.jsp"+(String)session.getAttribute("postPAGE_URL")+"&user="+username;
                            response.sendRedirect(pageURL);
                        }

                        //If page name = Topics, got to topics page
                        if(pageName.equals("Topics")){
                            String pageURL = "Topics.jsp?";
                            response.sendRedirect(pageURL);
                        }

                    }
                    else
                    //else if user does not exist
                    if(port.doesUserExist(username, password)==-1){
                        response.sendRedirect("UserNotFound.html");//got to error page
                    }
                    else
                    //else if password is wrong
                    if(port.doesUserExist(username, password)==2){
                            response.sendRedirect("InvalidPassword.html");// go to error page
                        }
                }else {
                    //else if username field and password field are not valid then
                    //display username error
                    out.println("<p style=\"color:white; font-size:10pt; text-align:center; \">"+usernameError+"</p>");
                    //display password errors
                    out.println("<p style=\"color:white; font-size:10pt; text-align:center; \">"+passwordError+"</p>");
                }

            }

        %>
         </form><!--form close-->

    </body><!--body close-->
</html><!--html close-->

```

Join.jsp

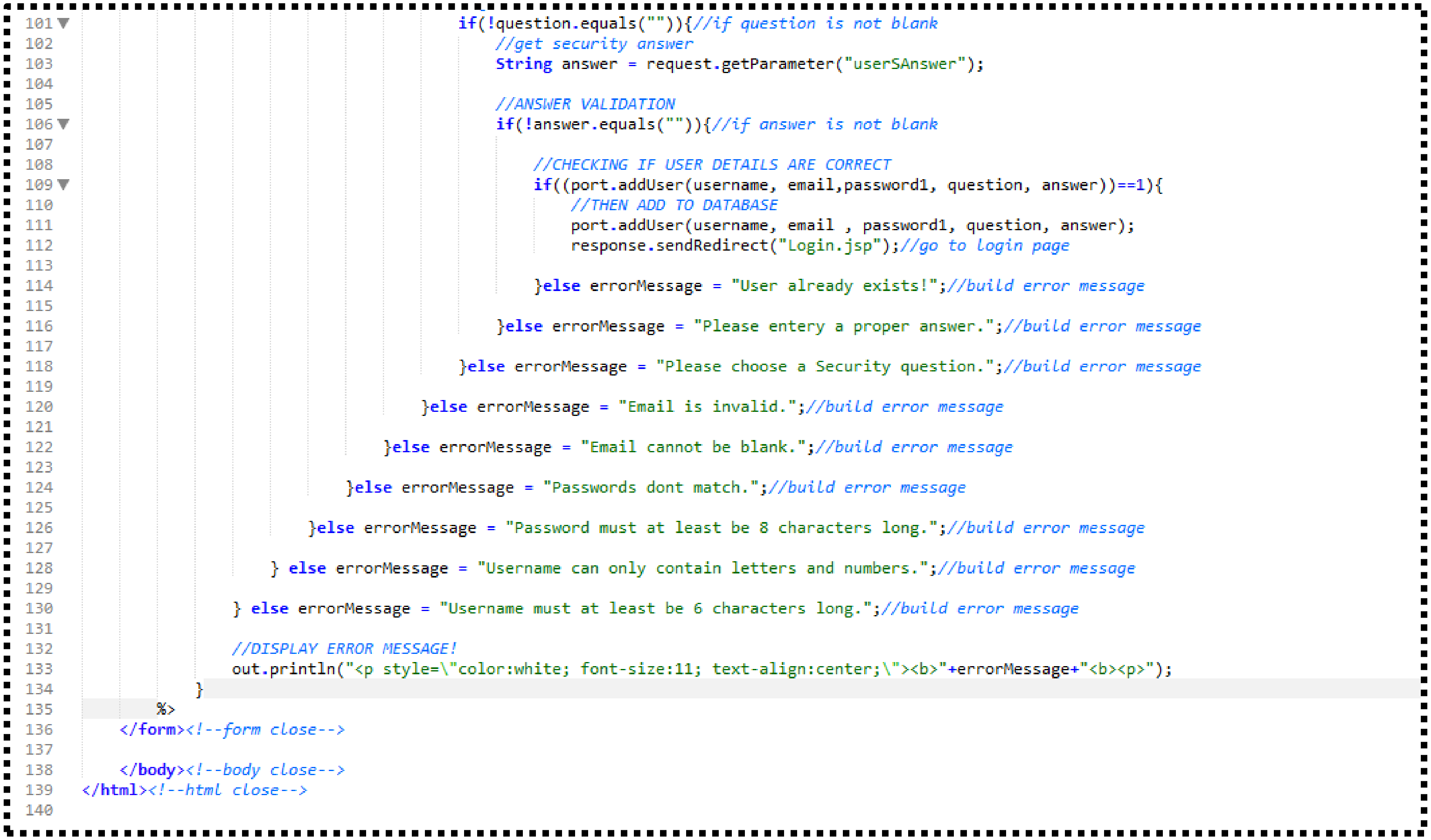
Used to register a user.



```
<%--
    Filename                : Join
    Author                  : Aysham Hameed
    Created                 : Dec 9, 2018, 12:43:16 PM
    Operating System        : Windows 10
    Version                 : NetBeans IDE 8.2
    Description             : Used to register a user.
--%>

<%@page import="java.util.regex.Pattern"%><!--import regex to verify email-->
<html><!--html open-->
    <head><!--head open-->
        <title>Join</title><!--page title-->
        <!--using multiple style sheets-->
        <link rel="stylesheet" href="CSS/headerStyle.css"><!--sheet 1-->
        <link rel="stylesheet" href="CSS/joinStyle.css"><!--sheet 2-->
    </head><!--head close-->


    <body><!--body open-->

        <!--HEADER & NAVIGATION BAR-->
        <header><!--header open-->
            <div class="container"><!--div open-->
                <a href="Topics.jsp"><!--link to home page-->
                   <img src="CSS/logo.png" alt="logo" class="logo"/><!--page logo-->
                </a><!--link close-->

                <nav><!--nav open-->
                    <ul><!--unordered list-->
                        <li><a href="Topics.jsp">Home</a></li><!--home button link-->
                        <li><a href="Login.jsp">Login</a></li><!--login button link-->
                    </ul><!--unordered list close-->
                </nav><!--nav close-->

            </div><!--div close-->
        </header><!--header close-->






        <!--JOIN FORM-->
        <form class="joinForm" action="Join.jsp" method="post"><!--form open-->
            <h1>JOIN</h1><!--heading inside form-->
            <!--username field-->
            <input type="text" name="userName" placeholder="Username">
            <!--password field-->
            <input type="password" name="userPassword1" placeholder="Password">
```



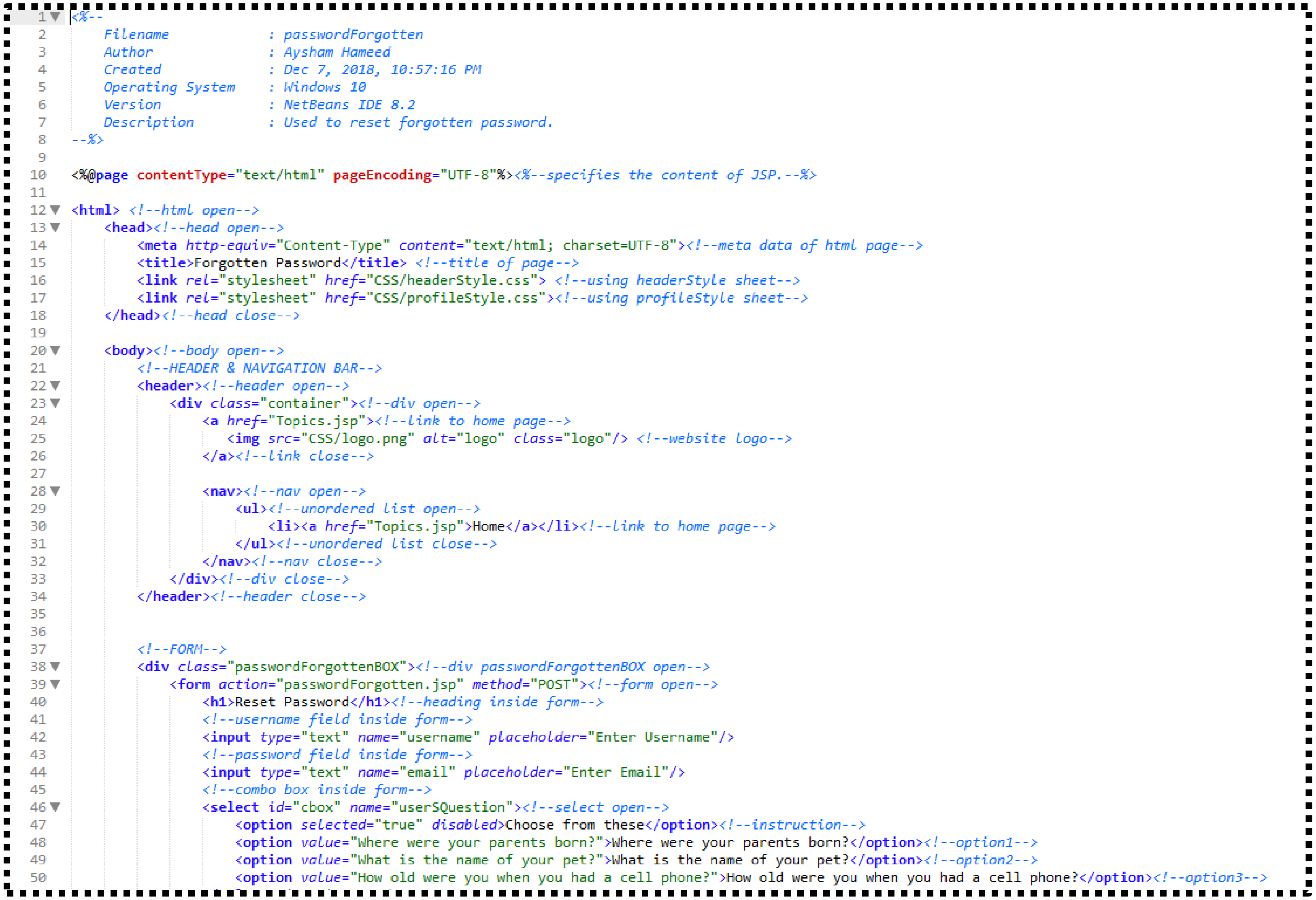
```
    <!--confirm password field-->
    <input type="password" name="userPassword2" placeholder="Confirm Password">
    <!--email field-->
    <input type="text" name="userEmail" placeholder="Email">
    <!--combo box field-->
    <select id="cbox" name="userSQuestion"><!--options-->
        <option selected="true" disabled>Security Questions</option>
            <option value="Where were your parents born?">Where were your parents born?</option><!--option1-->
            <option value="What is the name of your pet?">What is the name of your pet?</option><!--option2-->
            <option value="How old were you when you had a cell phone?">How old were you when you had a cell phone?</option><!--option3-->
    </select><!--options close-->
    <!--answer field-->
    <input type="text" name="userSAnswer" placeholder="Answer">
    <!--join button-->
    <input type="submit" value="Join" >

<!--JSP FUNCTIONALITY-->
<%  //Instantiating web service
    serviceClient.ProcessPosts service = new serviceClient.ProcessPosts();
    serviceClient.Processor port = service.getProcessorPort();


    //If Join button is pressed
    if("POST".equals(request.getMethod())){
        //to build error string
        String errorMessage = "";

        //USERNAME VALIDATION
        String username = request.getParameter("userName");//getting username from form
        if(username.length()>5){//checking username length
            boolean userNameValid = Pattern.matches("\\w+", username);//verifying username
            if(userNameValid==true){//if username is valid
                String password1 = request.getParameter("userPassword1");//get password 1

                //PASSWORD VALIDATION
                if(password1.length()>7){//if password 1 length is greater than 7 then
                    //get password 2 (confirmation)
                    String password2 = request.getParameter("userPassword2")==null? "":request.getParameter("userPassword2");
                    if(password1.equals(password2)){ //if both passwords match (confirmed)
                    String email = request.getParameter("userEmail");//get user email from form

                        //EMAIL VALIDATION
                        if(!email.equals("")){//if email is not blank then
                            //check email format
                            boolean emailValid = Pattern.matches("[\\w]+[@][\\w]+[.][\\w]+", email);
                            if(emailValid==true){//if email is valid
                                //get security question
                                String question = request.getParameter("userSQuestion")==null ? "":request.getParameter("userSQuestion");

                                //QUESTION VALIDATION
```



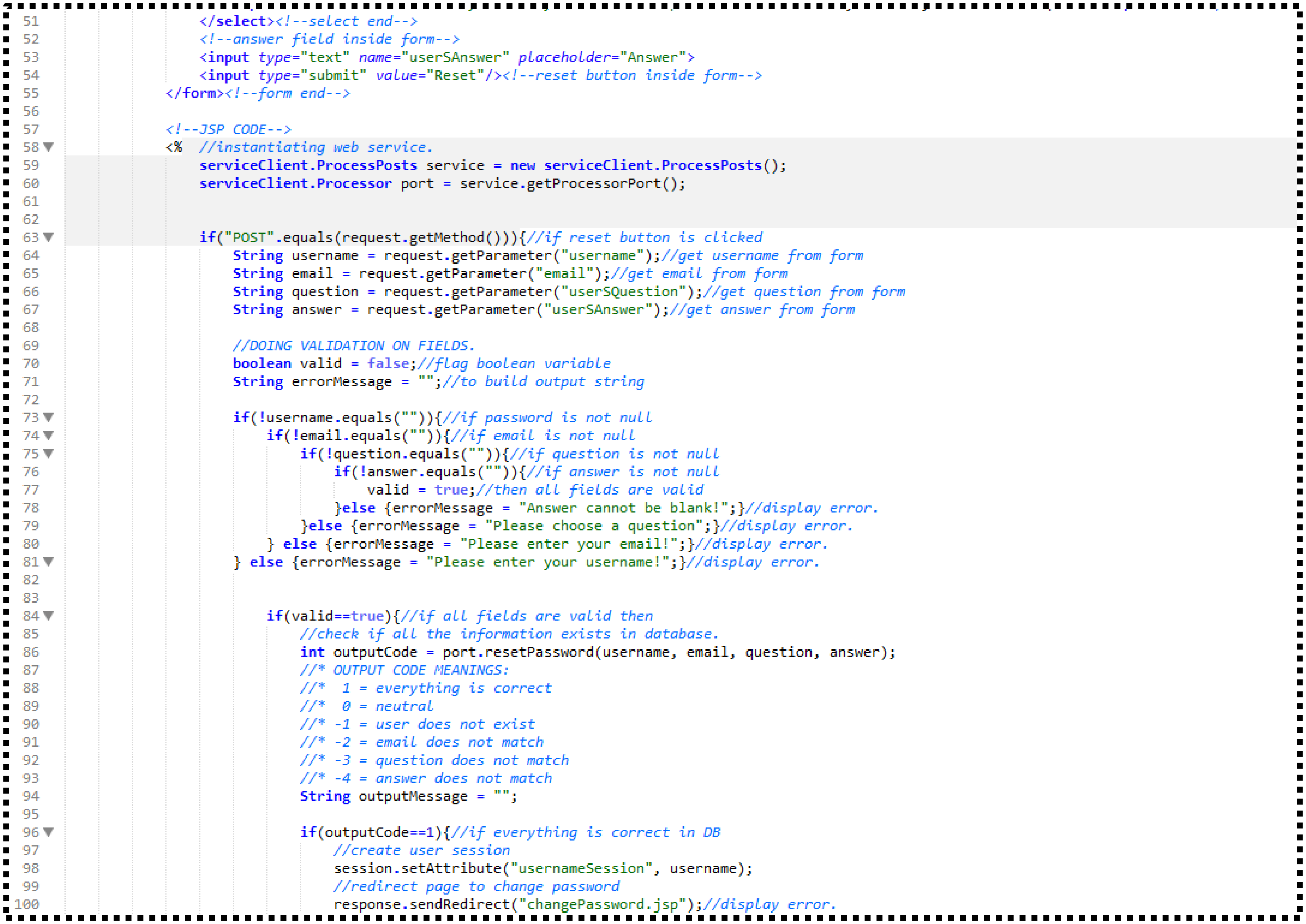
```
                                        if(!question.equals("")){//if question is not blank
                                            //get security answer
                                            String answer = request.getParameter("userSAnswer");

                                            //ANSWER VALIDATION
                                            if(!answer.equals("")){//if answer is not blank

                                                //CHECKING IF USER DETAILS ARE CORRECT
                                                if((port.addUser(username, email,password1, question, answer))==1){
                                                    //THEN ADD TO DATABASE
                                                    port.addUser(username, email , password1, question, answer);
                                                    response.sendRedirect("Login.jsp");//go to login page

                                                }else errorMessage = "User already exists!";//build error message

                                            }else errorMessage = "Please entery a proper answer.";//build error message

                                        }else errorMessage = "Please choose a Security question.";//build error message

                                    }else errorMessage = "Email is invalid.";//build error message

                                }else errorMessage = "Email cannot be blank.";//build error message

                            }else errorMessage = "Passwords dont match.";//build error message

                        }else errorMessage = "Password must at least be 8 characters long.";//build error message

                    } else errorMessage = "Username can only contain letters and numbers.";//build error message

                } else errorMessage = "Username must at least be 6 characters long.";//build error message

                //DISPLAY ERROR MESSAGE!
                out.println("<p style=\"color:white; font-size:11; text-align:center;\"><b>"+errorMessage+"<b><p>");
            }
        %>
    </form><!--form close-->

    </body><!--body close-->
</html><!--html close-->

```

passwordFogotten.jsp

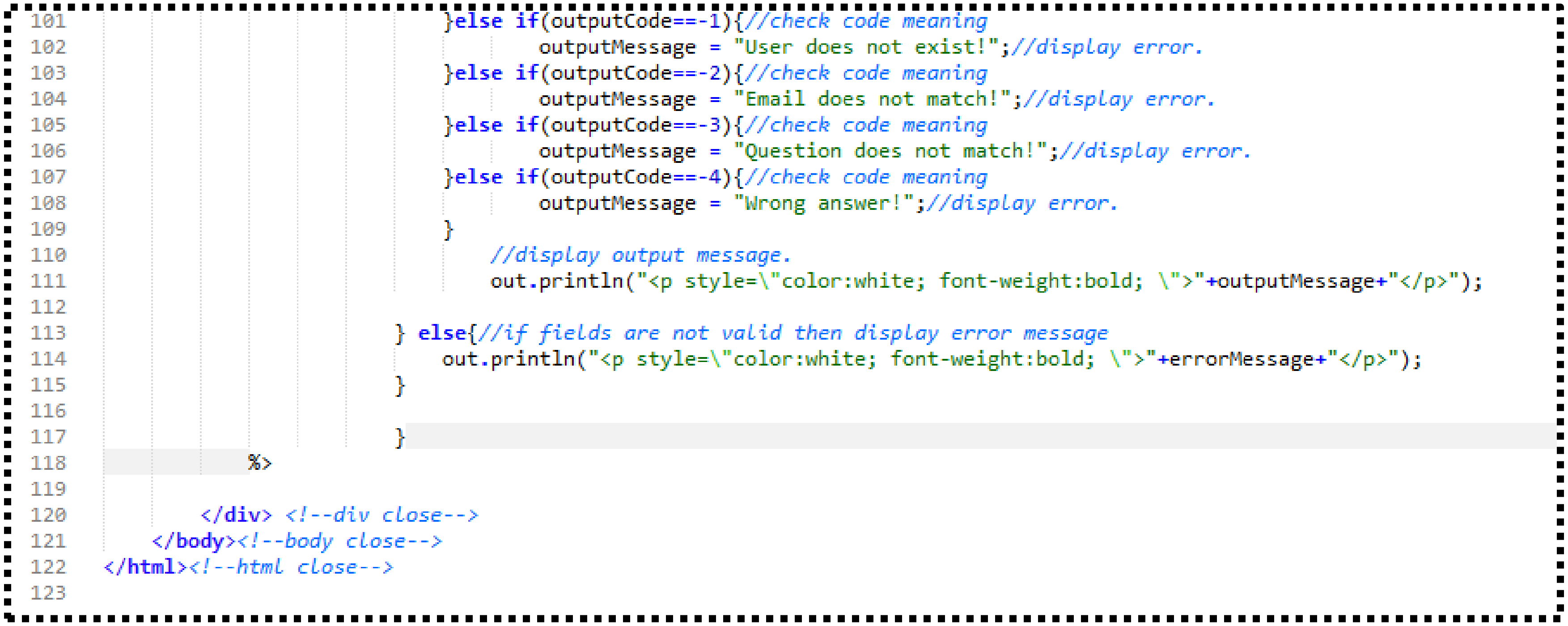
Used to reset forgotten password.



```
<%--
    Filename            : passwordForgotten
    Author              : Aysham Hameed
    Created             : Dec 7, 2018, 10:57:16 PM
    Operating System    : Windows 10
    Version             : NetBeans IDE 8.2
    Description         : Used to reset forgotten password.
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%><%--specifies the content of JSP.--%>

<html> <!--html open-->
    <head><!--head open-->
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"><!--meta data of html page-->
        <title>Forgotten Password</title> <!--title of page-->
        <link rel="stylesheet" href="CSS/headerStyle.css"> <!--using headerStyle sheet-->
        <link rel="stylesheet" href="CSS/profileStyle.css"><!--using profileStyle sheet-->
    </head><!--head close-->

    <body><!--body open-->
        <!--HEADER & NAVIGATION BAR-->
        <header><!--header open-->
            <div class="container"><!--div open-->
                <a href="Topics.jsp"><!--link to home page-->
                   <img src="CSS/logo.png" alt="logo" class="logo"/> <!--website logo-->
                </a><!--link close-->

                <nav><!--nav open-->
                    <ul><!--unordered list open-->
                        <li><a href="Topics.jsp">Home</a></li><!--link to home page-->
                    </ul><!--unordered list close-->
                </nav><!--nav close-->
            </div><!--div close-->
        </header><!--header close-->


        <!--FORM-->
        <div class="passwordForgottenBOX"><!--div passwordForgottenBOX open-->
            <form action="passwordForgotten.jsp" method="POST"><!--form open-->
                <h1>Reset Password</h1><!--heading inside form-->
                <!--username field inside form-->
                <input type="text" name="username" placeholder="Enter Username"/>
                <!--password field inside form-->
                <input type="text" name="email" placeholder="Enter Email"/>
                <!--combo box inside form-->
                <select id="cbox" name="userSQuestion"><!--select open-->
                    <option selected="true" disabled>Choose from these</option><!--instruction-->
                    <option value="Where were your parents born?">Where were your parents born?</option><!--option1-->
                    <option value="What is the name of your pet?">What is the name of your pet?</option><!--option2-->
                    <option value="How old were you when you had a cell phone?">How old were you when you had a cell phone?</option><!--option3-->
```



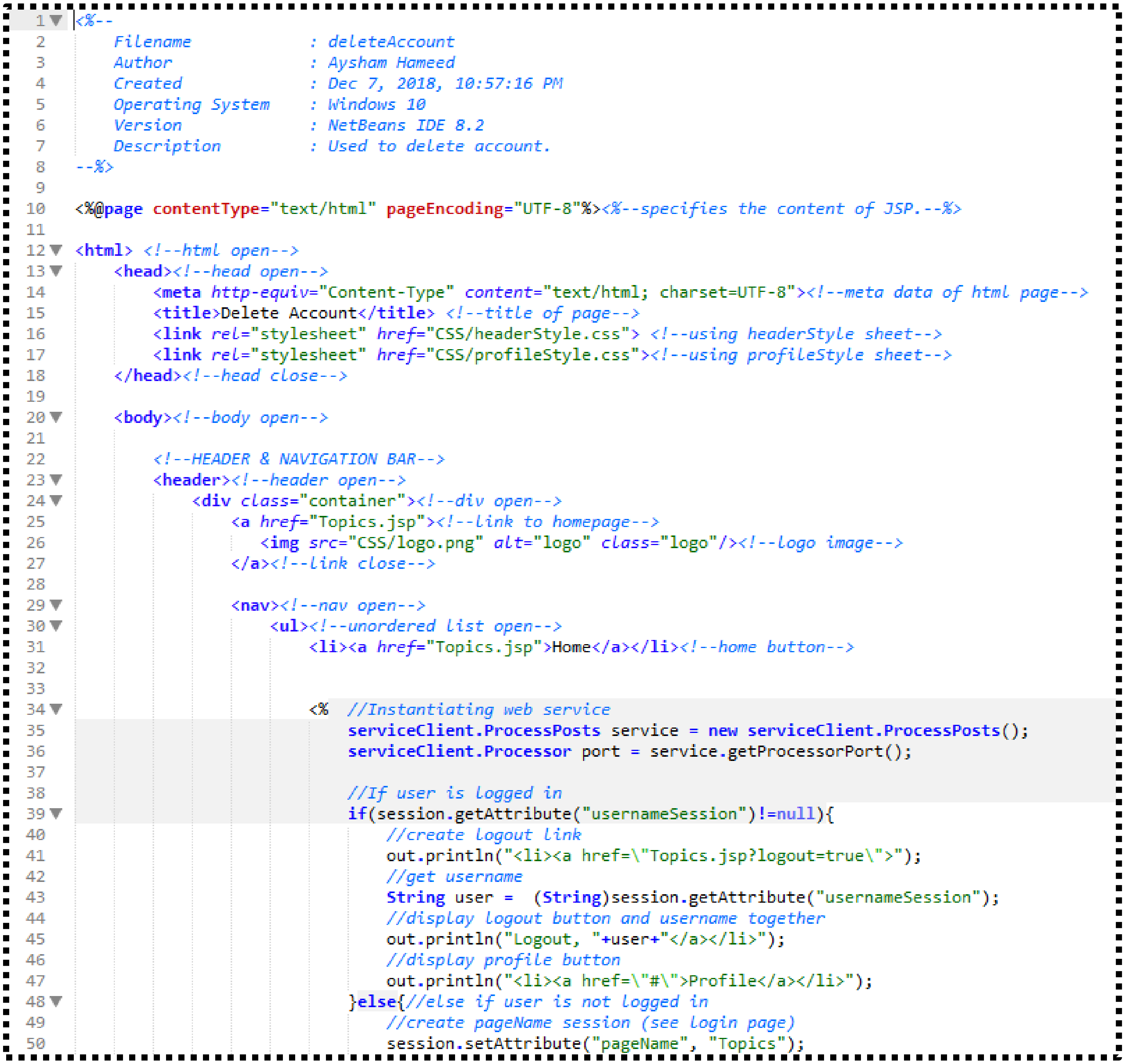
```
                </select><!--select end-->
                <!--answer field inside form-->
                <input type="text" name="userSAnswer" placeholder="Answer">
                <input type="submit" value="Reset"/><!--reset button inside form-->
            </form><!--form end-->

            <!--JSP CODE-->
            <%  //instantiating web service.
                serviceClient.ProcessPosts service = new serviceClient.ProcessPosts();
                serviceClient.Processor port = service.getProcessorPort();


                if("POST".equals(request.getMethod())){//if reset button is clicked
                    String username = request.getParameter("username");//get username from form
                    String email = request.getParameter("email");//get email from form
                    String question = request.getParameter("userSQuestion");//get question from form
                    String answer = request.getParameter("userSAnswer");//get answer from form

                    //DOING VALIDATION ON FIELDS.
                    boolean valid = false;//flag boolean variable
                    String errorMessage = "";//to build output string

                    if(!username.equals("")){//if password is not null
                        if(!email.equals("")){//if email is not null
                            if(!question.equals("")){//if question is not null
                                if(!answer.equals("")){//if answer is not null
                                    valid = true;//then all fields are valid
                                }else {errorMessage = "Answer cannot be blank!";}//display error.
                            }else {errorMessage = "Please choose a question";}//display error.
                        } else {errorMessage = "Please enter your email!";}//display error.
                    } else {errorMessage = "Please enter your username!";}//display error.


                        if(valid==true){//if all fields are valid then
                            //check if all the information exists in database.
                            int outputCode = port.resetPassword(username, email, question, answer);
                            //* OUTPUT CODE MEANINGS:
                            //*  1 = everything is correct
                            //*  0 = neutral
                            //* -1 = user does not exist
                            //* -2 = email does not match
                            //* -3 = question does not match
                            //* -4 = answer does not match
                            String outputMessage = "";

                            if(outputCode==1){//if everything is correct in DB
                                //create user session
                                session.setAttribute("usernameSession", username);
                                //redirect page to change password
                                response.sendRedirect("changePassword.jsp");//display error.
```



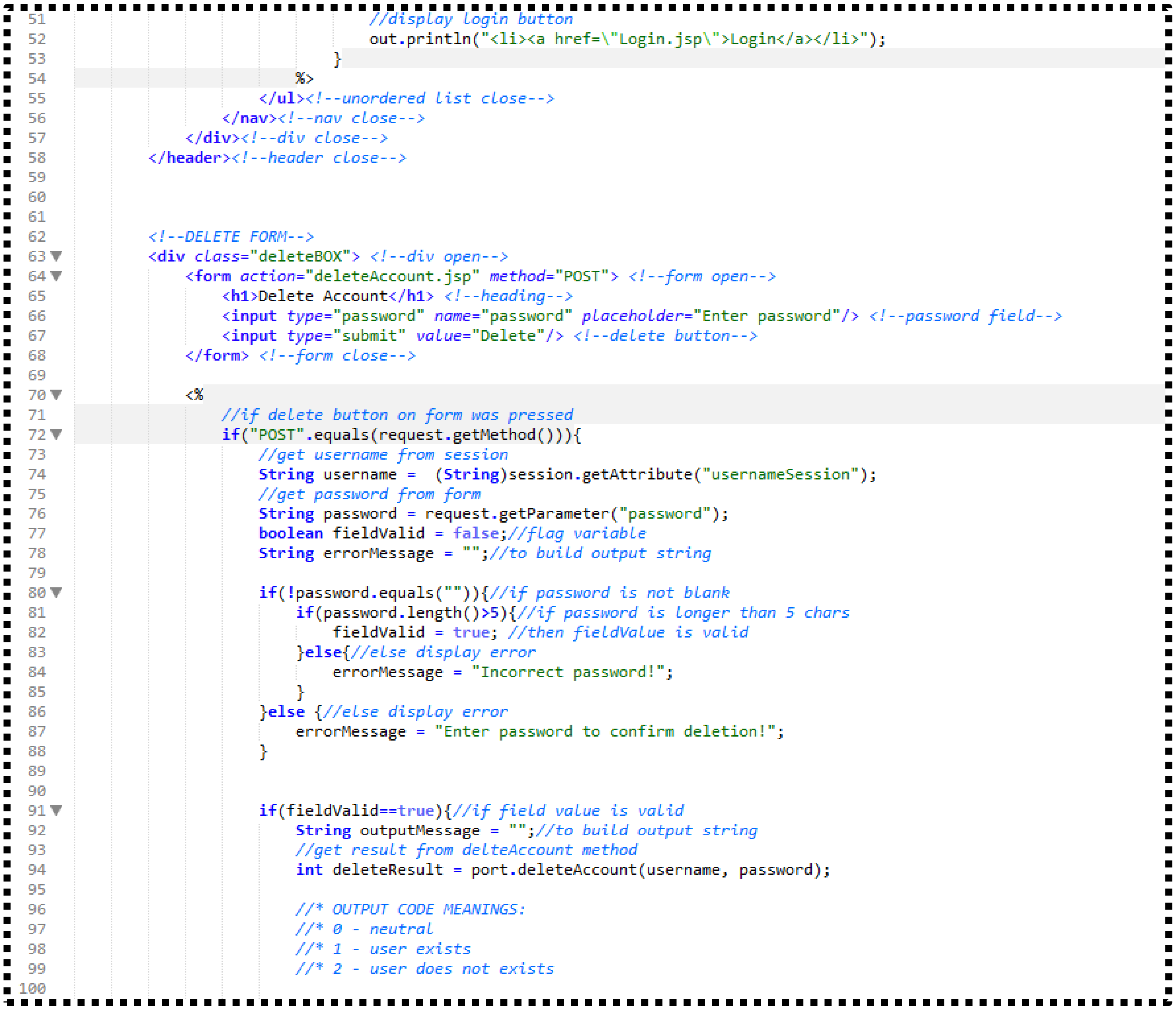
```
                            }else if(outputCode==-1){//check code meaning
                                    outputMessage = "User does not exist!";//display error.
                            }else if(outputCode==-2){//check code meaning
                                    outputMessage = "Email does not match!";//display error.
                            }else if(outputCode==-3){//check code meaning
                                    outputMessage = "Question does not match!";//display error.
                            }else if(outputCode==-4){//check code meaning
                                    outputMessage = "Wrong answer!";//display error.
                            }
                                //display output message.
                                out.println("<p style=\"color:white; font-weight:bold; \">"+outputMessage+"</p>");

                        } else{//if fields are not valid then display error message
                            out.println("<p style=\"color:white; font-weight:bold; \">"+errorMessage+"</p>");
                        }

                        }
            %>

        </div> <!--div close-->
    </body><!--body close-->
</html><!--html close-->

```

deleteAccount.jsp

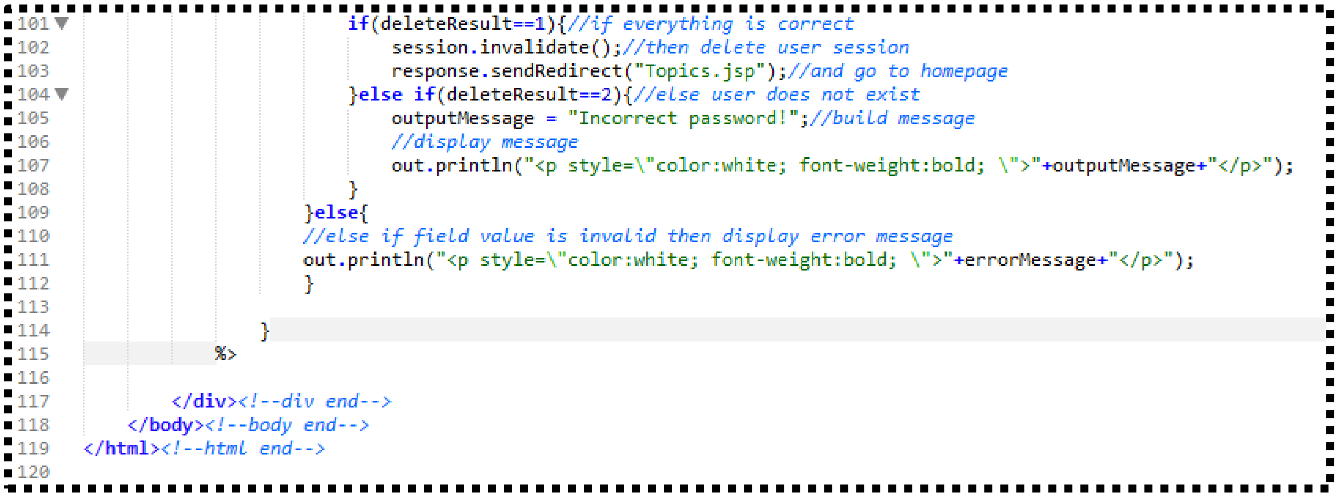
Used to delete account.



```
<%--
    Filename            : deleteAccount
    Author              : Aysham Hameed
    Created             : Dec 7, 2018, 10:57:16 PM
    Operating System    : Windows 10
    Version             : NetBeans IDE 8.2
    Description         : Used to delete account.
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%><%--specifies the content of JSP.--%>

<html> <!--html open-->
    <head><!--head open-->
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"><!--meta data of html page-->
        <title>Delete Account</title> <!--title of page-->
        <link rel="stylesheet" href="CSS/headerStyle.css"> <!--using headerStyle sheet-->
        <link rel="stylesheet" href="CSS/profileStyle.css"><!--using profileStyle sheet-->
    </head><!--head close-->

    <body><!--body open-->

        <!--HEADER & NAVIGATION BAR-->
        <header><!--header open-->
            <div class="container"><!--div open-->
                <a href="Topics.jsp"><!--link to homepage-->
                   <img src="CSS/logo.png" alt="logo" class="logo"/><!--logo image-->
                </a><!--link close-->

                <nav><!--nav open-->
                    <ul><!--unordered list open-->
                        <li><a href="Topics.jsp">Home</a></li><!--home button-->


                        <%  //Instantiating web service
                            serviceClient.ProcessPosts service = new serviceClient.ProcessPosts();
                            serviceClient.Processor port = service.getProcessorPort();

                            //If user is logged in
                            if(session.getAttribute("usernameSession")!=null){
                                //create logout link
                                out.println("<li><a href=\"Topics.jsp?logout=true\">");
                                //get username
                                String user =  (String)session.getAttribute("usernameSession");
                                //display logout button and username together
                                out.println("Logout, "+user+"</a></li>");
                                //display profile button
                                out.println("<li><a href=\"#\">Profile</a></li>");
                            }else{//else if user is not logged in
                                //create pageName session (see login page)
                                session.setAttribute("pageName", "Topics");
```



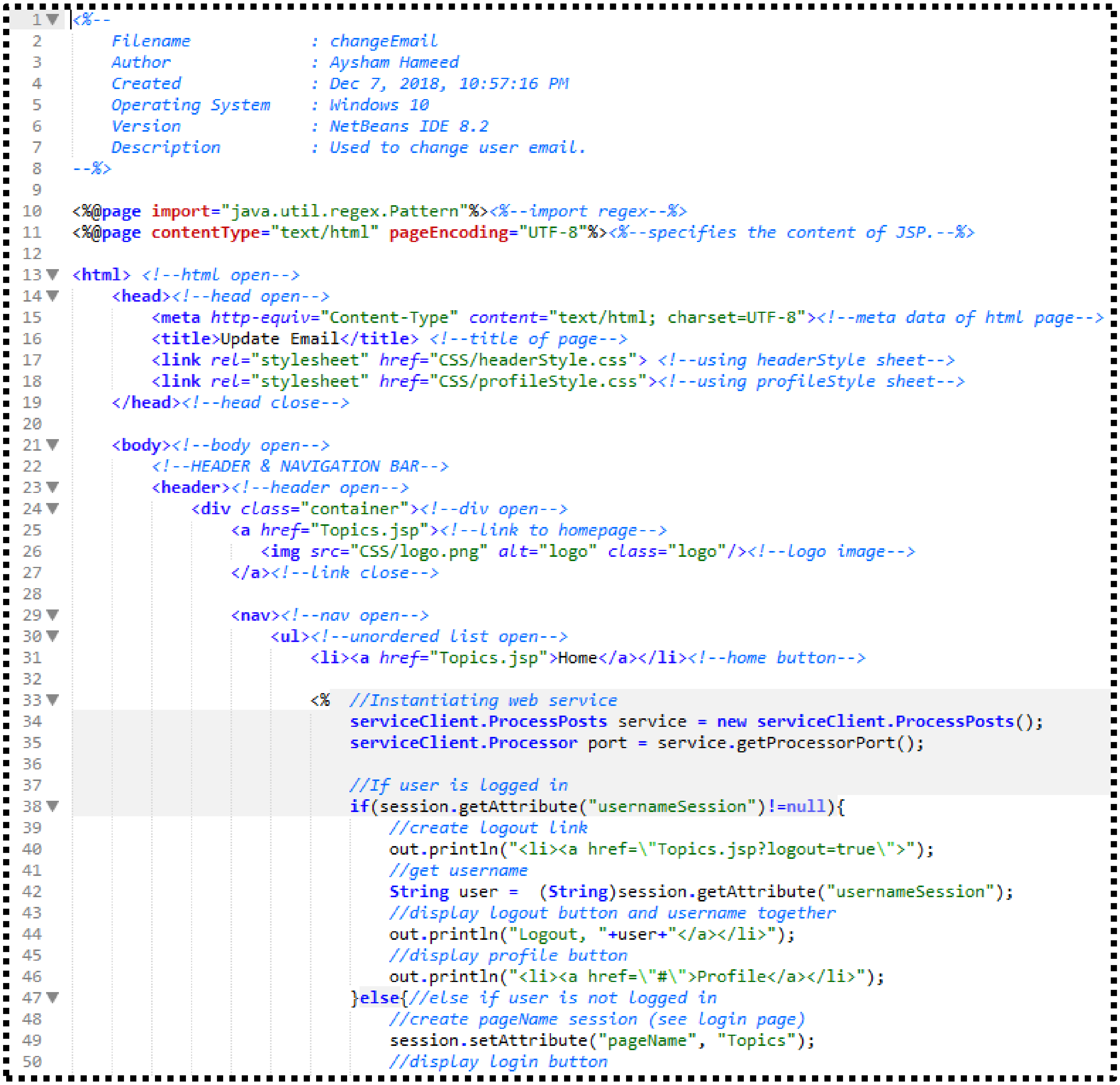
```
                                //display login button
                                out.println("<li><a href=\"Login.jsp\">Login</a></li>");
                            }
                        %>
                    </ul><!--unordered list close-->
                </nav><!--nav close-->
            </div><!--div close-->
        </header><!--header close-->



        <!--DELETE FORM-->
        <div class="deleteBOX"> <!--div open-->
            <form action="deleteAccount.jsp" method="POST"> <!--form open-->
                <h1>Delete Account</h1> <!--heading-->
                <input type="password" name="password" placeholder="Enter password"/> <!--password field-->
                <input type="submit" value="Delete"/> <!--delete button-->
            </form> <!--form close-->

            <%
                //if delete button on form was pressed
                if("POST".equals(request.getMethod())){
                    //get username from session
                    String username =  (String)session.getAttribute("usernameSession");
                    //get password from form
                    String password = request.getParameter("password");
                    boolean fieldValid = false;//flag variable
                    String errorMessage = "";//to build output string

                    if(!password.equals("")){//if password is not blank
                        if(password.length()>5){//if password is longer than 5 chars
                            fieldValid = true; //then fieldValue is valid
                        }else{//else display error
                            errorMessage = "Incorrect password!";
                        }
                    }else {//else display error
                        errorMessage = "Enter password to confirm deletion!";
                    }


                    if(fieldValid==true){//if field value is valid
                        String outputMessage = "";//to build output string
                        //get result from delteAccount method
                        int deleteResult = port.deleteAccount(username, password);

                        //* OUTPUT CODE MEANINGS:
                        //* 0 - neutral
                        //* 1 - user exists
                        //* 2 - user does not exists

```



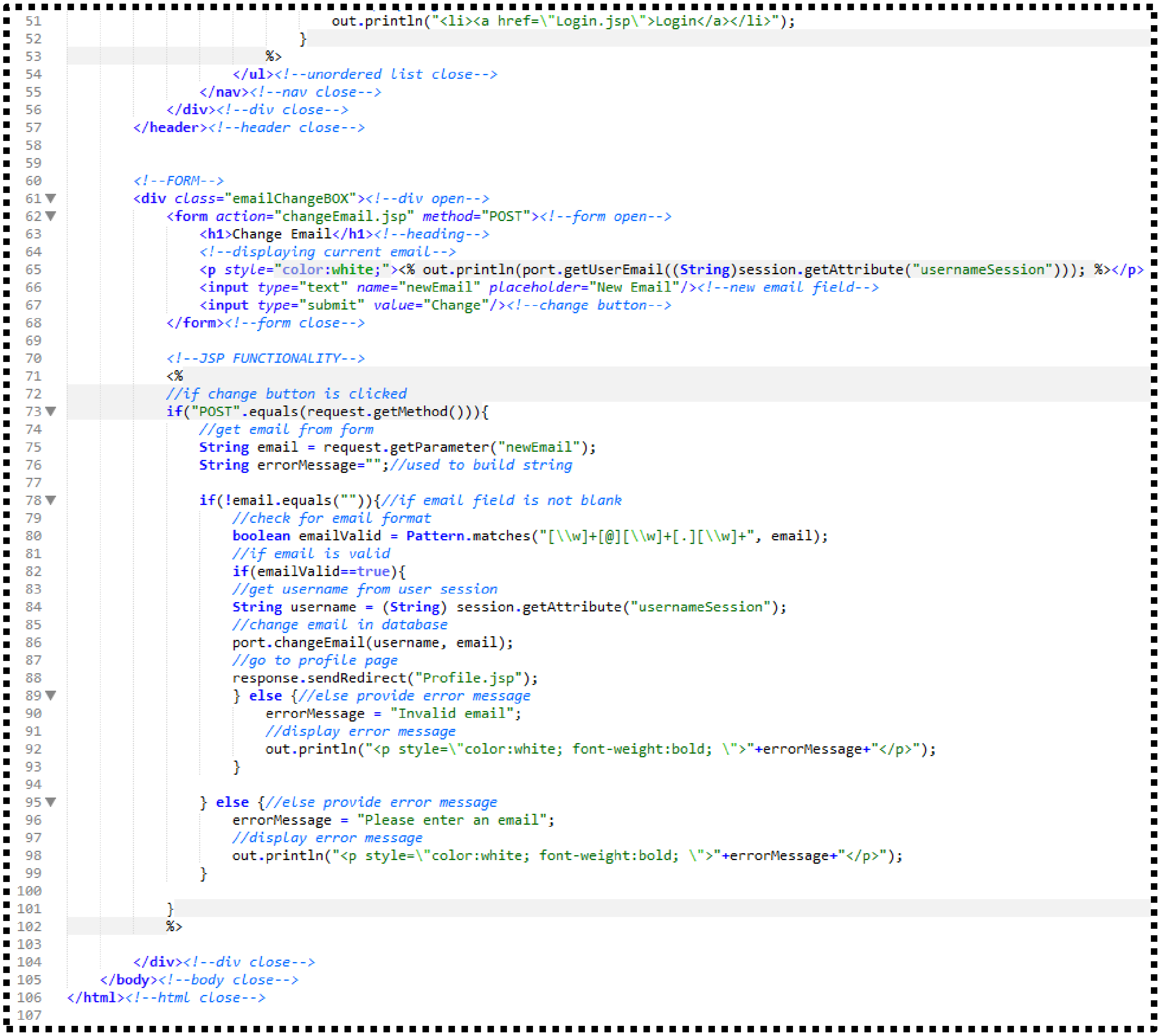
```
                        if(deleteResult==1){//if everything is correct
                            session.invalidate();//then delete user session
                            response.sendRedirect("Topics.jsp");//and go to homepage
                        }else if(deleteResult==2){//else user does not exist
                            outputMessage = "Incorrect password!";//build message
                            //display message
                            out.println("<p style=\"color:white; font-weight:bold; \">"+outputMessage+"</p>");
                        }
                    }else{
                    //else if field value is invalid then display error message
                    out.println("<p style=\"color:white; font-weight:bold; \">"+errorMessage+"</p>");
                    }

                }
            %>

        </div><!--div end-->
    </body><!--body end-->
</html><!--html end-->

```

changeEmail.jsp

Used to change user email.



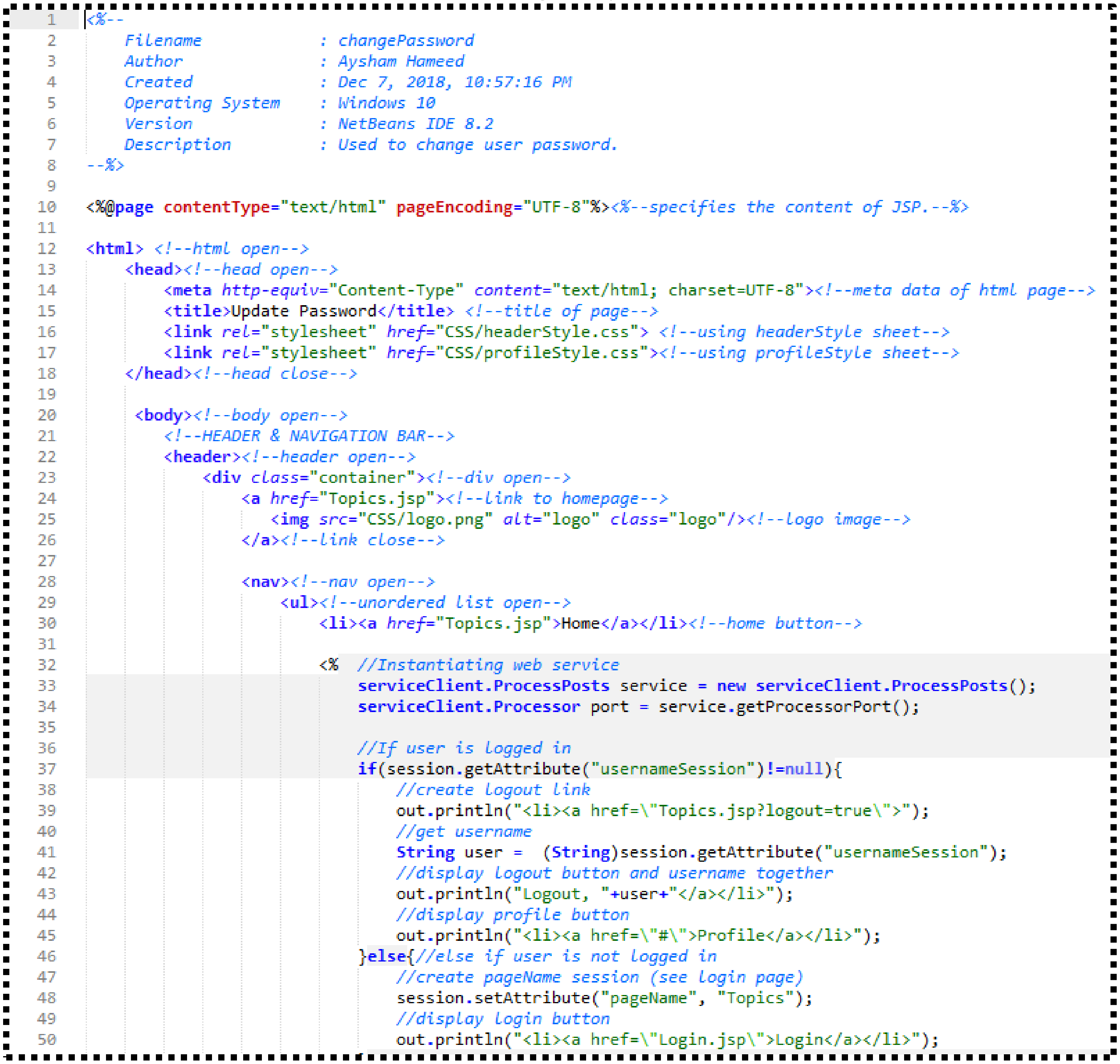
```
<%--
    Filename            : changeEmail
    Author              : Aysham Hameed
    Created             : Dec 7, 2018, 10:57:16 PM
    Operating System    : Windows 10
    Version             : NetBeans IDE 8.2
    Description         : Used to change user email.
--%>

<%@page import="java.util.regex.Pattern"%><%--import regex--%>
<%@page contentType="text/html" pageEncoding="UTF-8"%><%--specifies the content of JSP.--%>

<html> <!--html open-->
    <head><!--head open-->
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"><!--meta data of html page-->
        <title>Update Email</title> <!--title of page-->
        <link rel="stylesheet" href="CSS/headerStyle.css"> <!--using headerStyle sheet-->
        <link rel="stylesheet" href="CSS/profileStyle.css"><!--using profileStyle sheet-->
    </head><!--head close-->

    <body><!--body open-->
        <!--HEADER & NAVIGATION BAR-->
        <header><!--header open-->
            <div class="container"><!--div open-->
                <a href="Topics.jsp"><!--link to homepage-->
                   <img src="CSS/logo.png" alt="logo" class="logo"/><!--logo image-->
                </a><!--link close-->

                <nav><!--nav open-->
                    <ul><!--unordered list open-->
                        <li><a href="Topics.jsp">Home</a></li><!--home button-->

                        <%  //Instantiating web service
                            serviceClient.ProcessPosts service = new serviceClient.ProcessPosts();
                            serviceClient.Processor port = service.getProcessorPort();

                            //If user is logged in
                            if(session.getAttribute("usernameSession")!=null){
                                //create logout link
                                out.println("<li><a href=\"Topics.jsp?logout=true\">");
                                //get username
                                String user =  (String)session.getAttribute("usernameSession");
                                //display logout button and username together
                                out.println("Logout, "+user+"</a></li>");
                                //display profile button
                                out.println("<li><a href=\"#\">Profile</a></li>");
                            }else{//else if user is not logged in
                                //create pageName session (see login page)
                                session.setAttribute("pageName", "Topics");
                                //display login button
```



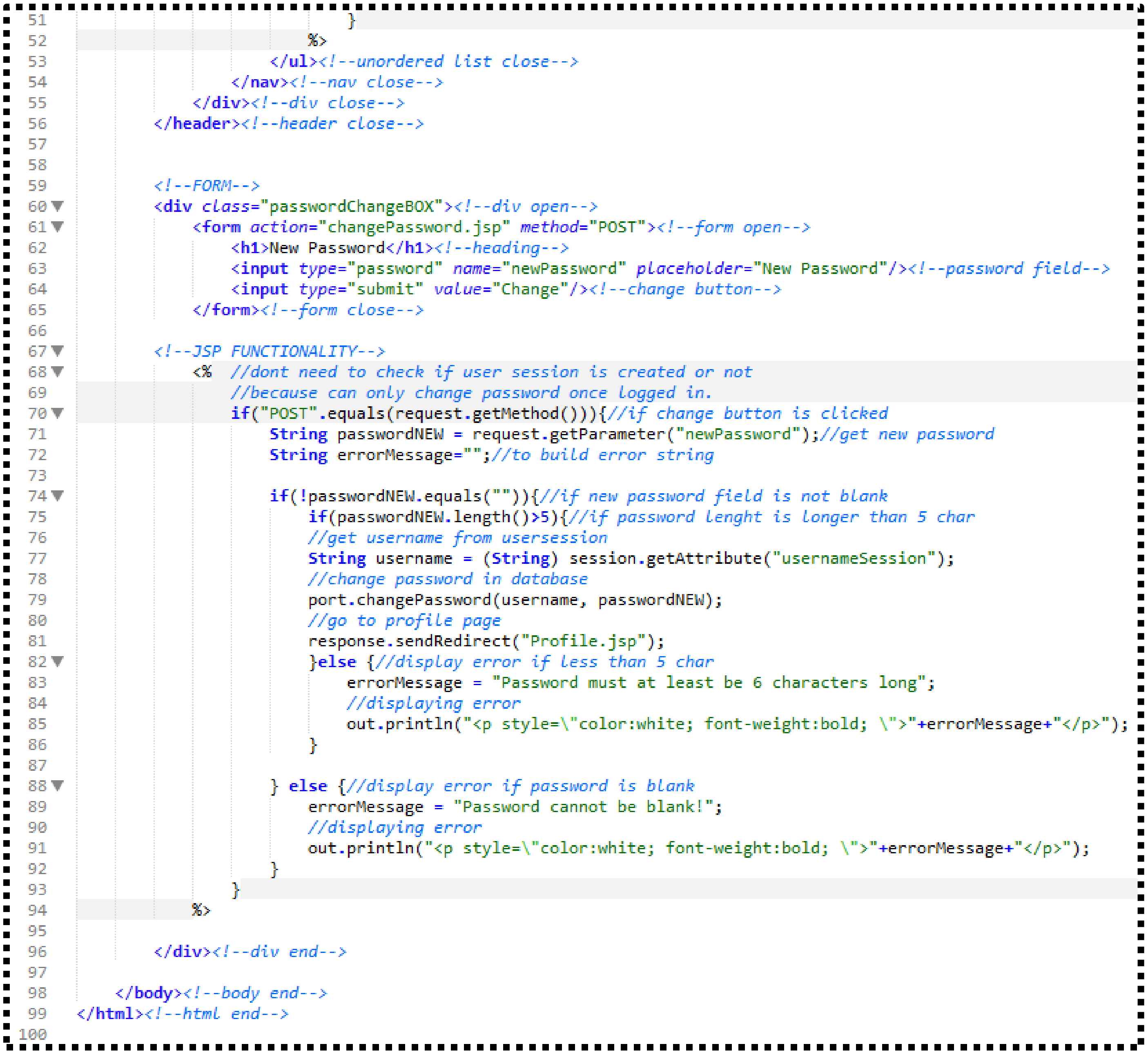
```
                                out.println("<li><a href=\"Login.jsp\">Login</a></li>");
                            }
                        %>
                    </ul><!--unordered list close-->
                </nav><!--nav close-->
            </div><!--div close-->
        </header><!--header close-->


        <!--FORM-->
        <div class="emailChangeBOX"><!--div open-->
            <form action="changeEmail.jsp" method="POST"><!--form open-->
                <h1>Change Email</h1><!--heading-->
                <!--displaying current email-->
                <p style="color:white;"><% out.println(port.getUserEmail((String)session.getAttribute("usernameSession"))); %></p>
                <input type="text" name="newEmail" placeholder="New Email"/><!--new email field-->
                <input type="submit" value="Change"/><!--change button-->
            </form><!--form close-->

            <!--JSP FUNCTIONALITY-->
            <%
            //if change button is clicked
            if("POST".equals(request.getMethod())){
                //get email from form
                String email = request.getParameter("newEmail");
                String errorMessage="";//used to build string

                if(!email.equals("")){//if email field is not blank
                    //check for email format
                    boolean emailValid = Pattern.matches("[\\w]+[@][\\w]+[.][\\w]+", email);
                    //if email is valid
                    if(emailValid==true){
                    //get username from user session
                    String username = (String) session.getAttribute("usernameSession");
                    //change email in database
                    port.changeEmail(username, email);
                    //go to profile page
                    response.sendRedirect("Profile.jsp");
                    } else {//else provide error message
                        errorMessage = "Invalid email";
                        //display error message
                        out.println("<p style=\"color:white; font-weight:bold; \">"+errorMessage+"</p>");
                    }

                } else {//else provide error message
                    errorMessage = "Please enter an email";
                    //display error message
                    out.println("<p style=\"color:white; font-weight:bold; \">"+errorMessage+"</p>");
                }

            }
            %>

        </div><!--div close-->
    </body><!--body close-->
</html><!--html close-->
```

changePassword.jsp

Used to change user password.



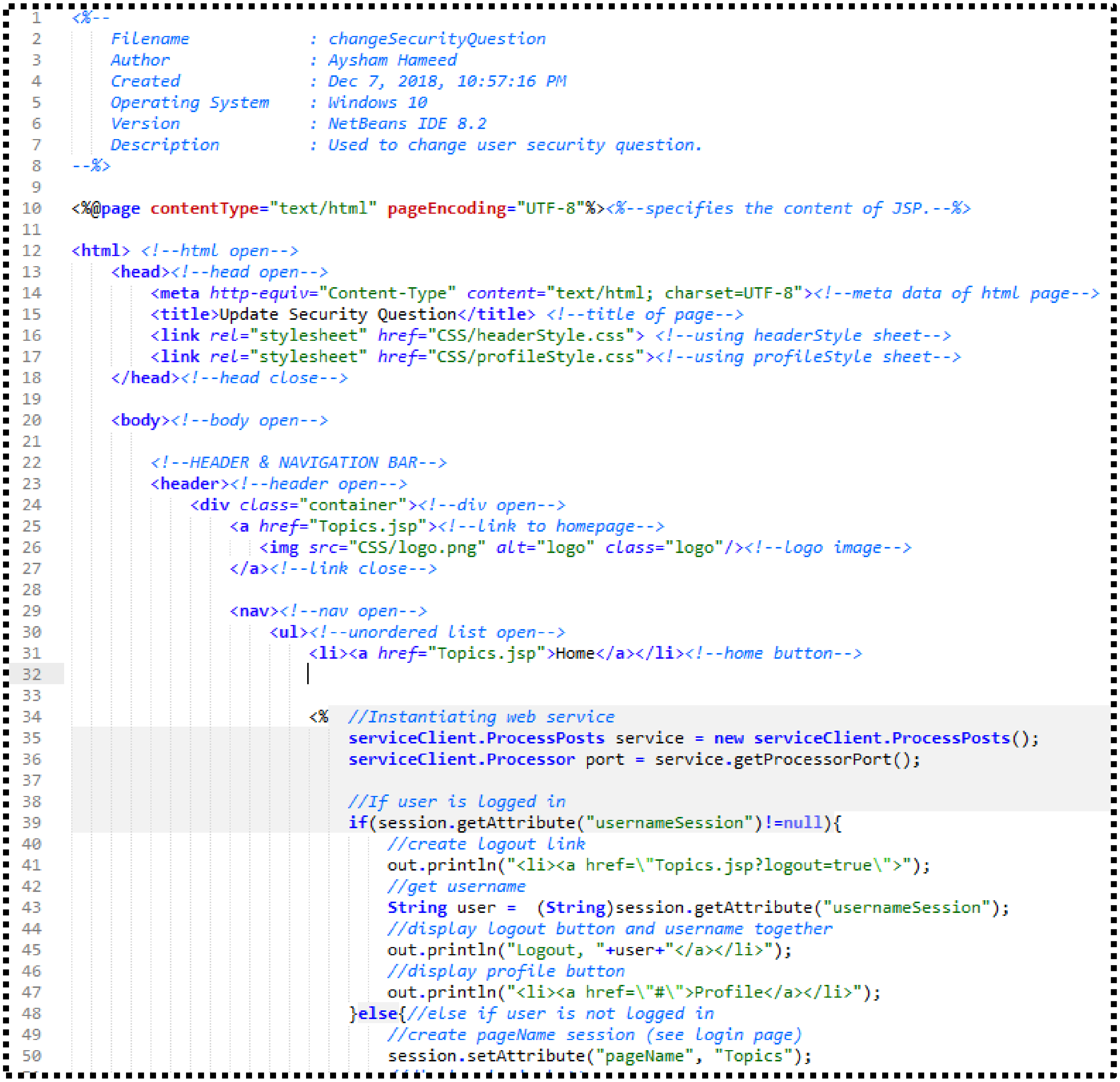
```
<%--
    Filename            : changePassword
    Author              : Aysham Hameed
    Created             : Dec 7, 2018, 10:57:16 PM
    Operating System    : Windows 10
    Version             : NetBeans IDE 8.2
    Description         : Used to change user password.
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%><%--specifies the content of JSP.--%>

<html> <!--html open-->
    <head><!--head open-->
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"><!--meta data of html page-->
        <title>Update Password</title> <!--title of page-->
        <link rel="stylesheet" href="CSS/headerStyle.css"> <!--using headerStyle sheet-->
        <link rel="stylesheet" href="CSS/profileStyle.css"><!--using profileStyle sheet-->
    </head><!--head close-->

     <body><!--body open-->
        <!--HEADER & NAVIGATION BAR-->
        <header><!--header open-->
            <div class="container"><!--div open-->
                <a href="Topics.jsp"><!--link to homepage-->
                   <img src="CSS/logo.png" alt="logo" class="logo"/><!--logo image-->
                </a><!--link close-->

                <nav><!--nav open-->
                    <ul><!--unordered list open-->
                        <li><a href="Topics.jsp">Home</a></li><!--home button-->

                        <%  //Instantiating web service
                            serviceClient.ProcessPosts service = new serviceClient.ProcessPosts();
                            serviceClient.Processor port = service.getProcessorPort();

                            //If user is logged in
                            if(session.getAttribute("usernameSession")!=null){
                                //create logout link
                                out.println("<li><a href=\"Topics.jsp?logout=true\">");
                                //get username
                                String user =  (String)session.getAttribute("usernameSession");
                                //display logout button and username together
                                out.println("Logout, "+user+"</a></li>");
                                //display profile button
                                out.println("<li><a href=\"#\">Profile</a></li>");
                            }else{//else if user is not logged in
                                //create pageName session (see login page)
                                session.setAttribute("pageName", "Topics");
                                //display login button
                                out.println("<li><a href=\"Login.jsp\">Login</a></li>");
```



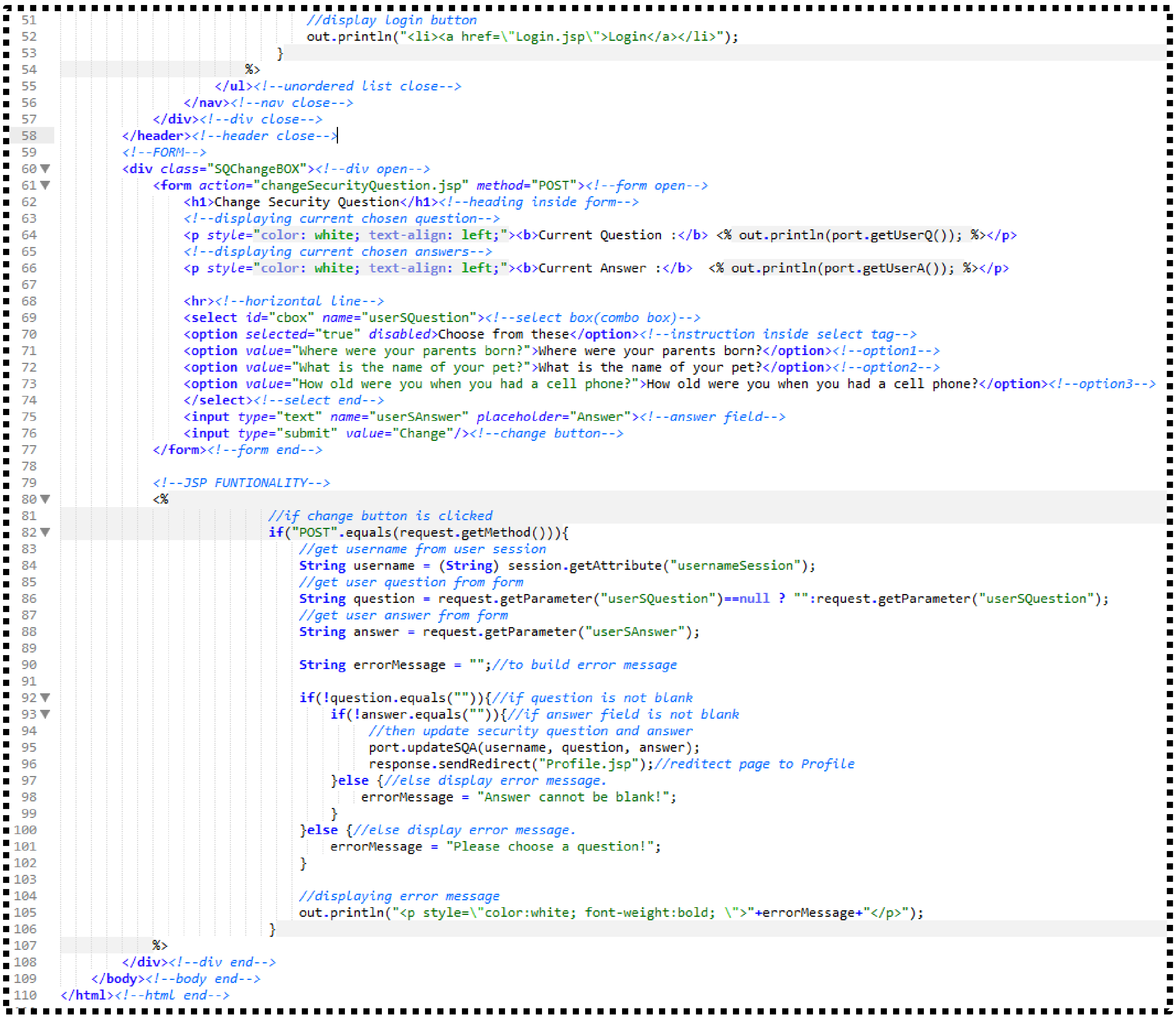
```
                            }
                        %>
                    </ul><!--unordered list close-->
                </nav><!--nav close-->
            </div><!--div close-->
        </header><!--header close-->


        <!--FORM-->
        <div class="passwordChangeBOX"><!--div open-->
            <form action="changePassword.jsp" method="POST"><!--form open-->
                <h1>New Password</h1><!--heading-->
                <input type="password" name="newPassword" placeholder="New Password"/><!--password field-->
                <input type="submit" value="Change"/><!--change button-->
            </form><!--form close-->

        <!--JSP FUNCTIONALITY-->
            <%  //dont need to check if user session is created or not
                //because can only change password once logged in.
                if("POST".equals(request.getMethod())){//if change button is clicked
                    String passwordNEW = request.getParameter("newPassword");//get new password
                    String errorMessage="";//to build error string

                    if(!passwordNEW.equals("")){//if new password field is not blank
                        if(passwordNEW.length()>5){//if password lenght is longer than 5 char
                        //get username from usersession
                        String username = (String) session.getAttribute("usernameSession");
                        //change password in database
                        port.changePassword(username, passwordNEW);
                        //go to profile page
                        response.sendRedirect("Profile.jsp");
                        }else {//display error if less than 5 char
                            errorMessage = "Password must at least be 6 characters long";
                            //displaying error
                            out.println("<p style=\"color:white; font-weight:bold; \">"+errorMessage+"</p>");
                        }

                    } else {//display error if password is blank
                        errorMessage = "Password cannot be blank!";
                        //displaying error
                        out.println("<p style=\"color:white; font-weight:bold; \">"+errorMessage+"</p>");
                    }
                }
            %>

        </div><!--div end-->

    </body><!--body end-->
</html><!--html end-->

```

changeSecurityQuestion.jsp

Used to change user security question.



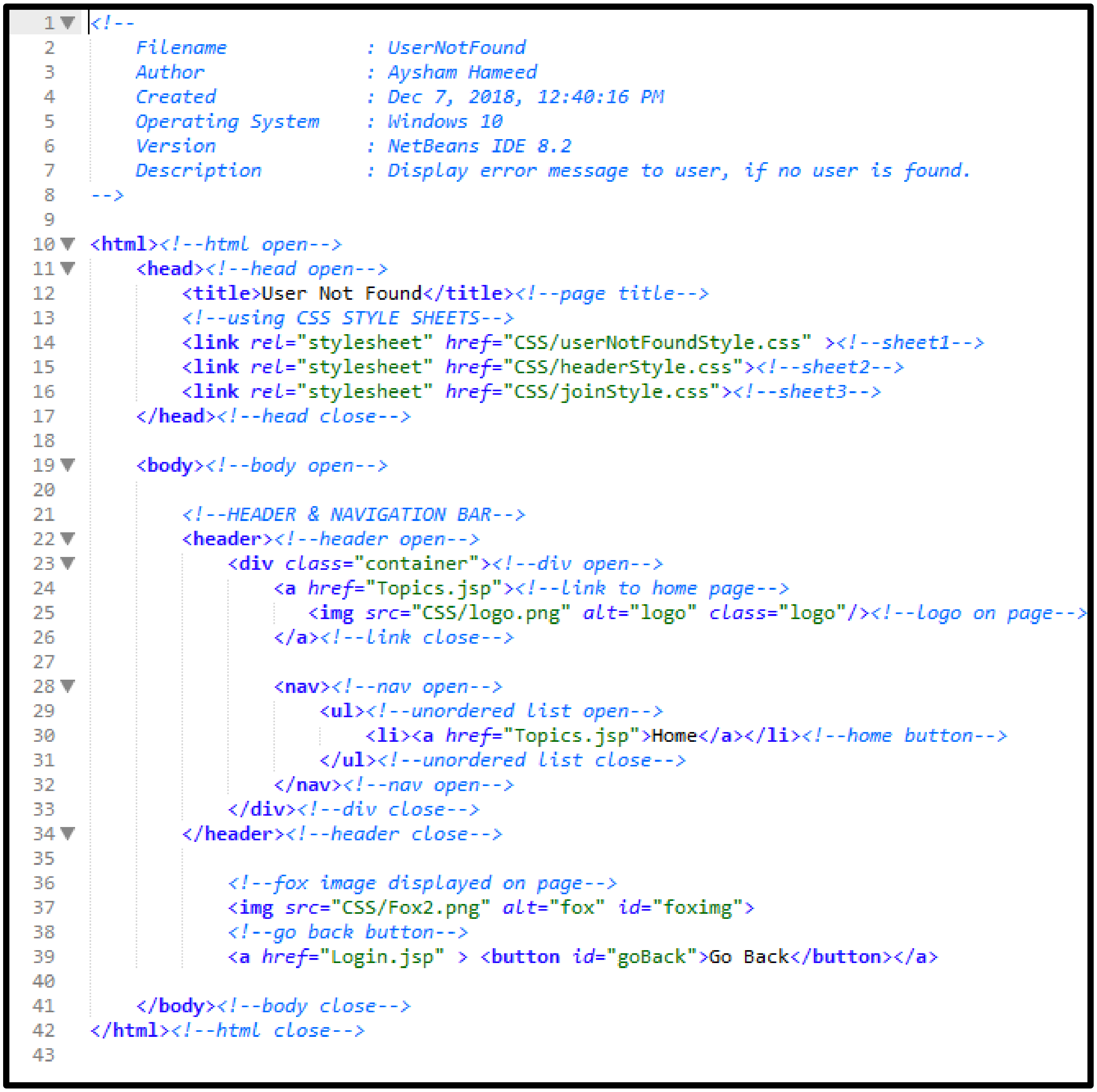
```
<%--
    Filename                : changeSecurityQuestion
    Author                  : Aysham Hameed
    Created                 : Dec 7, 2018, 10:57:16 PM
    Operating System        : Windows 10
    Version                 : NetBeans IDE 8.2
    Description             : Used to change user security question.
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%><%--specifies the content of JSP.--%>

<html> <!--html open-->
    <head><!--head open-->
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"><!--meta data of html page-->
        <title>Update Security Question</title> <!--title of page-->
        <link rel="stylesheet" href="CSS/headerStyle.css"> <!--using headerStyle sheet-->
        <link rel="stylesheet" href="CSS/profileStyle.css"><!--using profileStyle sheet-->
    </head><!--head close-->

    <body><!--body open-->

        <!--HEADER & NAVIGATION BAR-->
        <header><!--header open-->
            <div class="container"><!--div open-->
                <a href="Topics.jsp"><!--link to homepage-->
                  <img src="CSS/logo.png" alt="logo" class="logo"/><!--logo image-->
                </a><!--link close-->

                <nav><!--nav open-->
                    <ul><!--unordered list open-->
                        <li><a href="Topics.jsp">Home</a></li><!--home button-->


                        <%  //Instantiating web service
                            serviceClient.ProcessPosts service = new serviceClient.ProcessPosts();
                            serviceClient.Processor port = service.getProcessorPort();

                            //If user is logged in
                            if(session.getAttribute("usernameSession")!=null){
                                //create logout link
                                out.println("<li><a href=\"Topics.jsp?logout=true\">");
                                //get username
                                String user =  (String)session.getAttribute("usernameSession");
                                //display logout button and username together
                                out.println("Logout, "+user+"</a></li>");
                                //display profile button
                                out.println("<li><a href=\"#\">Profile</a></li>");
                            }else{//else if user is not logged in
                                //create pageName session (see login page)
                                session.setAttribute("pageName", "Topics");
```



```
                                //display login button
                                out.println("<li><a href=\"Login.jsp\">Login</a></li>");
                            }
                        %>
                    </ul><!--unordered list close-->
                </nav><!--nav close-->
            </div><!--div close-->
        </header><!--header close-->
        <!--FORM-->
        <div class="SQChangeBOX"><!--div open-->
            <form action="changeSecurityQuestion.jsp" method="POST"><!--form open-->
                <h1>Change Security Question</h1><!--heading inside form-->
                <!--displaying current chosen question-->
                <p style="color: white; text-align: left;"><b>Current Question :</b> <% out.println(port.getUserQ()); %></p>
                <!--displaying current chosen answers-->
                <p style="color: white; text-align: left;"><b>Current Answer :</b>  <% out.println(port.getUserA()); %></p>

                <hr><!--horizontal line-->
                <select id="cbox" name="userSQuestion"><!--select box(combo box)-->
                <option selected="true" disabled>Choose from these</option><!--instruction inside select tag-->
                <option value="Where were your parents born?">Where were your parents born?</option><!--option1-->
                <option value="What is the name of your pet?">What is the name of your pet?</option><!--option2-->
                <option value="How old were you when you had a cell phone?">How old were you when you had a cell phone?</option><!--option3-->
                </select><!--select end-->
                <input type="text" name="userSAnswer" placeholder="Answer"><!--answer field-->
                <input type="submit" value="Change"/><!--change button-->
            </form><!--form end-->

            <!--JSP FUNTIONALITY-->
            <%
                            //if change button is clicked
                            if("POST".equals(request.getMethod())){
                                //get username from user session
                                String username = (String) session.getAttribute("usernameSession");
                                //get user question from form
                                String question = request.getParameter("userSQuestion")==null ? "":request.getParameter("userSQuestion");
                                //get user answer from form
                                String answer = request.getParameter("userSAnswer");

                                String errorMessage = "";//to build error message

                                if(!question.equals("")){//if question is not blank
                                    if(!answer.equals("")){//if answer field is not blank
                                         //then update security question and answer
                                         port.updateSQA(username, question, answer);
                                         response.sendRedirect("Profile.jsp");//reditect page to Profile
                                    }else {//else display error message.
                                        errorMessage = "Answer cannot be blank!";
                                    }
                                }else {//else display error message.
                                    errorMessage = "Please choose a question!";
                                }

                                //displaying error message
                                out.println("<p style=\"color:white; font-weight:bold; \">"+errorMessage+"</p>");
                            }
            %>
        </div><!--div end-->
    </body><!--body end-->
</html><!--html end-->
```

UserNotFound.html

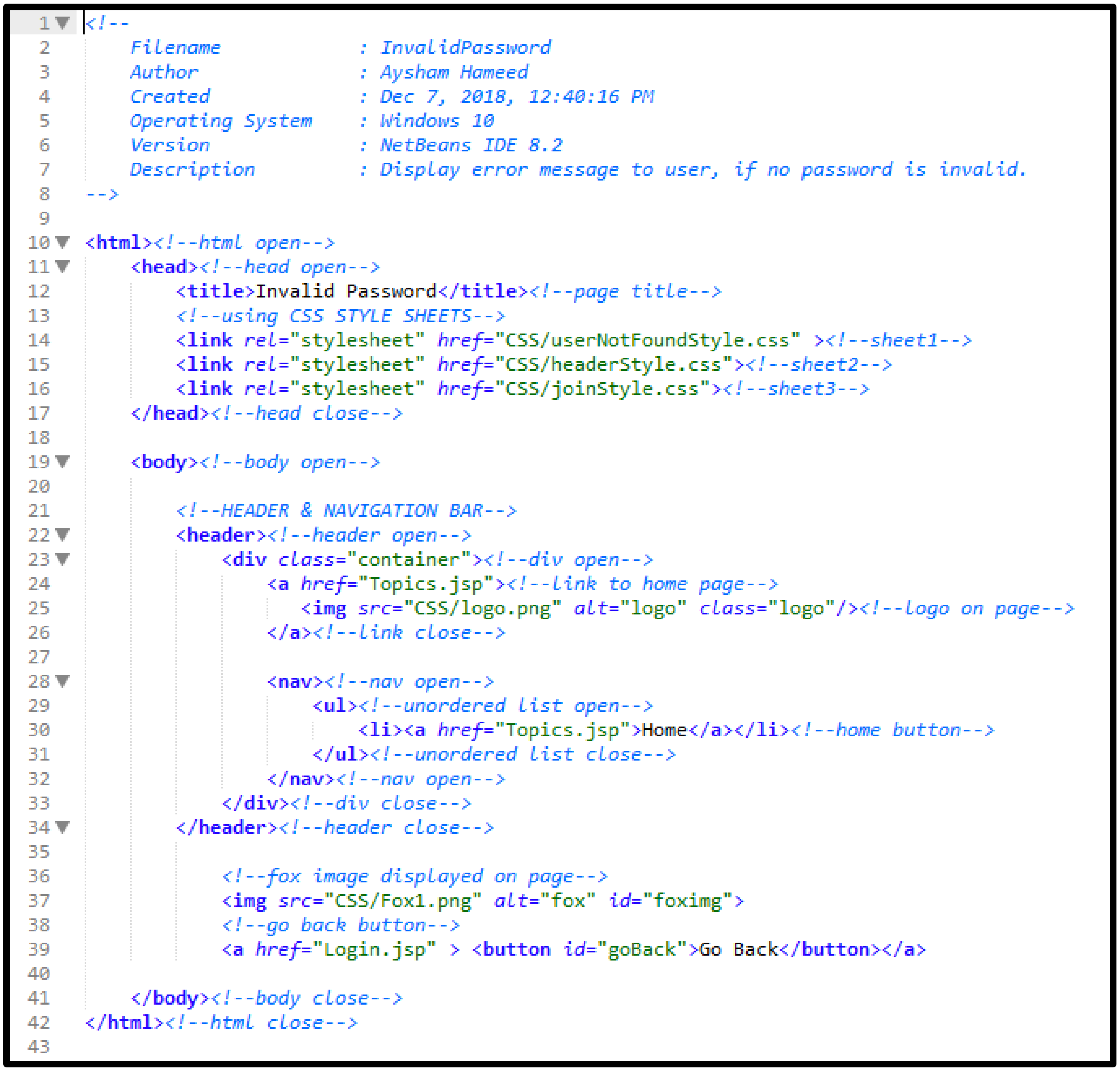
Displays error message to user, if no user is found.



```html
<!--
    Filename            : UserNotFound
    Author              : Aysham Hameed
    Created             : Dec 7, 2018, 12:40:16 PM
    Operating System    : Windows 10
    Version             : NetBeans IDE 8.2
    Description         : Display error message to user, if no user is found.
-->

<html><!--html open-->
    <head><!--head open-->
        <title>User Not Found</title><!--page title-->
        <!--using CSS STYLE SHEETS-->
        <link rel="stylesheet" href="CSS/userNotFoundStyle.css" ><!--sheet1-->
        <link rel="stylesheet" href="CSS/headerStyle.css"><!--sheet2-->
        <link rel="stylesheet" href="CSS/joinStyle.css"><!--sheet3-->
    </head><!--head close-->

    <body><!--body open-->

        <!--HEADER & NAVIGATION BAR-->
        <header><!--header open-->
            <div class="container"><!--div open-->
                <a href="Topics.jsp"><!--link to home page-->
                   <img src="CSS/logo.png" alt="logo" class="logo"/><!--logo on page-->
                </a><!--link close-->

                <nav><!--nav open-->
                    <ul><!--unordered list open-->
                        <li><a href="Topics.jsp">Home</a></li><!--home button-->
                    </ul><!--unordered list close-->
                </nav><!--nav open-->
            </div><!--div close-->
        </header><!--header close-->

            <!--fox image displayed on page-->
            <img src="CSS/Fox2.png" alt="fox" id="foximg">
            <!--go back button-->
            <a href="Login.jsp" > <button id="goBack">Go Back</button></a>

    </body><!--body close-->
</html><!--html close-->

```

InvalidPassword.html

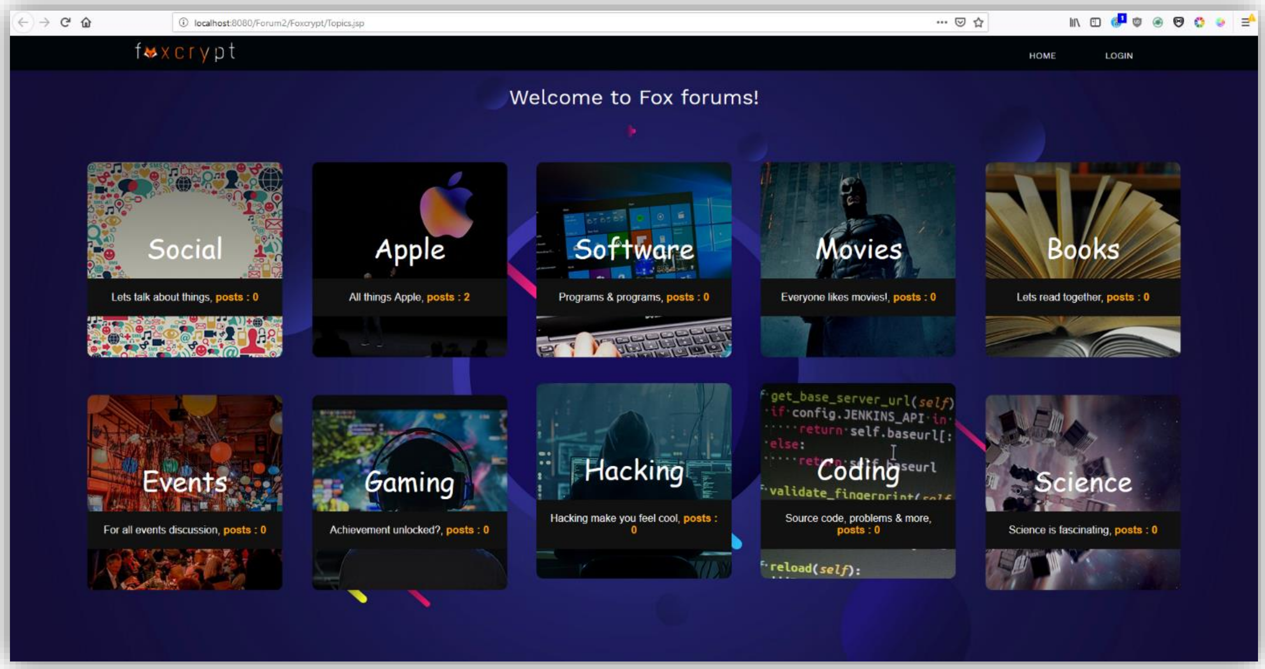
Displays error message to user, if password is wrong.



```html
<!--
    Filename            : InvalidPassword
    Author              : Aysham Hameed
    Created             : Dec 7, 2018, 12:40:16 PM
    Operating System    : Windows 10
    Version             : NetBeans IDE 8.2
    Description         : Display error message to user, if no password is invalid.
-->

<html><!--html open-->
    <head><!--head open-->
        <title>Invalid Password</title><!--page title-->
        <!--using CSS STYLE SHEETS-->
        <link rel="stylesheet" href="CSS/userNotFoundStyle.css" ><!--sheet1-->
        <link rel="stylesheet" href="CSS/headerStyle.css"><!--sheet2-->
        <link rel="stylesheet" href="CSS/joinStyle.css"><!--sheet3-->
    </head><!--head close-->

    <body><!--body open-->

        <!--HEADER & NAVIGATION BAR-->
        <header><!--header open-->
            <div class="container"><!--div open-->
                <a href="Topics.jsp"><!--link to home page-->
                   <img src="CSS/logo.png" alt="logo" class="logo"/><!--logo on page-->
                </a><!--link close-->

                <nav><!--nav open-->
                    <ul><!--unordered list open-->
                        <li><a href="Topics.jsp">Home</a></li><!--home button-->
                    </ul><!--unordered list close-->
                </nav><!--nav open-->
            </div><!--div close-->
        </header><!--header close-->

            <!--fox image displayed on page-->
            <img src="CSS/Fox1.png" alt="fox" id="foximg">
            <!--go back button-->
            <a href="Login.jsp" > <button id="goBack">Go Back</button></a>

    </body><!--body close-->
</html><!--html close-->

```

Screenshots

This is the main page (Homepage) and it contains all the categories/to fixed topics of the forum. (Cannot be changed by user)



Welcome to Fox forums!

Social — Lets talk about things, posts : 0

Apple — All things Apple, posts : 2

Software — Programs & programs, posts : 0

Movies — Everyone likes movies!, posts : 0

Books — Lets read together, posts : 0

Events — For all events discussion, posts : 0

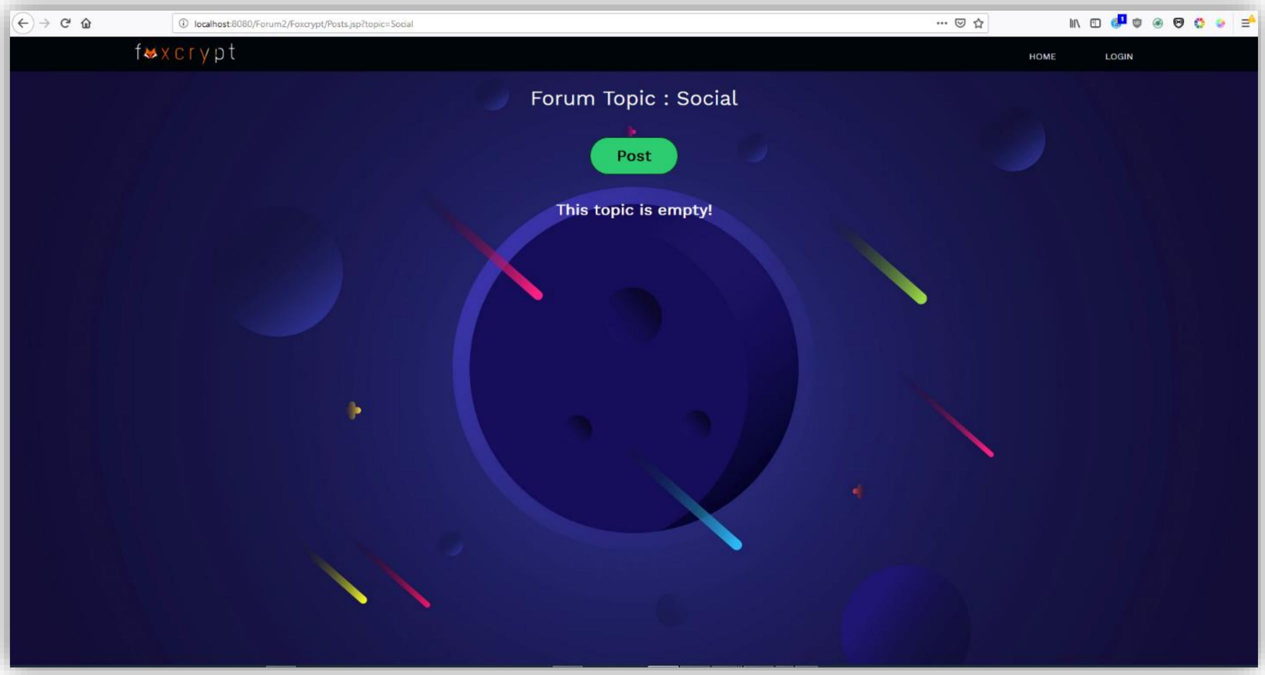
Gaming — Achievement unlocked?, posts : 0

Hacking — Hacking make you feel cool, posts : 0

Coding — Source code, problems & more, posts : 0

Science — Science is fascinating, posts : 0

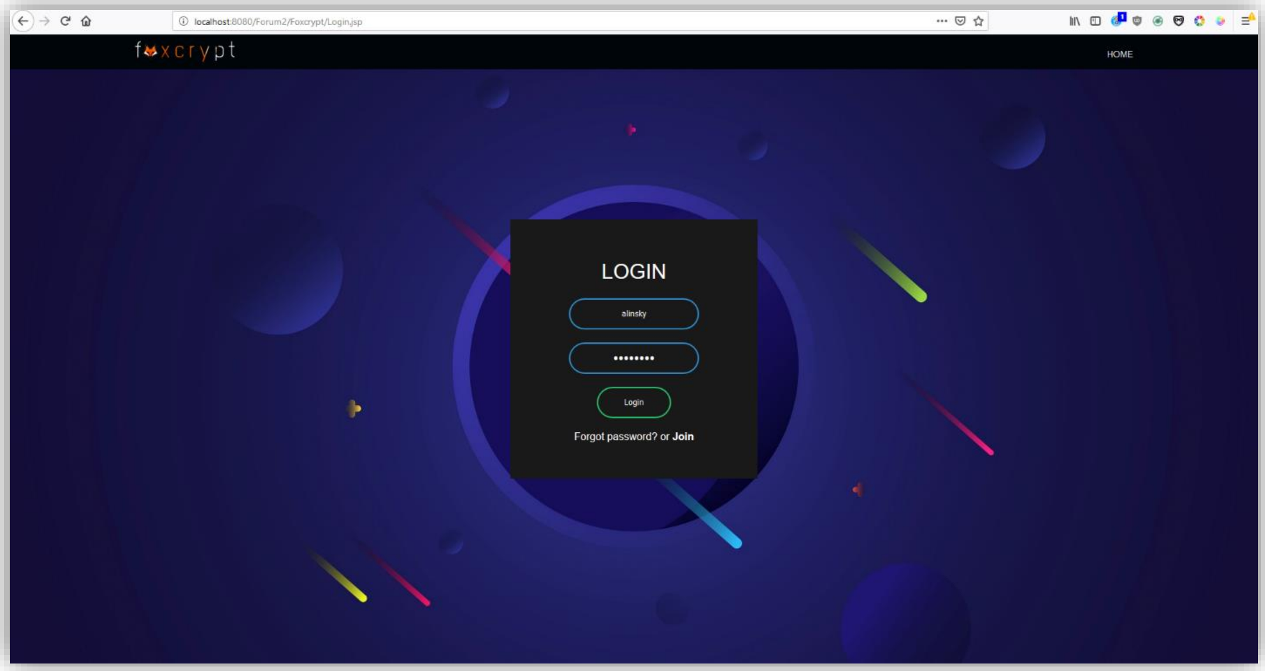
Posts page. Where the user creates a post. But need to login first in order to create it. But can view the posts posted by other users.



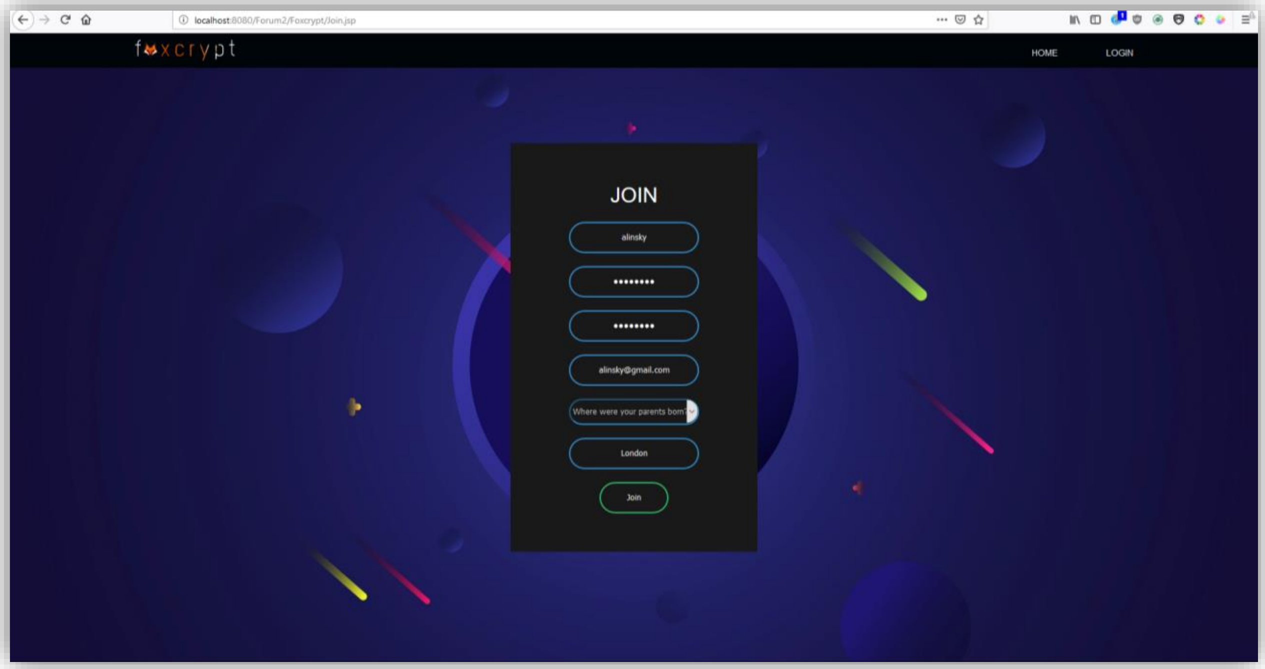
Forum Topic : Social

This topic is empty!

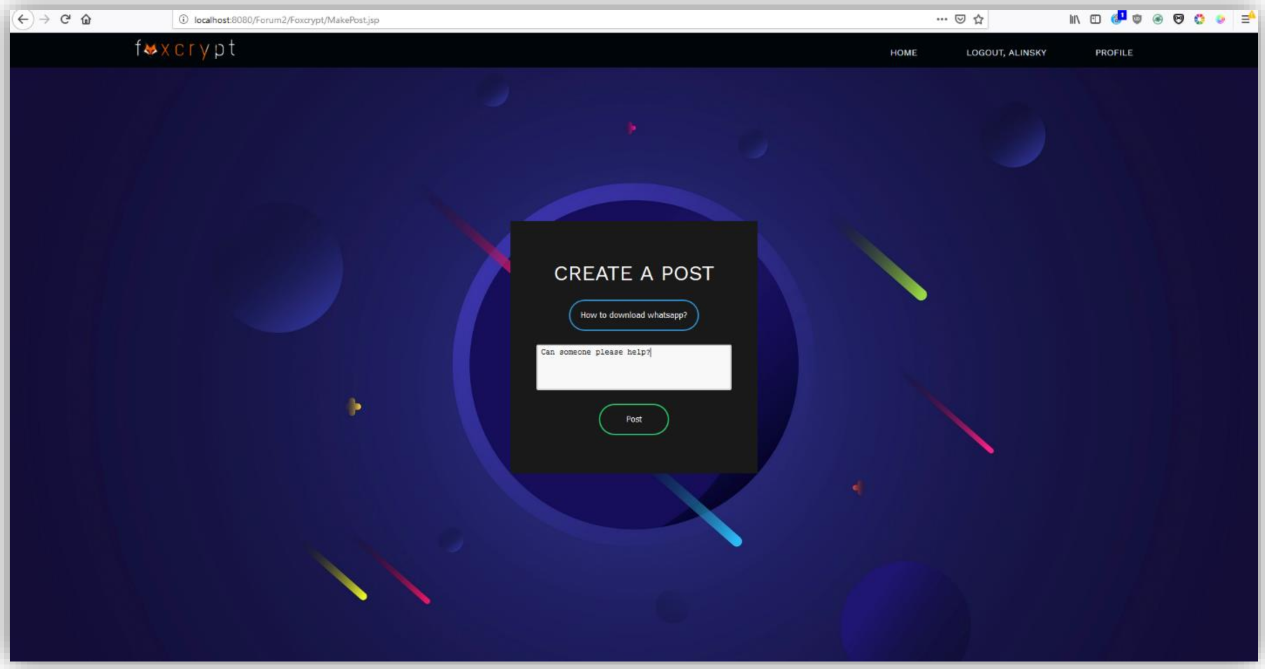
The Login page is where the user logins in with their username and password in order to Post, Comment and Reply and even to make changes to their Profile.



The Join page allows users to Register. And after registering redirects the user to the Login page. This page also performs validation on the input fields.



This page allows the user to create a new Post. Input fields contain error handling.

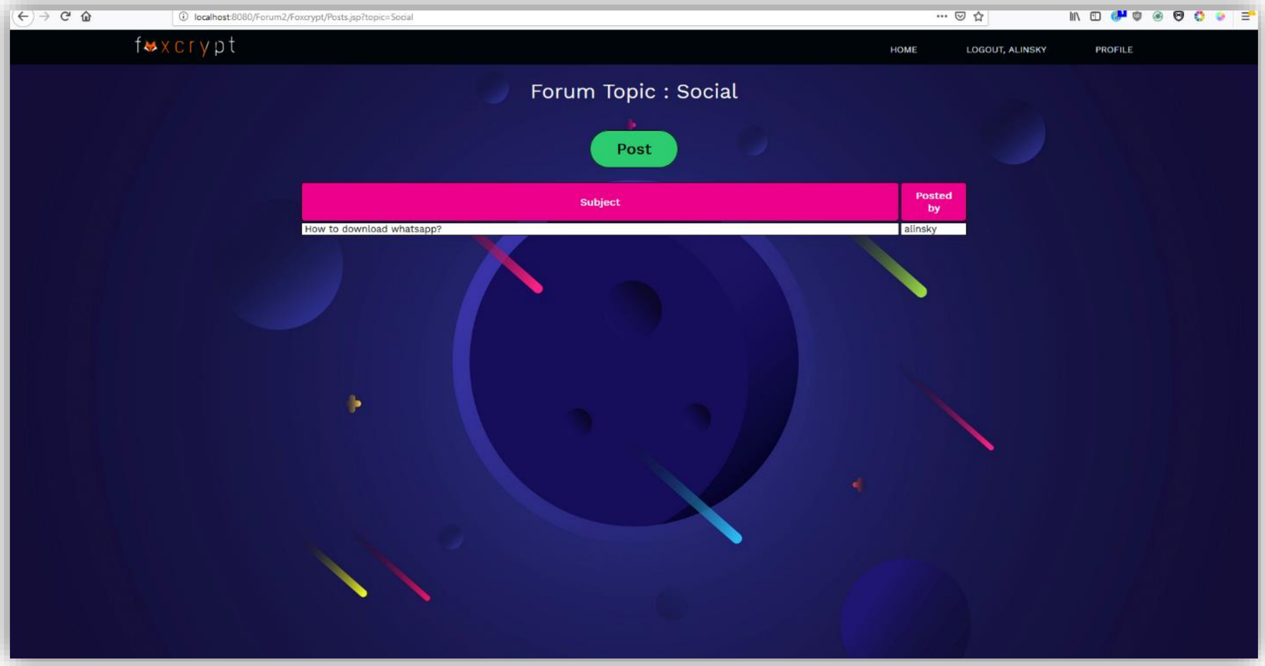


CREATE A POST

How to download whatsapp?

```
Can someone please help?
```

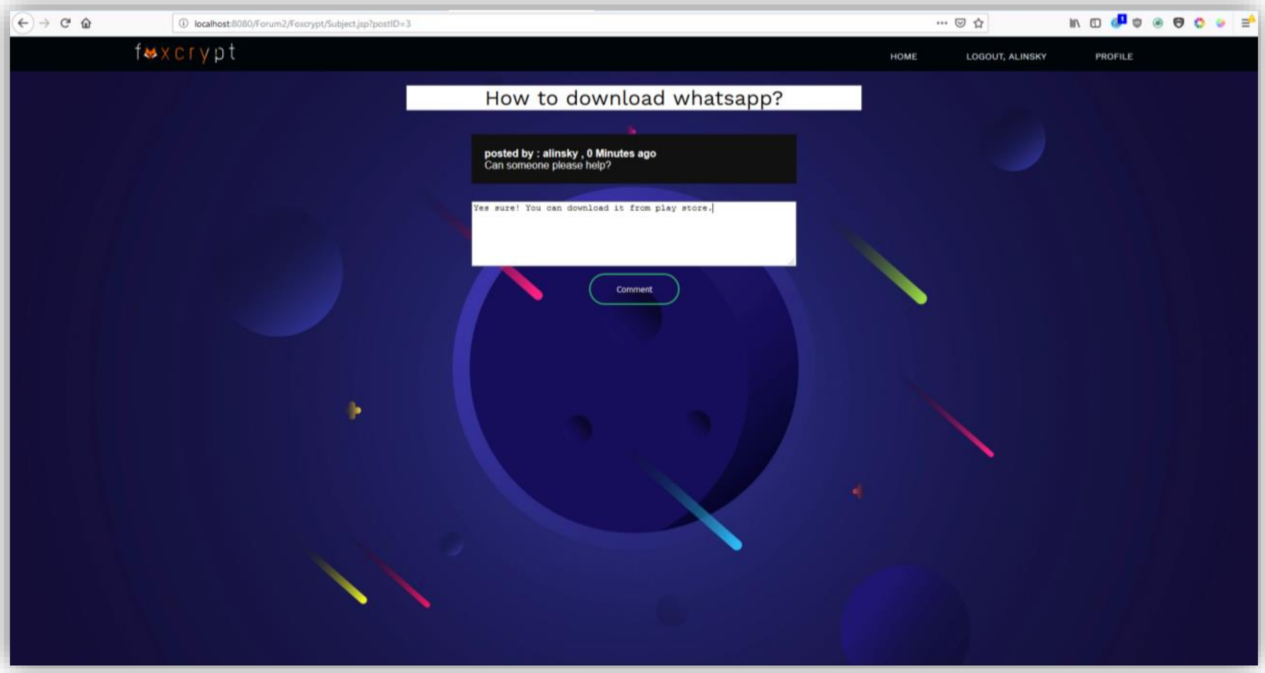
The user is able to view any post he/she clicks on.



Forum Topic : Social

Subject	Posted by
How to download whatsapp?	alinsky

This page displays information about the selected post. Information including Comments, Replies, Times & Descriptions.

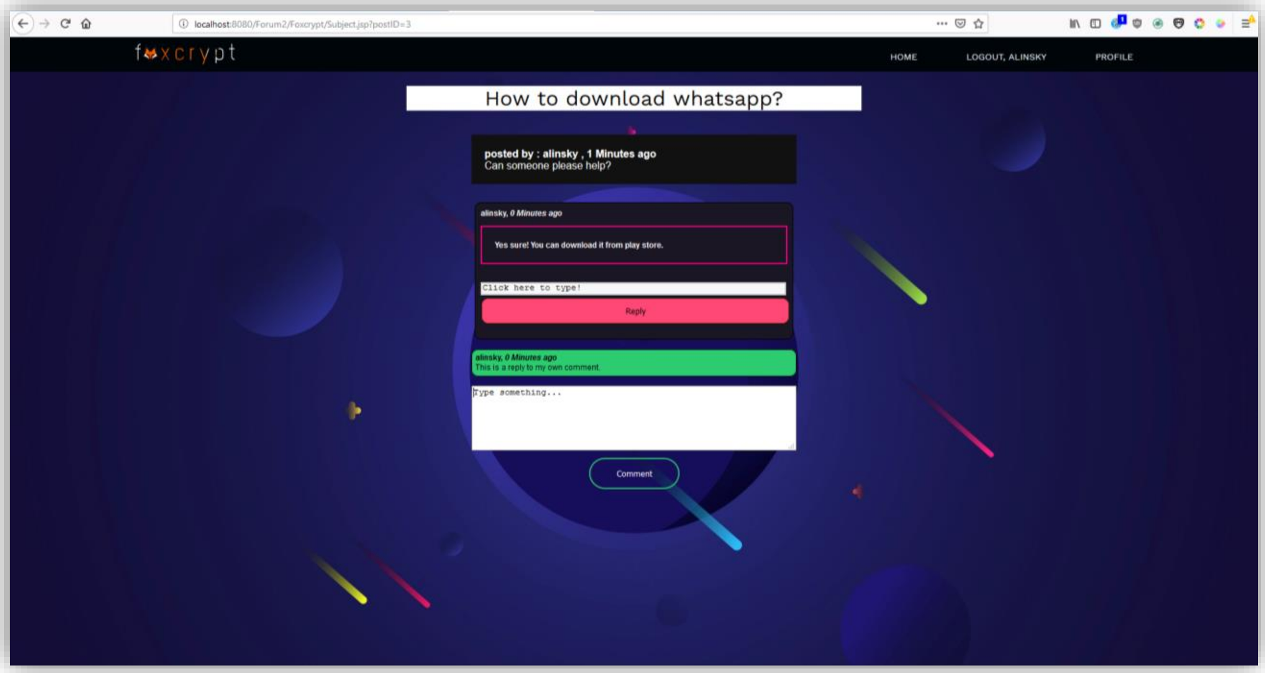


How to download whatsapp?

posted by : alinsky , 0 Minutes ago
Can someone please help?

Yes sure! You can download it from play store.

The user is able to Reply & Comment. If user is not logged in then he/she is redirected to the Login page. To first Login.



How to download whatsapp?

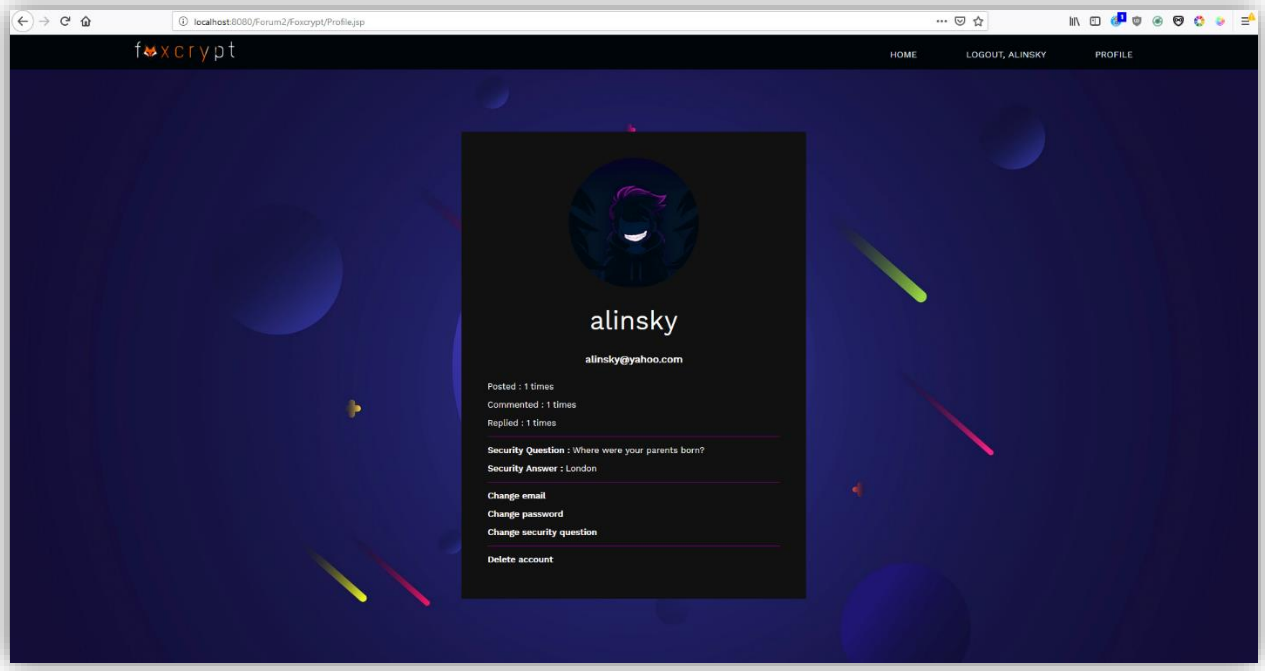
posted by : alinsky , 1 Minutes ago
Can someone please help?

alinsky, 0 Minutes ago

Yes sure! You can download it from play store.

alinsky, 0 Minutes ago
This is a reply to my own comment.

This page provides user with their information & options to change their personal information and also contains option to delete user profile.


localhost:8080/Forum2/Foxcrypt/Profile.jsp

alinsky

alinsky@yahoo.com

Posted : 1 times

Commented : 1 times

Replied : 1 times

Security Question : Where were your parents born?

Security Answer : London

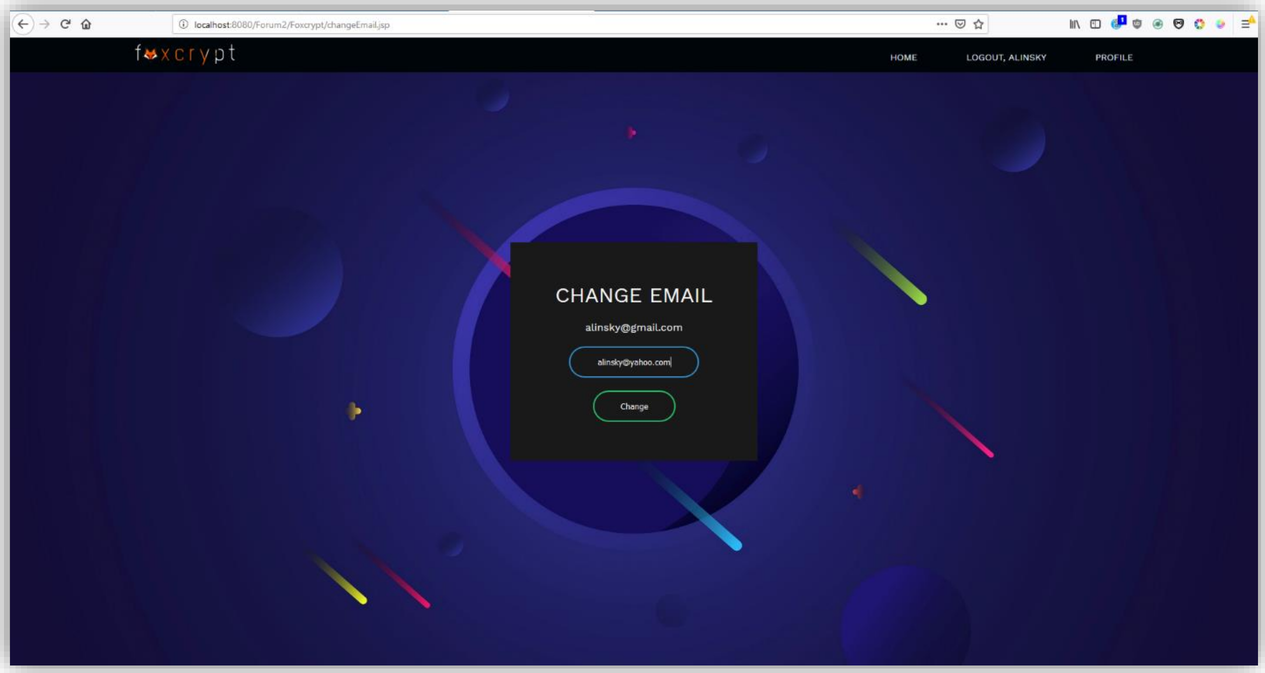
Change email

Change password

Change security question

Delete account

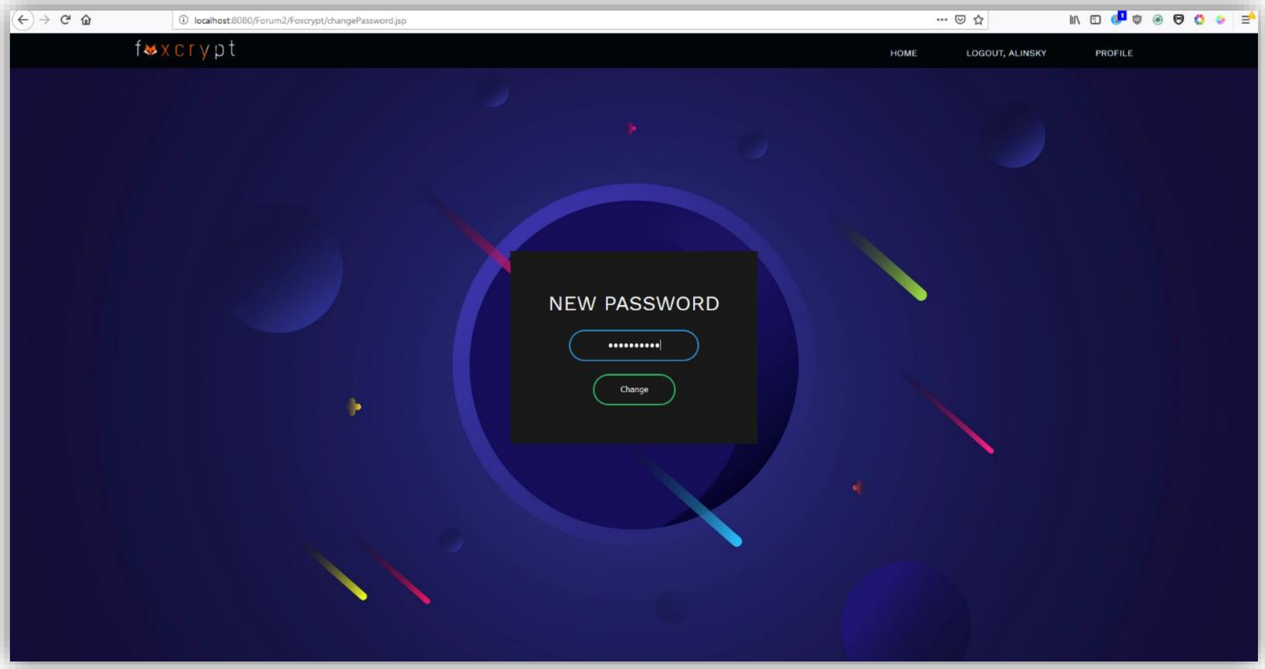
This page allows user to change their email.


localhost:8080/Forum2/Foxcrypt/changeEmail.jsp

alinsky@gmail.com

alinsky@yahoo.com

This page allows user to change their password.

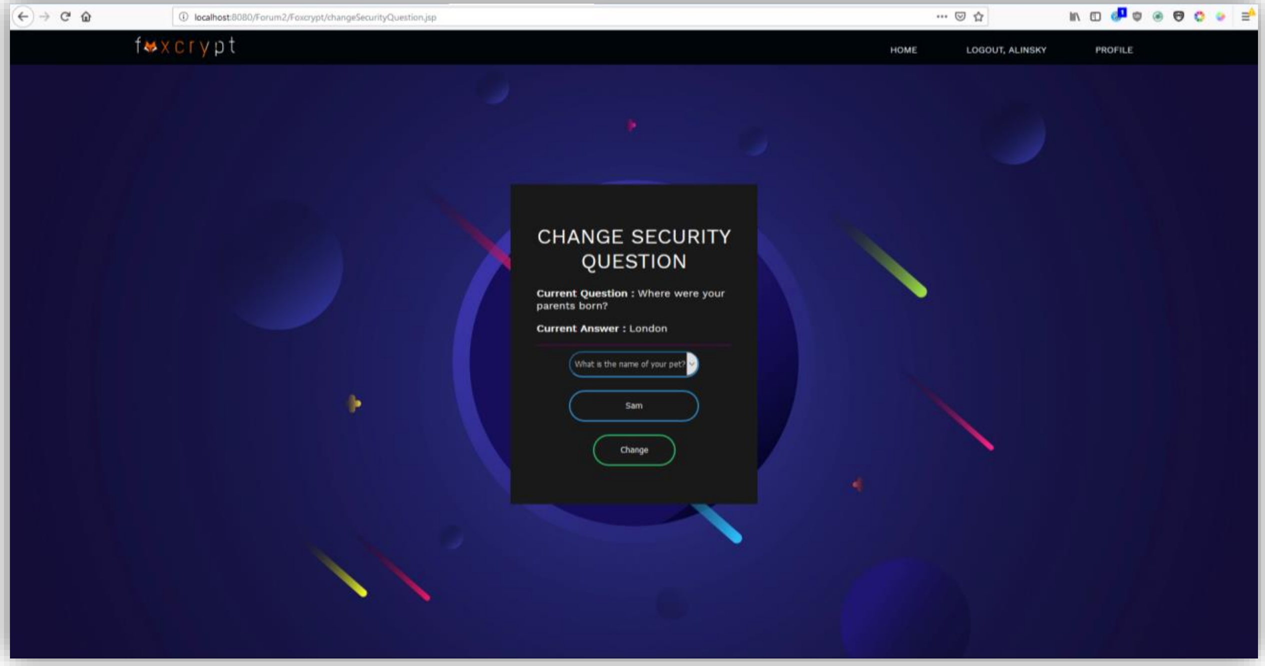


NEW PASSWORD

••••••••••

Change

This page allows users to change their security questions.



CHANGE SECURITY QUESTION

Current Question : Where were your parents born?

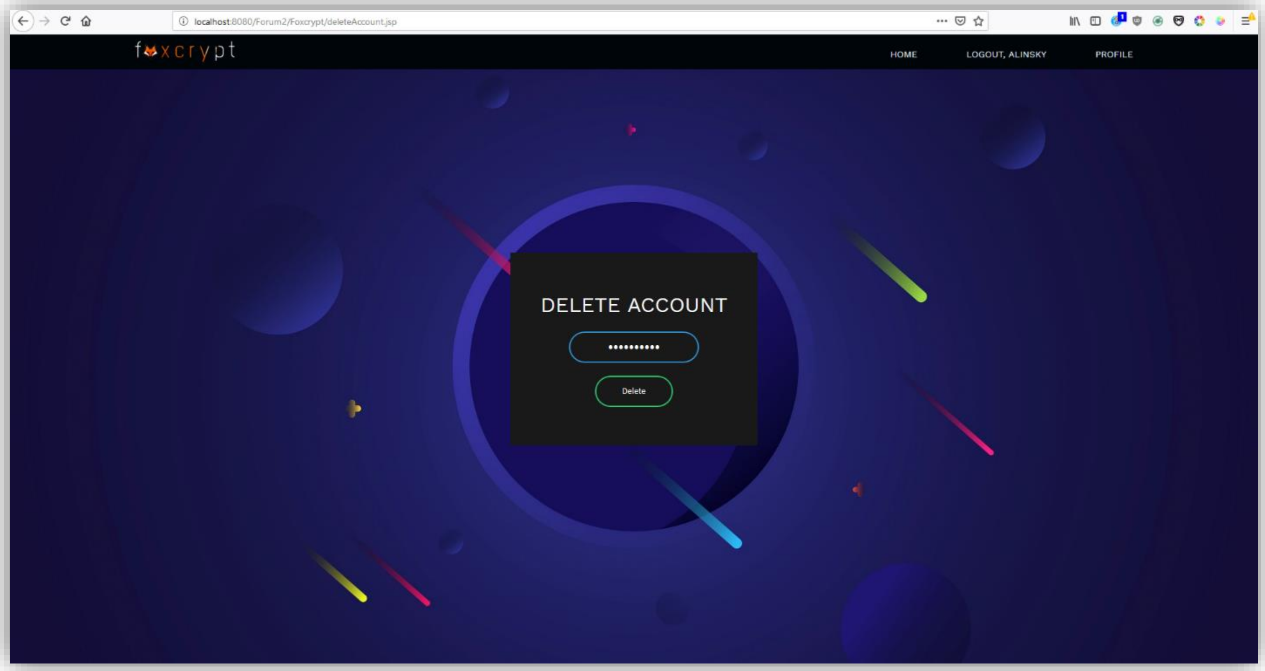
Current Answer : London

What is the name of your pet?

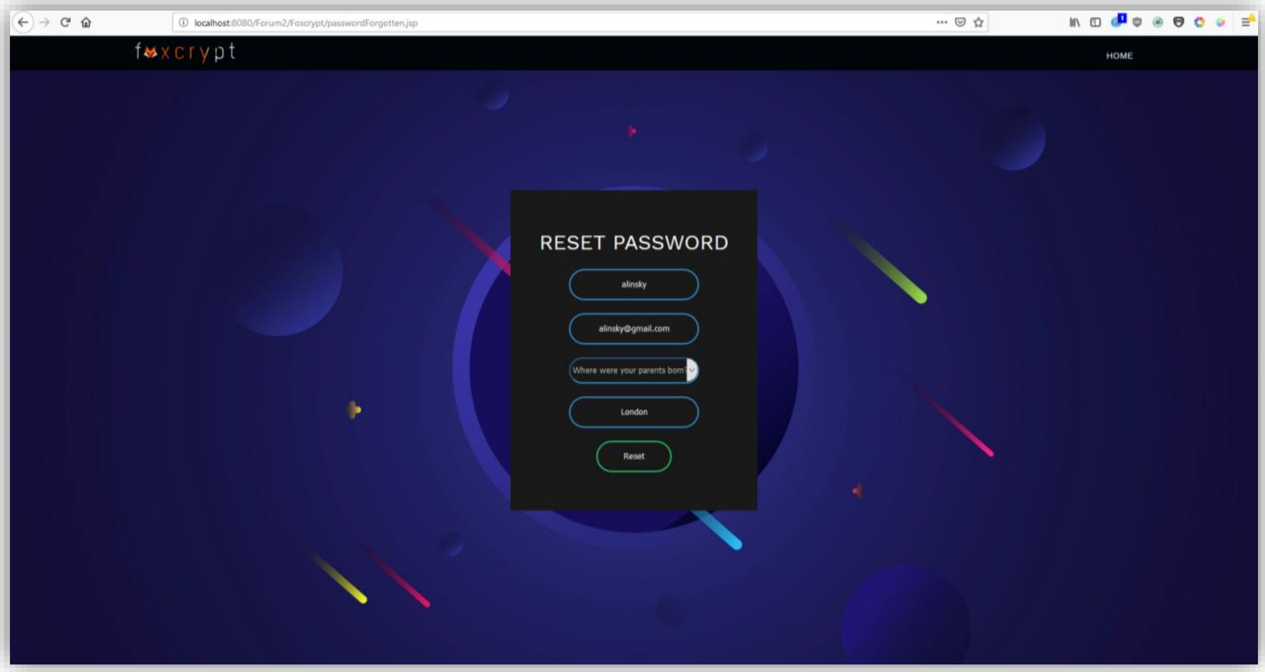
Sam

Change

The user also has the option to delete their profile. After which they are redirected to the homepage.



This page allows the user to reset password if they have forgotten. The input fields are validated first.


localhost:8080/Forum2/Foxcrypt/passwordForgotten.jsp
foxcrypt
HOME
RESET PASSWORD
alinsky
alinsky@gmail.com
Where were your parents born?
London
Reset

Bibliography & Additional Notes.

Bibliography for customizations:

- Designing login forms - https://youtu.be/HV7DtH3J2PU?list=PLnPn0BwaMLqjrwb-YIVNT0TO7D8Qxx9Yr
- Animated tiles - https://youtu.be/THIQgTqT-io?list=PLnPn0BwaMLqjrwb-YIVNT0TO7D8Qxx9Yr
- Navigation bar 1 - https://youtu.be/gQgqJG8ld7M?list=PLnPn0BwaMLqjrwb-YIVNT0TO7D8Qxx9Yr
- Navigation bar 2 - https://youtu.be/FEmysQARWFU?list=PLnPn0BwaMLqjrwb-YIVNT0TO7D8Qxx9Yr
- Tables & headers - https://youtu.be/5sbPjvbfzk8?list=PLnPn0BwaMLqjrwb-YIVNT0TO7D8Qxx9Yr

Important points:

- JDBC SQL Server driver is included in root of project folder.
- Java Docs are created for web service Processor class. Path: **Forum2/dist/javadoc/index.html.** *Select serviceServer when opening index file. serviceClient is generated by Net Beans.*
- Log file is generated and saved on desktop.

  I used GlassFish server instead of using Tomcat. The project does not really specify which server container to use. I used it because it did not run into any errors and worked flawlessly. Tomcat gave a number of problems. It does the same job as Tomcat. It also comes included with the NetBeans package.

  Video tutorial is included just in case.