# LEARNER'S ACADEMY – A BACKEND ADMIN PORTAL

This document contains the following

- Project and developer details
- Sprint planning and tasks achieved
- Core concepts used in the project
- Flowchart of the application
- Links to the GitHub repository
- Demonstration of product capabilities, appearance, and user interactions
- Unique Selling Points of the application
- Conclusion

## Project and Developer details:

### Project objective:

As a Full Stack Developer, designed and develop a backend administrative portal for the Learner's Academy. Use the GitHub repository to manage the project artifacts.

### Background of the problem statement:

Learner's Academy is a school that has an online management system. The system keeps track of its classes, subjects, students, and teachers. It has a back-office application with a single administrator login.

**The administrator can:**

- Set up a master list of all the subjects for all the classes
- Set up a master list of all the teachers
- Set up a master list of all the classes
- Assign classes for subjects from the master list
- Assign teachers to a class for a subject (A teacher can be assigned to different classes for different subjects)
- Get a master list of students (Each student must be assigned to a single class)

There will be an option to view a Class Report which will show all the information about the class, such as the list of students, subjects, and teachers

The goal of the company is to deliver a high-end quality product as early as possible.

## The flow and features of the application:

- Plan more than two sprints to complete the application
- Document the flow of the application and prepare a flow chart
- List the core concepts and algorithms being used to complete this application
- Implement the appropriate concepts such as exceptions, collections, and sorting techniques for source code optimization and increased performance

## You must use the following:

- Eclipse/IntelliJ: An IDE to code for the application
- Java: A programming language to develop web pages, databases, and others
- SQL: To create tables for admin, classes, students, and other specifics
- Git: To connect and push files from the local system to GitHub
- GitHub: To store the application code and track its versions
- Scrum: An efficient agile framework to deliver the product incrementally
- Search and Sort techniques: Data structures used for the project
- Specification document: Any open-source document or Google Docs

## The following requirements should be met:

- The source code should be pushed to your GitHub repository. You need to document the steps and write the algorithms in them.
- The submission of your GitHub repository link is mandatory. In order to track your task, you need to share the link to the repository. You can add a section to your document.
- Document the step-by-step process starting from sprint planning to the product release.
- Application should not close, exit, or throw an exception if the user specifies an invalid input.
- You need to submit the final specification document which includes:
    - Project and developer details

- Sprints planned and the tasks achieved in them
- Flowchart of the application
- Core concepts used in the project
- Links to the GitHub repository to verify the project completion

## Developer Details:

Mahendra Kumar Singh

mahendrakumarsingh9893@gmail.com

## Spring planning and Task completion:

**Sprint 1:** Analysed the application's features and prepared a flow chart and Git Repository. Implement the database schema for admin, classes, students, subjects, and teachers.

**Sprint 2:** Create basic CRUD operations for subjects, teachers, students, and classes. Implement the Class Report feature to display information about students, subjects, and teachers for a specific class.

**Sprint 3:** Tested the application numerous times to ensure a high-end quality product and pushed it to GitHub. Prepared this document highlighting the application's capabilities, appearance and user interactions

**Sprint 5:** Create database and tables. Connect the database to the project. Create models classes. Create a database utility class to retrieve data. Create login page.

**Sprint 6:** Create JSP files for all pages of the project. Create a servlet to get requests and send responses to the JSP files. Create a CSS file to format the contents. Debug and Test the project.

## Technologies and tools Used:

• Servlet: to do the business logic and works a controller for the project.

• JSP: to handle the presentation view.

• SQL: to create and manage the database.

• JDBC: to make operations on the database for the project.

• CSS: to format the contents.

• phpMyAdmin: to administrate and manage the database manually.

• Eclipse: to write and run the code.

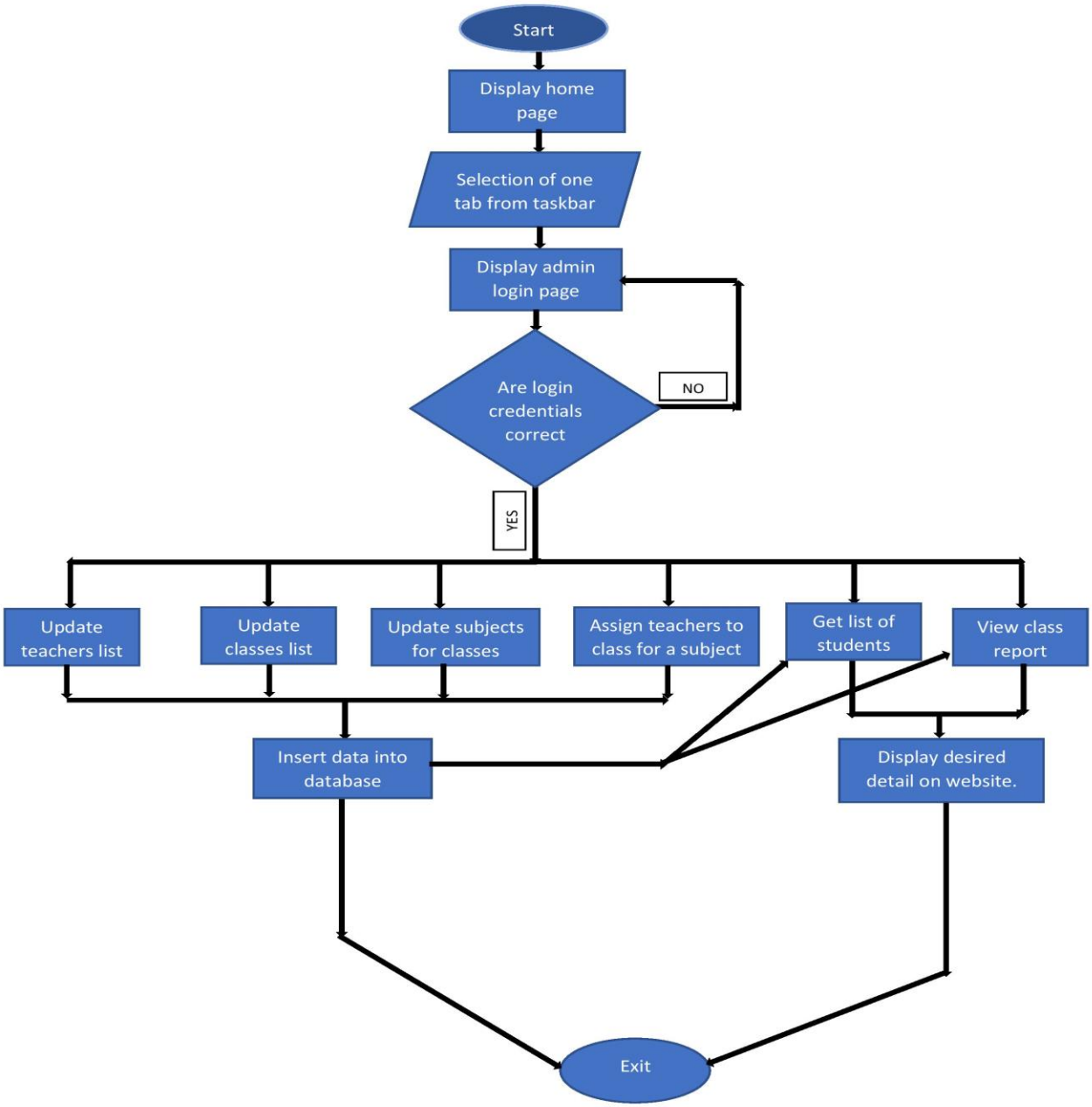
• Tomcat: to run and deploy servlet application.

## Links to the GitHub repository:

https://github.com/Mahendra1272/Learners-Academy.git

## Flowchart of the application:

## Flow Chart




## Pom.xml file:

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
	<modelVersion>4.0.0</modelVersion>
	<groupId>LearnersAcademy</groupId>
	<artifactId>LearnersAcademy</artifactId>
	<version>0.0.1-SNAPSHOT</version>
	<packaging>war</packaging>
	<dependencies>
		<dependency>
			<groupId>javax.servlet</groupId>
			<artifactId>javax.servlet-api</artifactId>
			<version>4.0.1</version>
			<scope>provided</scope>
		</dependency>
		<dependency>
			<groupId>mysql</groupId>
			<artifactId>mysql-connector-java</artifactId>
			<version>8.0.33</version>
		</dependency>
		<dependency>
			<groupId>org.hibernate</groupId>
			<artifactId>hibernate-core</artifactId>
			<version>5.6.14.Final</version>
		</dependency>
		<dependency>
			<groupId>jstl</groupId>
			<artifactId>jstl</artifactId>
			<version>1.2</version>
```

```
        </dependency>
    </dependencies>
    <build>
        <plugins>
            <plugin>
                <artifactId>maven-compiler-plugin</artifactId>
                <version>3.8.1</version>
                <configuration>
                    <source>1.8</source>
                    <target>1.8</target>
                </configuration>
            </plugin>
            <plugin>
                <artifactId>maven-war-plugin</artifactId>
                <version>3.2.3</version>
            </plugin>
        </plugins>
    </build>
</project>
```

## 1. Setting up HibernateUtil.java & hibernate.cfg.xml

- Create a package 'mypackage.util' in Java Resources.
- Create a java class HibernateUtil.java in mypackage.util.

### HibernateUtil.java

```
package mypackage.util;

import org.hibernate.SessionFactory;
import org.hibernate.cfg.Configuration;

import mypackage.entities.Admin;
import mypackage.entities.Clazz;
```

```
import mypackage.entities.Student;
import mypackage.entities.Subject;
import mypackage.entities.Teacher;

public class HibernateUtil {


	static SessionFactory sessionFactory = null;

	public static SessionFactory buildSessionFactory() {

		if(sessionFactory != null) {
			return sessionFactory;
		}

		// STEP 1: Create Configuration Object
		Configuration cfg = new
Configuration().configure("hibernate.cfg.xml").addAnnotatedClass(Student.class)

.addAnnotatedClass(Subject.class).addAnnotatedClass(Clazz.class).addAnnotatedClass(
Teacher.class).addAnnotatedClass(Admin.class);
		sessionFactory = cfg.buildSessionFactory();
		return sessionFactory;
	}
}
```

## Hibernate.cfg.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-configuration PUBLIC
  "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
  "http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">

<hibernate-configuration>
	<session-factory>
		<property
name="hibernate.connection.driver_class">com.mysql.jdbc.Driver</property>
		<property
name="hibernate.connection.url">jdbc:mysql://localhost:3306/priya</property>
		<property name="hibernate.connection.username">root</property>
		<property
name="hibernate.connection.password">Priya@0456</property>
```

```
        <property name="hibernate.show_sql">true</property>
        <property name="hibernate.format_sql">true</property>

        <property name="hbm2ddl.auto">update</property>
    </session-factory>
</hibernate-configuration>
```

## 2. Creating Entity classes

- Create a package ‘mypackage.entities’ in Java Resources.
- Create java classes Admin.java, Clazz.java, Student.java, Subject.java, Teacher.java files in mypackage.entities.
- Define data models, add ORM annotations and map relationships.

### Admin.java

```
package mypackage.entities;

import java.util.HashSet;
import java.util.Set;

import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.FetchType;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.OneToMany;
import javax.persistence.Table;

@Entity
@Table(name = "Admin")
public class Admin {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name = "admin_id")
    private int id;

    @Column(name = "admin_username")
    private String username;

    @Column(name = "admin_password")
```

```
private String password;

@OneToMany(mappedBy = "admin", fetch = FetchType.EAGER)
private Set<Subject> subjects = new HashSet<>();

@OneToMany(mappedBy = "admin", fetch = FetchType.EAGER)
private Set<Clazz> clazzes = new HashSet<>();

@OneToMany(mappedBy = "admin", fetch = FetchType.EAGER)
private Set<Teacher> teachers = new HashSet<>();

@OneToMany(mappedBy = "admin", fetch = FetchType.EAGER)
private Set<Student> students = new HashSet<>();

//Helper Methods
public void addSubject(Subject subject) {
	subjects.add(subject);
}

public void addClazz(Clazz clazz) {
	clazzes.add(clazz);
}

public void addTeacher(Teacher teacher) {
	teachers.add(teacher);
}

public void addStudent(Student student) {
	students.add(student);
}

//Getters and Setters
public int getId() {
	return id;
}

public void setId(int id) {
	this.id = id;
}
```

```
public String getUsername() {
	return username;
}

public void setUsername(String username) {
	this.username = username;
}

public String getPassword() {
	return password;
}

public void setPassword(String password) {
	this.password = password;
}

public Set<Subject> getSubjects() {
	return subjects;
}

public void setSubjects(Set<Subject> subjects) {
	this.subjects = subjects;
}

public Set<Clazz> getClazzes() {
	return clazzes;
}

public void setClazzes(Set<Clazz> clazzes) {
	this.clazzes = clazzes;
}

public Set<Teacher> getTeachers() {
	return teachers;
}

public void setTeachers(Set<Teacher> teachers) {
	this.teachers = teachers;
}
```

```
	public Set<Student> getStudents() {
		return students;
	}

	public void setStudents(Set<Student> students) {
		this.students = students;
	}
}
```

## Clazz.java

```
package mypackage.entities;

import java.util.HashSet;
import java.util.Set;

import javax.persistence.CascadeType;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.ManyToMany;
import javax.persistence.ManyToOne;
import javax.persistence.OneToMany;
import javax.persistence.Table;

@Entity
@Table(name = "Clazz")
public class Clazz {

	@Id
	@GeneratedValue(strategy = GenerationType.IDENTITY)
	@Column(name = "clazz_id")
	private int clazz_id;

	@Column(name = "clazz_name")
	private String clazz_name;

	@OneToMany(mappedBy = "clazz", cascade = CascadeType.ALL)
```

```
private Set<Subject> subjects = new HashSet<>();

@ManyToMany(mappedBy = "clazzes")
private Set<Teacher> teachers = new HashSet<>();

@ManyToOne(cascade = { CascadeType.PERSIST, CascadeType.MERGE })
@JoinColumn(name = "aid")
private Admin admin;

@OneToMany(mappedBy = "clazz", cascade = CascadeType.ALL)
private Set<Student> students = new HashSet<>();

// Helper Methods
public void addStudent(Student student) {
   this.students.add(student);
}

public void addSubject(Subject subject) {
        this.subjects.add(subject);
}

public void addTeacher(Teacher teacher) {
        this.teachers.add(teacher);
}

// Getters and Setters
public int getClazz_id() {
        return clazz_id;
}

public void setClazz_id(int clazz_id) {
        this.clazz_id = clazz_id;
}

public String getClazz_name() {
        return clazz_name;
}

public void setClazz_name(String clazz_name) {
        this.clazz_name = clazz_name;
```

```
	}

	public Set<Subject> getSubjects() {
		return subjects;
	}

	public void setSubjects(Set<Subject> subjects) {
		this.subjects = subjects;
	}

	public Set<Teacher> getTeachers() {
		return teachers;
	}

	public void setTeachers(Set<Teacher> teachers) {
		this.teachers = teachers;
	}

	public Admin getAdmin() {
		return admin;
	}

	public void setAdmin(Admin admin) {
		this.admin = admin;
	}

	public Set<Student> getStudents() {
		return students;
	}

	public void setStudents(Set<Student> students) {
		this.students = students;
	}


}
```

**Student.java**

```
package mypackage.entities;

import javax.persistence.CascadeType;
```

```
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.ManyToOne;
import javax.persistence.OneToOne;
import javax.persistence.Table;

@Entity
@Table(name = "Student")
public class Student {

	@Id
	@GeneratedValue(strategy = GenerationType.IDENTITY)
	@Column(name = "student_id")
	private int student_id;

	@Column(name = "student_name")
	private String student_name;

	@Column(name = "student_address")
	private String address;

	@OneToOne(cascade = CascadeType.ALL)
	@JoinColumn(name = "cid")
	private Clazz clazz;

	@ManyToOne(cascade = { CascadeType.PERSIST, CascadeType.MERGE })
	@JoinColumn(name = "aid")
	private Admin admin;

	// Helper Method
	public String getClazzName() {
	   if (clazz != null) {
	      return clazz.getClazz_name();
	   }
	   return "";
	}
```

```
// Getters and Setters
public int getStudent_id() {
	return student_id;
}

public void setStudent_id(int student_id) {
	this.student_id = student_id;
}

public String getStudent_name() {
	return student_name;
}

public void setStudent_name(String student_name) {
	this.student_name = student_name;
}

public String getAddress() {
	return address;
}

public void setAddress(String address) {
	this.address = address;
}

public Clazz getClazz() {
   return clazz;
}

public void setClazz(Clazz clazz) {
	this.clazz = clazz;
}

public Admin getAdmin() {
	return admin;
}

public void setAdmin(Admin admin) {
	this.admin = admin;
```

```
        }
}
```

## Subject.java

```
package mypackage.entities;

import javax.persistence.CascadeType;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.ManyToOne;
import javax.persistence.Table;

@Entity
@Table(name = "Subject")
public class Subject {

        @Id
        @GeneratedValue(strategy = GenerationType.IDENTITY)
        @Column(name = "subject_id")
        private int subject_id;

        @Column(name = "subject_name")
        private String subject_name;

        @Column(name = "subject_level")
        private String subject_level;

        @ManyToOne(cascade = { CascadeType.MERGE, CascadeType.PERSIST,
CascadeType.REFRESH })
        @JoinColumn(name = "cid")
        private Clazz clazz;

        @ManyToOne(cascade = CascadeType.MERGE)
        @JoinColumn(name = "aid")
        private Admin admin;

        // Helper Method
```

```
public String getClazzName() {
    if (clazz != null) {
        return clazz.getClazz_name();
    }
    return "";
}

// Getters and Setters
public int getSubject_id() {
        return subject_id;
}

public void setSubject_id(int subject_id) {
        this.subject_id = subject_id;
}

public String getSubject_name() {
        return subject_name;
}

public void setSubject_name(String subject_name) {
        this.subject_name = subject_name;
}

public String getSubject_level() {
        return subject_level;
}

public void setSubject_level(String subject_level) {
        this.subject_level = subject_level;
}

public Clazz getClazz() {
        return clazz;
}

public void setClazz(Clazz clazz) {
    this.clazz = clazz;
}
```

```
	public Admin getAdmin() {
		return admin;
	}

	public void setAdmin(Admin admin) {
		this.admin = admin;
	}
}
```

## Teacher.java

```
package mypackage.entities;

import java.util.HashSet;
import java.util.Set;

import javax.persistence.CascadeType;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.JoinTable;
import javax.persistence.ManyToMany;
import javax.persistence.ManyToOne;
import javax.persistence.Table;

@Entity
@Table(name = "Teacher")
public class Teacher {

	@Id
	@GeneratedValue(strategy = GenerationType.IDENTITY)
	@Column(name = "teacher_id")
	private int teacher_id;

	@Column(name = "teacher_name")
	private String teacher_name;

	@Column(name = "teacher_qual")
```

```
private String teacher_qual;

@ManyToMany(cascade = { CascadeType.PERSIST, CascadeType.MERGE })
@JoinTable(
      name = "teacher_class",
      joinColumns = { @JoinColumn(name = "teacher_id") },
      inverseJoinColumns = { @JoinColumn(name = "class_id") }
   )
private Set<Clazz> clazzes = new HashSet<>();

@ManyToOne(cascade = { CascadeType.PERSIST, CascadeType.MERGE })
@JoinColumn(name = "aid", referencedColumnName = "admin_id")
private Admin admin;

// Helper Method
public void addClass(Clazz clazz) {
        clazzes.add(clazz);
}

// Getters and Setters
public int getTeacher_id() {
        return teacher_id;
}

public void setTeacher_id(int teacher_id) {
        this.teacher_id = teacher_id;
}

public String getTeacher_name() {
        return teacher_name;
}

public void setTeacher_name(String teacher_name) {
        this.teacher_name = teacher_name;
}

public String getTeacher_qual() {
        return teacher_qual;
}
```

```
	public void setTeacher_qual(String teacher_qual) {
		this.teacher_qual = teacher_qual;
	}

	public String getClazzes() {
		StringBuffer sb = new StringBuffer();
		if(clazzes != null && clazzes.size()>0)
		{
			for(Clazz c : clazzes)
			{
				sb.append(c.getClazz_name() + ",");
			}
		}
		return sb.toString();

	}

	public void setClazzes(Set<Clazz> clazzes) {
		this.clazzes = clazzes;
	}

	public Admin getAdmin() {
		return admin;
	}

	public void setAdmin(Admin admin) {
		this.admin = admin;
	}
}
```

## 3. Creating DAO interfaces

- Create a package ‘mypackage.dao’ in Java Resources.
- Create interfaces AdminDao.java, ClazzDao.java, StudentDao.java, SubjectDao .java, TeacherDao.java.
- Define CRUD operations required for each entity.

### AdminDao.java

```
package mypackage.dao;

import mypackage.entities.Admin;

public interface AdminDao {
```

```
        void insert(Admin admin);
        Admin getAdmin(String username, String password);
}
```

**ClazzDao.java**

```
package mypackage.dao;

import java.util.List;

import mypackage.entities.Clazz;

public interface ClazzDao {

        void add(Clazz clazz) throws Exception;
        void delete(int id) throws Exception;
        Clazz getClazzById(int cid);
        List<Clazz> getAll();
   Clazz getByName(String className);
}
```

**StudentDao.java**

```
package mypackage.dao;

import java.util.List;

import mypackage.entities.Clazz;
import mypackage.entities.Student;

public interface StudentDao {

        void add(Student student) throws Exception;
        void delete(int sid) throws Exception;
        List<Student> getAll();
   List<Student> getByClass(Clazz clazz);
}
```

**SubjectDao .java**

```
package mypackage.dao;

import java.util.List;

import mypackage.entities.Clazz;
```

```
import mypackage.entities.Subject;

public interface SubjectDao {

        void add(Subject subject) throws Exception;
        void delete(int sid) throws Exception;
        void update(Subject subject) throws Exception;
        List<Subject> getAll();
        Subject getSubjectById(int subjectId);
   List<Subject> getByClass(Clazz clazz);
}
```

**TeacherDao.java**

```
package mypackage.dao;

import java.util.List;

import mypackage.entities.Teacher;

public interface TeacherDao {

        void add(Teacher teacher) throws Exception;
        void delete(int tid) throws Exception;
        List<Teacher> getAll();
}
```

## 4. Creating DAOImpl classes

- Create a package 'mypackage.daoimpl' in Java Resources.
- Create java classes AdminDaoImpl.java, ClazzDaoImpl.java, StudentDaoImpl .java, SubjectDaoImpl.java and TeacherDaoImpl.java.
- Provide concrete implementations for the CRUD operations and other data access methods defined in the DAO interface.

**AdminDaoImpl.java**

```
package mypackage.daoimpl;

import org.hibernate.Query;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;

import mypackage.dao.AdminDao;
import mypackage.entities.Admin;
```

```
import mypackage.util.HibernateUtil;

public class AdminDaoImpl implements AdminDao {

	@Override
	public void insert(Admin admin) {

		SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
		Session session = sessionFactory.openSession();
		Transaction tx = null;
		try {
			tx = session.beginTransaction();
			session.save(admin);
			tx.commit();
		} catch (Exception e) {
			e.printStackTrace();
		} finally {
			session.close();
		}
	}

	@Override
	public Admin getAdmin(String username, String password) {

		SessionFactory sessionFactory = null;
		Session session = null;
		Query<Admin> query = null;
		Admin admin = null;
		try {
			sessionFactory = HibernateUtil.buildSessionFactory();
			session = sessionFactory.openSession();
			query = session.createQuery("select a from
mypackage.entities.Admin a where a.username = ?1 and a.password = ?2");
			query.setParameter(1, username); // Set the value for the first
ordinal parameter
			query.setParameter(2, password);
			admin = query.uniqueResult();
		} catch (Exception e) {
			e.printStackTrace();
		}
```

```
                finally {
                        session.close();
                }
                return admin;
        }

}
```

**ClazzDaoImpl.java**

```
package mypackage.daoimpl;

import java.util.List;

import org.hibernate.Query;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;

import mypackage.dao.ClazzDao;
import mypackage.entities.Clazz;
import mypackage.util.HibernateUtil;

public class ClazzDaoImpl implements ClazzDao {

        @Override
        public void add(Clazz clazz) throws Exception {

                SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
                Session session = sessionFactory.openSession();
                Transaction tx = null;
                try {
                        tx = session.beginTransaction();
                        session.save(clazz);
                        tx.commit();
                } catch (Exception e) {
                        if (tx != null) {
           tx.rollback();
        }
        throw new Exception("Failed to add teacher.", e);
                } finally {
        session.close();
```

```
    }
        }

        @Override
        public void delete(int id) throws Exception {

                SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
                Session session = sessionFactory.openSession();
                Transaction tx = null;
                try {
                        tx = session.beginTransaction();
                        Clazz clazz = session.get(Clazz.class, id);
                        session.delete(clazz);
                        tx.commit();
                } catch (Exception e) {
                        if (tx != null) {
           tx.rollback();
        }
        throw new Exception("Failed to delete teacher.", e);
                } finally {
        session.close();
    }
        }

        @Override
        public List<Clazz> getAll() {

                SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
                Session session = sessionFactory.openSession();
                Query<Clazz> query = session.createQuery("select c from
mypackage.entities.Clazz c");
                return query.list();
        }

        @Override
        public Clazz getClazzById(int cid) {

                SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
                Session session = sessionFactory.openSession();
```

```
                Query<Clazz> query = session.createQuery("select c from
mypackage.entities.Clazz c where c.clazz_id = ?1");
                query.setParameter(1, cid);
                Clazz result = (Clazz) query.uniqueResult();
                session.close();
                return result;
        }

        @Override
        public Clazz getByName(String className) {

                SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
      Session session = sessionFactory.openSession();
      Query<Clazz> query = session.createQuery("SELECT c FROM Clazz c WHERE
c.className = :className");
      query.setParameter("className", className);
      Clazz result = query.uniqueResult();
      session.close();
      return result;
        }
}
```

**StudentDaoImpl .java**

```
package mypackage.daoimpl;

import java.util.List;

import org.hibernate.Query;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;

import mypackage.dao.StudentDao;
import mypackage.entities.Clazz;
import mypackage.entities.Student;
import mypackage.util.HibernateUtil;

public class StudentDaoImpl implements StudentDao {

        @Override
        public void add(Student student) throws Exception {
```

```
            SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
            Session session = sessionFactory.openSession();
            Transaction tx = null;
            try {
                  tx = session.beginTransaction();
                  session.save(student);
                  tx.commit();
            } catch (Exception e) {
                  if (tx != null) {
      tx.rollback();
   }
   throw new Exception("Failed to add teacher.", e);
            } finally {
   session.close();
}
   }

   @Override
   public void delete(int sid) throws Exception {

            SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
            Session session = sessionFactory.openSession();
            Transaction tx = null;
            try {
                  tx = session.beginTransaction();
                  Student student = session.get(Student.class, sid);
                  session.delete(student);
                  tx.commit();
            } catch (Exception e) {
                  if (tx != null) {
      tx.rollback();
   }
   throw new Exception("Failed to delete teacher.", e);
            } finally {
   session.close();
}
   }

   @Override
```

```
	public List<Student> getAll() {

		SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
		Session session = sessionFactory.openSession();
		Query<Student> query = session.createQuery("select s from
mypackage.entities.Student s");
		return query.list();
	}

	@Override
	public List<Student> getByClass(Clazz clazz) {

		SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
      Session session = sessionFactory.openSession();
      Query<Student> query = session.createQuery("SELECT s FROM Student s
WHERE s.clazz = :clazz");
      query.setParameter("clazz", clazz);
      List<Student> students = query.list();
      session.close();
      return students;
	}
}
```

**SubjectDaoImpl.java**

```
package mypackage.daoimpl;

import java.util.List;

import org.hibernate.Query;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;

import mypackage.dao.SubjectDao;
import mypackage.entities.Clazz;
import mypackage.entities.Subject;
import mypackage.util.HibernateUtil;

public class SubjectDaoImpl implements SubjectDao {

	@Override
```

```
    public void add(Subject subject) throws Exception {

            SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
            Session session = sessionFactory.openSession();
            Transaction tx = null;
            try {
                    tx = session.beginTransaction();
                    session.save(subject);
                    tx.commit();
            } catch (Exception e) {
                    if (tx != null) {
       tx.rollback();
    }
    throw new Exception("Failed to add subject.", e);
} finally {
    session.close();
}
    }

    @Override
    public void delete(int sid) throws Exception {

            SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
            Session session = sessionFactory.openSession();
            Transaction tx = null;
            try {
                    tx = session.beginTransaction();
                    Subject subject = session.get(Subject.class, sid);
                    session.delete(subject);
                    tx.commit();
            } catch (Exception e) {
                    if (tx != null) {
       tx.rollback();
    }
    throw new Exception("Failed to delete subject.", e);
            } finally {
    session.close();
}
    }
```

```
        @Override
        public void update(Subject subject) throws Exception {

                SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
                Session session = sessionFactory.openSession();
                Transaction tx = null;
                try {
                        tx = session.beginTransaction();
                        Subject sub = session.get(Subject.class, subject.getSubject_id());
                        session.update(sub);
                        tx.commit();
                } catch (Exception e) {
                        if (tx != null) {
           tx.rollback();
        }
        throw new Exception("Failed to update subject.", e);
                } finally {
        session.close();
      }
        }

        @Override
        public List<Subject> getAll() {

                SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
                Session session = sessionFactory.openSession();
                Query<Subject> query = session.createQuery("select s from
mypackage.entities.Subject s");
                return query.list();
        }

        @Override
        public Subject getSubjectById(int subjectId) {

                SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
                Session session = sessionFactory.openSession();
                Query<Subject> query = session.createQuery("select s from
mypackage.entities.Subject s where s.subject_id = ?1");
                query.setParameter(1, subjectId);
                Subject result = (Subject) query.uniqueResult();
```

```
                session.close();
                return result;
        }

        @Override
        public List<Subject> getByClass(Clazz clazz) {

                SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
      Session session = sessionFactory.openSession();
      Query<Subject> query = session.createQuery("SELECT s FROM Subject s
WHERE s.clazz = :clazz");
      query.setParameter("clazz", clazz);
      List<Subject> subjects = query.list();
      session.close();
      return subjects;
        }
}
```

**TeacherDaoImpl.java**

```
package mypackage.daoimpl;

import java.util.List;

import org.hibernate.Query;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;

import mypackage.dao.TeacherDao;
import mypackage.entities.Teacher;
import mypackage.util.HibernateUtil;

public class TeacherDaoImpl implements TeacherDao{

        @Override
        public void add(Teacher teacher) throws Exception {

                SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
                Session session = sessionFactory.openSession();
                Transaction tx = null;
                try {
```

```
                tx = session.beginTransaction();
                session.save(teacher);
                tx.commit();
            } catch (Exception e) {
                if (tx != null) {
    tx.rollback();
    }
    throw new Exception("Failed to add teacher.", e);
            } finally {
    session.close();
}

    }

    @Override
    public void delete(int tid) throws Exception {

            SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
            Session session = sessionFactory.openSession();
            Transaction tx = null;
            try {
                tx = session.beginTransaction();
                Teacher teacher = session.get(Teacher.class, tid);
                session.delete(teacher);
                tx.commit();
            } catch (Exception e) {
                if (tx != null) {
    tx.rollback();
    }
    throw new Exception("Failed to delete teacher.", e);
            } finally {
    session.close();
}
    }

    @Override
    public List<Teacher> getAll() {

            SessionFactory sessionFactory = HibernateUtil.buildSessionFactory();
            Session session = sessionFactory.openSession();
```

```
		Query<Teacher> query = session.createQuery("select t from
mypackage.entities.Teacher t");
		return query.list();
	}
}
```

## 5. Creating .jsp files

- Create .jsp files in webapp folder in src.
- Create index.jsp file which serves as the entry point of the application and other .jsp files for creating views in the web application.
- Create .css files or embed the CSS styles in .jsp file itself.

### index.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
	pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<style>
.welcome {
	text-align: center;
	padding-bottom: 20px;
}
fieldset {
	text-align: center;
	border: 2px #01257D solid;
}
legend {
	text-align: center;
	color: #01257D;
	font-size: 20px;
	font-weight: 700;
}
.option {
	font-size: 12px;
	padding-top: 8px;
}
</style>

<meta charset="UTF-8">
<title>Login Page</title>
```

```
</head>
<body>

	<%@ include file="header.jsp"%>

	<div class="welcome">
		<h1>Learner's Academy</h1>
	</div>
	<form action="logincontroller" method="post" style = "width: 400px; margin: 0px auto;">
		<fieldset>
			<legend>Admin Login</legend><br>
			Username : <input type="text" id="username" name="username" required><br><br>
			Password : <input type="password" id="password" name="password" required><br><br>
			<input type="submit" value="Submit">

			<div class="option">
				Not a registered user?<a href="register.jsp">Click here</a>
			</div>
		</fieldset>
	</form>

</body>
</html>
```

## about.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
  <style>
    h1 {
      color: #01257D;
      text-align: center;
    }
```

```
        .details-container {
            width: 600px;
            color: #01257D;
            text-align: justify;
            padding-left: 618px;
        }

        .details {
            width: 300px;
            word-wrap: break-word;
            text-align: center;
        }
    </style>
    <meta charset="UTF-8">
    <title>About Developer</title>
</head>
<body>
    <%@ include file="header.jsp"%>

    <h1> About Developer </h1>

    <div class="details-container">
        <div class="details">
            <p>
                Name : Tejeswi Devi Priya<br><br>
                Email ID : priyapillarisetty19@gmail.com
            </p>
            <p>
                Thank you for exploring the Learner's Academy Backend Admin Portal.
            </p>
            <p>
                Should you have any queries or feedback regarding the portal, please feel free to reach out to me at the provided email address.
            </p>
        </div>
    </div>
</body>
</html>
```

## addclazz.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
	pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<style>
form {
	padding-top: 50px;
}

fieldset {
	text-align: center;
	border: 2px #01257D solid;
	overflow: auto;
}

legend {
	text-align: center;
	color: #01257D;
	font-size: 20px;
	font-weight: 700;
}

label {
	display: inline-block;
	width: 80px;
	text-align: right;
	font-weight: 600;
	color: #01257D;
	margin-right: 5px;
}
input[type="text"] {
	width: 167px;
}
</style>
<meta charset="UTF-8">
<title>Add Clazz</title>
</head>
<body>
	<%@ include file="header.jsp"%>
```

```
	<form action="clazzcontroller" method="post" style="width: 400px; margin: 0px auto;">
		<fieldset>
			<legend>Add Clazz form</legend><br>
			<label>Name : </label><input type="text" name="clazzname" required><br><br>

			<input type="submit" value="Submit">
		</fieldset>
	</form>
</body>
</html>
```

## addstudent.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
	pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<%@ page import="java.util.List"%>
<%@ page import="mypackage.entities.Clazz"%>
<%@ page import="mypackage.dao.ClazzDao"%>
<%@ page import="mypackage.daoimpl.ClazzDaoImpl"%>
<head>
<style>
form {
	padding-top: 50px;
}

fieldset {
	text-align: center;
	border: 2px #01257D solid;
	overflow: auto;
}

legend {
	text-align: center;
	color: #01257D;
	font-size: 20px;
	font-weight: 700;
```

```
}

label {
	display: inline-block;
	width: 120px;
	text-align: right;
	font-weight: 600;
	color: #01257D;
	margin-right: 5px;
}
input[type="text"], select {
	width: 167px;
}
select option[disabled] {
	text-align: center;
	padding: 0;
}
</style>
<meta charset="UTF-8">
<title>Add Student</title>
</head>
<body>
	<%@ include file="header.jsp"%>

	<%
		ClazzDao clazzDao = new ClazzDaoImpl();
		List<Clazz> clazzes = clazzDao.getAll();
		request.setAttribute("clazzes", clazzes);
	%>
	<form action="studentcontroller" method="post" style="width: 400px; margin:
0px auto;">
		<fieldset>
			<legend>Add Student form</legend><br>
			<label>Name : </label><input type="text" name="studentname"
required><br><br>
			<label>Address : </label><input type="text"
name="studentaddress" required>
			<p>
			<label> Assign Class : </label> <select name="clazzname"
id="clazzname" required>
```

```
                                <option disabled selected value>-- select class --</option>
                                <% for (Clazz clazz : clazzes) { %>
                                <option value="<%= clazz.getClazz_id() %>"><%=
clazz.getClazz_name() %></option>
                                <% } %>
                        </select>
                        </p>
                        <input type="submit" value="Submit">
                </fieldset>
        </form>
</body>
</html>
```

## addsubject.jsp

```
<%@page import="mypackage.serviceimpl.ClazzServiceImpl"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
        pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<%@ page import="java.util.List"%>
<%@ page import="mypackage.entities.Clazz"%>
<%@ page import="mypackage.dao.ClazzDao"%>
<%@ page import="mypackage.daoimpl.ClazzDaoImpl"%>
<head>
<style>
form {
        padding-top: 50px;
}

fieldset {
        text-align: center;
        border: 2px #01257D solid;
        overflow: auto;
}

legend {
        text-align: center;
        color: #01257D;
        font-size: 20px;
        font-weight: 700;
```

```
}

label {
	display: inline-block;
	width: 120px;
	text-align: right;
	font-weight: 600;
	color: #01257D;
	margin-right: 5px;
}

input[type="text"], select {
	width: 167px;
}
select option[disabled] {
	text-align: center;
	padding: 0;
}
</style>
<meta charset="UTF-8">
<title>Add Subject</title>
</head>
<body>
	<%@ include file="header.jsp"%>

	<%
		ClazzDao clazzDao = new ClazzDaoImpl();
		List<Clazz> clazzes = clazzDao.getAll();
		request.setAttribute("clazzes", clazzes);
	%>
	<form action="subjectcontroller" method="post"
		style="width: 400px; margin: 0px auto;">
		<fieldset>
			<legend>Add Subject form</legend>
			<br> <label>Name: </label><input type="text"
name="subjectname"
				required>
			<p>
				<label>Level: </label> <select name="subjectlevel"
id="subjectlevel"
```

```
                                        required>
                                        <option disabled selected value>-- select level --
</option>
                                        <option value="Level I">Level I</option>
                                        <option value="Level II">Level II</option>
                                    </select>
                            </p>
                            <p>
                            <label> Assign Class : </label> <select name="clazzname"
id="clazzname"
                                    required>
                                    <option disabled selected value>-- select class --</option>
                                    <% for (Clazz clazz : clazzes) { %>
                                    <option value="<%= clazz.getClazz_id() %>"><%=
clazz.getClazz_name() %></option>
                                    <% } %>
                            </select>
                            </p>

                            <input type="submit" value="Submit">
                    </fieldset>
            </form>
</body>
</html>
```

## addteacher.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
        pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<%@ page import="java.util.List"%>
<%@ page import="mypackage.entities.Clazz" %>
<%@ page import="mypackage.dao.ClazzDao" %>
<%@ page import="mypackage.daoimpl.ClazzDaoImpl" %>
<head>
<style>
form {
        padding-top: 50px;
}
```

```
fieldset {
	text-align: center;
	border: 2px #01257D solid;
	overflow: auto;
}

legend {
	text-align: center;
	color: #01257D;
	font-size: 20px;
	font-weight: 700;
}

label {
	display: inline-block;
	width: 120px;
	text-align: right;
	font-weight: 600;
	color: #01257D;
	margin-right: 5px;
}
input[type="text"], select {
	width: 167px;
}
select option[disabled] {
	text-align: center;
	padding: 0;
}
</style>
<meta charset="UTF-8">
<title>Add Teacher</title>
</head>
<body>
	<%@ include file="header.jsp"%>

	<%
		ClazzDao clazzDao = new ClazzDaoImpl();
		List<Clazz> clazzes = clazzDao.getAll();
		request.setAttribute("clazzes", clazzes);
	%>
```

```
        <form action="teachercontroller" method="post" style="width: 400px; margin:
0px auto;">
                <fieldset>
                        <legend>Add Teacher form</legend><br>
                        <label>Name : </label><input type="text" name="teachername"
required>
                        <p>
                                <label>Qualification : </label> <select
name="qualification" id="qualification" required>
                                        <option disabled selected value>-- select
qualification --</option>
                                        <option value="BTech">BTech</option>
                                        <option value="MTech">MTech</option>
                                </select>
                        </p>
                        <p>
                        <label> Assign Class : </label> <select name="clazzname"
id="clazzname" required>
                                <option disabled selected value>-- select class --</option>
                                <% for (Clazz clazz : clazzes) { %>
                                <option value="<%= clazz.getClazz_id() %>"><%=
clazz.getClazz_name() %></option>
                                <% } %>
                        </select>
                        </p>

                        <input type="submit" value="Submit">
                </fieldset>
        </form>
</body>
</html>
```

## clazzlist.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
   pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<%@ page import="java.util.List" %>
<%@ page import="mypackage.entities.Clazz" %>
<%@ page import="mypackage.dao.ClazzDao" %>
```

```
<%@ page import="mypackage.daoimpl.ClazzDaoImpl" %>
<head>
<style>
 table {
  border-collapse: collapse;
  width: 50%;
  margin: auto;
 }
 th, td {
        border: 2px #01257D solid;
  text-align: center;
 }
 th {
  color: #333;
  font-weight: bold;
 }
 .head1 {
        color: #01257D;
        padding-top: 10px;
        padding-bottom: 10px;
        text-align: center;
 }
</style>

<meta charset="UTF-8">
<title>Class List</title>
</head>
<body>
        <%@ include file="header.jsp"%>
        <%
                ClazzDao clazzDao = new ClazzDaoImpl();
                List<Clazz> clazzes = clazzDao.getAll();
                request.setAttribute("clazzes", clazzes);
        %>
        <div class="head1">
                <h1> Class List </h1>
        </div>
                <table>
                        <tr>
                                <th> Class ID </th>
```

```
                        <th> Class Name </th>
                        <th> Actions </th>
                    </tr>
        <% int sequence = 1; %>
        <%
            for(Clazz clazz : clazzes) {
        %>
                    <tr>
                        <td><%= sequence %></td>
                        <td><%= clazz.getClazz_name() %></td>
                        <td>
                            <form action="clazzlist.jsp" method="POST">
                                <input type="hidden" name="cid"
value="<%= clazz.getClazz_id() %>">
                                <button type="submit" name="action"
value="delete">Delete</button>
                            </form>
                        </td>
                    </tr>
                    <% sequence++; %>
        <%
            }
        %>
                    <tr>
                        <td colspan="4">
                            <form action="addclazz.jsp" method="GET">
                                <button type="submit"> Add Class
</button>
                            </form>
                        </td>
                    </tr>
            </table>
        <%
            String action = request.getParameter("action");
            if (action != null && action.equals("delete")) {
                int cid = Integer.parseInt(request.getParameter("cid"));
                clazzDao.delete(cid);
                response.sendRedirect("clazzlist.jsp");
            }
        %>
```

```
</body>
</html>
```

## header.jsp

```
<!DOCTYPE htaml>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<style>
@import
        url("https://fonts.googleapis.com/css2?family=Open+Sans&display=swap");

body {
        font-family: "Open Sans", sans-serif;
        margin: 0;
        background: url(https://images.unsplash.com/photo-1464618663641-
bbdd760ae84a?ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8fA%3D%3D
&auto=format&fit=crop&w=1170&q=80);
   background-repeat: no-repeat;
   background-size: cover;
}
a {
        text-decoration: none;
        color: #00FFFF;
        padding-left: 15px;
}
a:hover {
        color: #fff;
}
.header {
        border-bottom: 1px solid #ccc;
        display: flex;
        background: #01257D;
        justify-content: space-between;
}
.site-name h1 {
        font-size: 28px;
```

```
		margin: 10px 10px 0px 10px;
		display: inline-block;
}
.nav-options ul, .nav-options li {
		margin: 0;
		padding-right: 20px;
}
.nav-options li {
		display: inline-block;
		margin: 23px 15px 16px 16px;
}
</style>
<title>Header</title>
</head>
<body>
		<header class="header">
			<div class="site-name">
				<h1>
					<a href="home.jsp">Learner's Academy</a>
				</h1>
			</div>
			<nav class="nav-options">
				<ul class="nav">
					<li><a href="home.jsp">Home</a></li>
					<li><a href="about.jsp">About</a></li>
					<li><a href="logout.jsp">Logout</a></li>
				</ul>
			</nav>
		</header>
</body>
</html>
```

## home.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" href="home-styles.css">
<meta charset="UTF-8">
```

```
<title>Home Page</title>
</head>
<body>
	<%@ include file="header.jsp"%>

	<div class="heading">
		<h2>Welcome to Learner's Academy</h2>
	</div>
	<div class="tiles">
		<div class="column-3 option-1">
			<h2 class="option-text"><a
href="subjectslist.jsp">Subjects<br>List</a></h2>
		</div>
		<div class="column-3 option-2">
			<h2 class="option-text"><a
href="teacherslist.jsp">Teachers<br>List</a></h2>
		</div>
		<div class="column-3 option-3">
			<h2 class="option-text"><a
href="clazzlist.jsp">Classes<br>List</a></h2>
		</div>
		<div class="column-3 option-4">
			<h2 class="option-text"><a
href="studentslist.jsp">Students<br>List</a></h2>
		</div>
		<div class="column-3 option-5">
			<h2 class="option-text">Learner's Academy</h2>
		</div>
		<div class="column-3 option-6">
			<h2 class="option-text"><a href="viewclassreport.jsp">View
Class<br>Report</a></h2>
		</div>
	</div>
</body>
</html>
```

## invalidcred.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
	pageEncoding="UTF-8"%>
<!DOCTYPE html>
```

```
<html>
<head>
<style>
h2 {
	padding-left: 10px;
	padding-top: 20px;
	text-align: center;
}
</style>
<meta charset="UTF-8">
<title>Invalid Credentials</title>
</head>
<body>

	<%@ include file="header.jsp"%>

	<h2>You have entered invalid credentials! <br><br>
	<a href="index.jsp">Click here</a> to try again.</h2>
</body>
</html>
```

## logout.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<%@ page import="javax.servlet.http.HttpSession.*" %>
<head>
<style>
h2 {
	padding-left: 10px;
	padding-top: 20px;
	text-align: center;
}
</style>
<meta charset="UTF-8">
<title>Logged out</title>
</head>
<body>
```

```
    <%@ include file="header.jsp"%>

    <h2>You have logged out successfully. <br><br>
    <a href="index.jsp">Click here</a> to login again.</h2>

    <%
        HttpSession sess = request.getSession(false);
        if (sess != null) {
            session.invalidate();
        }
    %>

</body>
</html>
```

## register.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<style>
form {
        padding-top: 50px;
}
fieldset {
        text-align: center;
        border: 2px #01257D solid;
        overflow: auto;
}
legend {
        text-align: center;
        color: #01257D;
        font-size: 20px;
        font-weight: 700;
}
</style>
<meta charset="UTF-8">
<title>Register</title>
</head>
```

```
<body>
	<%@ include file="header.jsp"%>

	<form action="registercontroller" method="post" style = "width: 400px; margin:
0px auto;">
		<fieldset>
			<legend>Registration form</legend><br>
			Username : <input type="text" name="username"
required><br><br>
			Password : <input type="password" name="password"
required><br><br>
			<input type="submit" value="Submit">
		</fieldset>
	</form>
</body>
</html>
```

## registrationsuccessful.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
   pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<style>
h2 {
	padding-left: 10px;
	padding-top: 20px;
	text-align: center;
}
</style>
<meta charset="UTF-8">
<title>Registration Successful!</title>
</head>
<body>

	<%@ include file="header.jsp"%>

	<h2>Your registration is successful. <br><br>
	<a href="index.jsp">Click here</a> to login.</h2>
```

```
</body>
</html>
```

## studentslist.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<%@ page import="java.util.List" %>
<%@ page import="mypackage.entities.Student" %>
<%@ page import="mypackage.dao.StudentDao" %>
<%@ page import="mypackage.daoimpl.StudentDaoImpl" %>
<head>
<style>
  table {
    border-collapse: collapse;
    width: 50%;
    margin: auto;
  }
  th, td {
          border: 2px #01257D solid;
    text-align: center;
  }
  th {
    color: #333;
    font-weight: bold;
  }
  .head1 {
          color: #01257D;
          padding-top: 10px;
          padding-bottom: 10px;
          text-align: center;
  }
</style>

<meta charset="UTF-8">
<title>Students List</title>
</head>
<body>
          <%@ include file="header.jsp"%>
          <%
```

```
		StudentDao studentDao = new StudentDaoImpl();
		List<Student> students = studentDao.getAll();
		request.setAttribute("students", students);
	%>
	<div class="head1">
		<h1> Students List </h1>
	</div>
		<table>
			<tr>
				<th> Student ID </th>
				<th> Student Name </th>
				<th> Student Address </th>
				<th> Assigned Class </th>
				<th> Actions </th>
			</tr>
	<% int sequence = 1; %>
	<%
		for(Student student : students) {
	%>
			<tr>
				<td><%= sequence %></td>
				<td><%= student.getStudent_name() %></td>
				<td><%= student.getAddress() %></td>
				<td><%= student.getClazzName() %></td>

				<td>
					<form action="studentslist.jsp" method="POST">
						<input type="hidden" name="sid"
value="<%= student.getStudent_id() %>">
						<button type="submit" name="action"
value="delete">Delete</button>
					</form>
				</td>
			</tr>
			<% sequence++; %>
	<%
		}
	%>
			<tr>
				<td colspan="5">
```

```
                                        <form action="addstudent.jsp" method="GET">
                                                <button type="submit">Add
Student</button>
                                        </form>
                                </td>
                        </tr>
                </table>
        <%
                String action = request.getParameter("action");
                if (action != null && action.equals("delete")) {
                        int sid = Integer.parseInt(request.getParameter("sid"));
                        studentDao.delete(sid);
                        response.sendRedirect("studentslist.jsp");
                }
        %>
</body>
</html>
```

## subjectslist.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
   pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<%@ page import="java.util.List" %>
<%@ page import="mypackage.entities.Subject" %>
<%@ page import="mypackage.dao.SubjectDao" %>
<%@ page import="mypackage.daoimpl.SubjectDaoImpl" %>
<head>
<style>
 table {
   border-collapse: collapse;
   width: 50%;
   margin: auto;
 }
 th, td {
        border: 2px #01257D solid;
   text-align: center;
 }
 th {
   color: #333;
```

```
   font-weight: bold;
  }
  .head1 {
        color: #01257D;
        padding-top: 10px;
        padding-bottom: 10px;
        text-align: center;
  }
</style>

<meta charset="UTF-8">
<title>Subjects List</title>
</head>
<body>
        <%@ include file="header.jsp"%>
        <%
                SubjectDao subjectDao = new SubjectDaoImpl();
                List<Subject> subjects = subjectDao.getAll();
                request.setAttribute("subjects", subjects);
        %>
        <div class="head1">
                <h1> Subjects List </h1>
        </div>
                <table>
                        <tr>
                                <th> Subject ID </th>
                                <th> Subject Name </th>
                                <th> Subject Qualification </th>
                                <th> Assigned Class </th>
                                <th> Actions </th>
                        </tr>
        <% int sequence = 1; %>
        <%
                for(Subject subject : subjects) {
        %>
                        <tr>
                                <td><%= sequence %></td>
                                <td><%= subject.getSubject_name() %></td>
                                <td><%= subject.getSubject_level() %></td>
                                <td><%= subject.getClazzName() %></td>
```

```
                                <td>
                                    <form action="subjectslist.jsp" method="POST">
                                        <input type="hidden" name="sid"
value="<%= subject.getSubject_id() %>">
                                        <button type="submit" name="action"
value="delete">Delete</button>
                                    </form>
                                </td>
                            </tr>
                            <% sequence++; %>
                <%
                    }
                %>
                            <tr>
                                <td colspan="5">
                                    <form action="addsubject.jsp" method="GET">
                                        <button type="submit">Add
Subject</button>
                                    </form>
                                </td>
                            </tr>
                        </table>
                <%
                    String action = request.getParameter("action");
                    if (action != null && action.equals("delete")) {
                        int sid = Integer.parseInt(request.getParameter("sid"));
                        subjectDao.delete(sid);
                        response.sendRedirect("subjectslist.jsp");
                    }
                %>
</body>
</html>
```

## teacherslist.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<%@ page import="java.util.List" %>
<%@ page import="mypackage.entities.Teacher" %>
```

```
<%@ page import="mypackage.dao.TeacherDao" %>
<%@ page import="mypackage.daoimpl.TeacherDaoImpl" %>
<head>
<style>
 table {
   border-collapse: collapse;
   width: 50%;
   margin: auto;
 }
 th, td {
        border: 2px #01257D solid;
   text-align: center;
 }
 th {
   color: #333;
   font-weight: bold;
 }
 .head1 {
        color: #01257D;
        padding-top: 10px;
        padding-bottom: 10px;
        text-align: center;
 }
</style>

<meta charset="UTF-8">
<title>Teachers List</title>
</head>
<body>
        <%@ include file="header.jsp"%>
        <%
                TeacherDao teacherDao = new TeacherDaoImpl();
                List<Teacher> teachers = teacherDao.getAll();
                request.setAttribute("teachers", teachers);
        %>
        <div class="head1">
                <h1> Teachers List </h1>
        </div>
                <table>
                        <tr>
```

```
                    <th> Teacher ID </th>
                    <th> Teacher Name </th>
                    <th> Teacher Qualification </th>
                    <th> Actions </th>
                </tr>
        <% int sequence = 1; %>
        <%
            for(Teacher teacher : teachers) {
        %>
                <tr>
                    <td><%= sequence %></td>
                    <td><%= teacher.getTeacher_name() %></td>
                    <td><%= teacher.getTeacher_qual() %></td>
                    <td>
                        <form action="teacherslist.jsp" method="POST">
                            <input type="hidden" name="tid"
value="<%= teacher.getTeacher_id() %>">
                            <button type="submit" name="action"
value="delete">Delete</button>
                        </form>
                    </td>
                </tr>
                <% sequence++; %>
        <%
            }
        %>
                <tr>
                    <td colspan="4">
                        <form action="addteacher.jsp" method="GET">
                            <button type="submit">Add
Teacher</button>
                        </form>
                    </td>
                </tr>
            </table>
        <%
            String action = request.getParameter("action");
            if (action != null && action.equals("delete")) {
                int tid = Integer.parseInt(request.getParameter("tid"));
                teacherDao.delete(tid);
```

```
                    response.sendRedirect("teacherslist.jsp");
                }
        %>
</body>
</html>
```

## viewclassreport.jsp

```
<%@page import="org.hibernate.internal.build.AllowSysOut"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
        pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<%@ page import="java.util.List"%>
<%@ page import="mypackage.entities.Student"%>
<%@ page import="mypackage.entities.Subject"%>
<%@ page import="mypackage.entities.Clazz"%>
<%@ page import="mypackage.entities.Teacher"%>
<%@ page import="mypackage.dao.ClazzDao"%>
<%@ page import="mypackage.daoimpl.ClazzDaoImpl"%>
<head>
<style>
table {
        border-collapse: collapse;
        width: 50%;
        margin: auto;
}

th, td {
        border: 2px #01257D solid;
        text-align: center;
}

th {
        color: #01257D;
        font-weight: bold;
}
td {
        color : #fff;
}
```

```
.head1 {
	color: #01257D;
	padding-top: 10px;
	padding-bottom: 10px;
	text-align: center;
}
</style>
<meta charset="UTF-8">
<title>Class Report</title>
</head>
<body>
	<%@ include file="header.jsp"%>

	<%
   ClazzDao clazzDao = new ClazzDaoImpl();
   List<Clazz> clazzes = clazzDao.getAll();
   request.setAttribute("clazzes", clazzes);
%>
	<div class="head1">
		<h1>Class Report</h1>
	</div>
	<table>
		<tr>
			<th>Student ID</th>
			<th>Student Name</th>
			<th>Assigned Subject</th>
			<th>Assigned Class</th>
			<th>Assigned Teacher</th>
		</tr>
		<%
      int sequence = 1;
      %>
		<%
      for (Clazz clazz : clazzes) {
   %>
		<tr>
			<td><%= sequence %></td>
			<td>
				<%
               for (Student student : clazz.getStudents()) {
```

```
                out.println(student.getStudent_name() + "<br>");
            }
        %>
                    </td>
                    <td>
                        <%
            for (Subject subject : clazz.getSubjects()) {
                out.println(subject.getSubject_name() + " " + subject.getSubject_level()+
"<br>");
            }
        %>
                    </td>
                    <td><%= clazz.getClazz_name() %></td>
                    <td>
                        <%
            for (Teacher teacher : clazz.getTeachers()) {
                out.println(teacher.getTeacher_name() + "<br>");
            }
        %>
                    </td>
                </tr>
                <%
      sequence++;
   %>
                <%
      }
   %>
        </table>
</body>
</html>
```

## home-styles.css

```
@charset "UTF-8";

.heading {
  text-align:center;
  color:#01257D;
  margin-top: 25px;
  margin-bottom:25px;
  font-size: 24px;
```

```
}

.column-3 {
  float: left;
  width: 25%;
  padding: 10px;
  margin:20px;
  height: 150px;
}

.tiles{
        padding-left: 150px;
}

.option-1 {
  background-color: #01257D;
}
.option-2 {
  background-color: #01257D;
}
.option-3 {
  background-color: #01257D;
}
.option-4 {
  background-color: #01257D;
}
.option-5 {
  background-image: url('laptop-open-book-assortment.jpg');
  background-size: cover;
  background-position: center;
}
.option-6 {
  background-color: #01257D;
}

.option-text {
  text-align:center;
  font-size:30px;
  color: white;
}
```

```
.option-1:hover {
  background-color:#98c1d9;
}
.option-2:hover {
   background-color: #98c1d9;
}
.option-3:hover {
  background-color:#98c1d9;
}
.option-4:hover {
   background-color: #98c1d9;
}
.option-6:hover {
   background-color: #98c1d9;
}
```

## 6. Creating Controller java servlets

- Create a package ‘mypackage.controllers’ in Java Resources.
- Create java servlets ClazzController.java, LoginController.java, RegisterContoller.java, StudentController.java, SubjectController.java and TeacherContoller.java.

### ClazzController.java

```
package mypackage.controllers;

import java.io.IOException;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import mypackage.entities.Admin;
import mypackage.models.ClazzModel;
import mypackage.serviceimpl.ClazzServiceImpl;
import mypackage.services.ClazzService;

@WebServlet("/clazzcontroller")
```

```
public class ClazzController extends HttpServlet {
	private static final long serialVersionUID = 1L;

	private ClazzService clazzService = new ClazzServiceImpl();

	protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {

		String clazzName = request.getParameter("clazzname");
		ClazzModel clazzModel = new ClazzModel(clazzName);
		HttpSession session = request.getSession();
		Admin admin = (Admin) session.getAttribute("admin");
		clazzService.addClazz(admin, clazzModel);
		RequestDispatcher rd = request.getRequestDispatcher("clazzlist.jsp");
		rd.forward(request, response);
	}

}
```

## LoginController.java

```
package mypackage.controllers;

import java.io.IOException;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import mypackage.entities.Admin;
import mypackage.models.LoginModel;
import mypackage.serviceimpl.AdminServiceImpl;
import mypackage.services.AdminService;

@WebServlet("/logincontroller")
public class LoginController extends HttpServlet {
	private static final long serialVersionUID = 1L;

	private AdminService adminService = new AdminServiceImpl();
```

```
	protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {

		String username = request.getParameter("username");
		String password = request.getParameter("password");

		LoginModel loginModel = new LoginModel();
		loginModel.setUsername(username);
		loginModel.setPassword(password);
		Admin admin = adminService.getAdmin(loginModel);
		if(admin != null) {
			HttpSession session = request.getSession();
			session.setAttribute("admin", admin);
			RequestDispatcher rd =
request.getRequestDispatcher("home.jsp");
			rd.forward(request, response);
		}
		else
			response.sendRedirect("invalidcred.jsp");
	}
}
```

**RegisterContoller.java**

```
package mypackage.controllers;

import java.io.IOException;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import mypackage.models.AdminModel;
import mypackage.serviceimpl.AdminServiceImpl;
import mypackage.services.AdminService;

@WebServlet("/registercontroller")
public class RegisterController extends HttpServlet {
	private static final long serialVersionUID = 1L;
```

```
	private AdminService adminService = new AdminServiceImpl();

	protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {

		String username = request.getParameter("username");
		String password = request.getParameter("password");

		AdminModel adminModel = new AdminModel(username, password);
		adminService.register(adminModel);
		RequestDispatcher rd =
request.getRequestDispatcher("registrationsuccessful.jsp");
		rd.forward(request, response);

	}

}
```

**StudentController.java**

```
package mypackage.controllers;

import java.io.IOException;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import mypackage.entities.Admin;
import mypackage.models.StudentModel;
import mypackage.serviceimpl.StudentServiceImpl;
import mypackage.services.StudentService;

@WebServlet("/studentcontroller")
public class StudentController extends HttpServlet {
	private static final long serialVersionUID = 1L;

	private StudentService studentService = new StudentServiceImpl();
```

```
	protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {

		String studentName = request.getParameter("studentname");
		String studentAddress = request.getParameter("studentaddress");
		String clazzName = request.getParameter("clazzname");

		StudentModel studentModel = new StudentModel(studentName,
studentAddress, clazzName);
		HttpSession session = request.getSession();
		Admin admin = (Admin) session.getAttribute("admin");
		studentService.addStudent(admin, studentModel);
		RequestDispatcher rd = request.getRequestDispatcher("studentslist.jsp");
		rd.forward(request, response);
	}

}
```

## SubjectController.java

```
package mypackage.controllers;

import java.io.IOException;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import mypackage.dao.ClazzDao;
import mypackage.daoimpl.ClazzDaoImpl;
import mypackage.entities.Admin;
import mypackage.entities.Clazz;
import mypackage.models.SubjectModel;
import mypackage.serviceimpl.SubjectServiceImpl;
import mypackage.services.SubjectService;

@WebServlet("/subjectcontroller")
```

```
public class SubjectController extends HttpServlet {
	private static final long serialVersionUID = 1L;

	private SubjectService subjectService = new SubjectServiceImpl();

   protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {

	String subjectName = request.getParameter("subjectname");
	String subjectLevel = request.getParameter("subjectlevel");
	String clazzName = request.getParameter("clazzname");

	SubjectModel subjectModel = new SubjectModel(subjectName, subjectLevel,
clazzName);
	HttpSession session = request.getSession();
	Admin admin = (Admin) session.getAttribute("admin");
	subjectService.addSubject(admin, subjectModel);

	RequestDispatcher rd = request.getRequestDispatcher("subjectslist.jsp");
	rd.forward(request, response);
	}
}
```

## TeacherContoller.java

```
package mypackage.controllers;

import java.io.IOException;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import mypackage.entities.Admin;
import mypackage.models.TeacherModel;
import mypackage.serviceimpl.TeacherServiceImpl;
import mypackage.services.TeacherService;
```

```
@WebServlet("/teachercontroller")
public class TeacherController extends HttpServlet {
	private static final long serialVersionUID = 1L;

	private TeacherService teacherService = new TeacherServiceImpl();

	protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {

		String teacherName = request.getParameter("teachername");
		System.out.println(teacherName);
		String qualification = request.getParameter("qualification");
		System.out.println(qualification);
		String clazzName = request.getParameter("clazzname");
		System.out.println(clazzName);

		TeacherModel teacherModel = new TeacherModel(teacherName,
qualification, clazzName);
		HttpSession session = request.getSession();
		Admin admin = (Admin) session.getAttribute("admin");
		teacherService.addTeacher(admin, teacherModel);

		RequestDispatcher rd = request.getRequestDispatcher("teacherslist.jsp");
		rd.forward(request, response);
	}

}
```

## 7. Creating Model classes

- Create a package ‘mypackage.models’ in Java Resources.
- Create java classes AdminModel.java, ClazzModel.java, LoginModel.java, StudentModel.java, SubjectModel.java and TeacherModel.java.

### AdminModel.java

```
package mypackage.models;

public class AdminModel {

	private String username;
	private String password;
```

```
	public AdminModel() {
		// TODO Auto-generated constructor stub
	}

	public AdminModel(String username, String password) {
		super();
		this.username = username;
		this.password = password;
	}

	// Getters and Setters
	public String getUsername() {
		return username;
	}

	public void setUsername(String username) {
		this.username = username;
	}

	public String getPassword() {
		return password;
	}

	public void setPassword(String password) {
		this.password = password;
	}

}
```

## ClazzModel.java

```
package mypackage.models;

public class ClazzModel {

	private String name;

	public ClazzModel() {
		// TODO Auto-generated constructor stub
	}
```

```
	public ClazzModel(String name) {
		super();
		this.name = name;
	}

	public String getName() {
		return name;
	}

	public void setName(String name) {
		this.name = name;
	}
}
```

## LoginModel.java

```
package mypackage.models;

public class LoginModel {

	private String username;
	private String password;

	public LoginModel() {
		// TODO Auto-generated constructor stub
	}

	public LoginModel(String username, String password) {
		super();
		this.username = username;
		this.password = password;
	}

	public String getUsername() {
		return username;
	}

	public void setUsername(String username) {
		this.username = username;
	}
```

```
	public String getPassword() {
		return password;
	}

	public void setPassword(String password) {
		this.password = password;
	}

}
```

## StudentModel.java

```
package mypackage.models;

public class StudentModel {

	private String name;
	private String address;
	private String cid;

	public StudentModel() {
		// TODO Auto-generated constructor stub
	}

	public StudentModel(String name, String address, String cid) {
		super();
		this.name = name;
		this.address = address;
		this.cid = cid;
	}

	public String getName() {
		return name;
	}

	public void setName(String name) {
		this.name = name;
	}

	public String getAddress() {
```

```
		return address;
	}

	public void setAddress(String address) {
		this.address = address;
	}

	public String getCid() {
		return cid;
	}

	public void setCid(String cid) {
		this.cid = cid;
	}
}
```

## SubjectModel.java

```
package mypackage.models;

public class SubjectModel {

	private String name;
	private String level;
	private String cid;

	public SubjectModel() {
		// TODO Auto-generated constructor stub
	}

	public SubjectModel(String name, String level, String cid) {
		super();
		this.name = name;
		this.level = level;
		this.cid = cid;
	}

	public String getName() {
		return name;
	}
```

```
	public void setName(String name) {
		this.name = name;
	}

	public String getLevel() {
		return level;
	}

	public void setLevel(String level) {
		this.level = level;
	}

	public String getCid() {
		return cid;
	}

	public void setCid(String cid) {
		this.cid = cid;
	}
}
```

## TeacherModel.java

```
package mypackage.models;

public class TeacherModel {

	private String name;
	private String qualification;
	private String cid;

	public TeacherModel() {
		// TODO Auto-generated constructor stub
	}

	public TeacherModel(String name, String qualification, String cid) {
		super();
		this.name = name;
		this.qualification = qualification;
		this.cid = cid;
	}
```

```
	public String getName() {
		return name;
	}

	public void setName(String name) {
		this.name = name;
	}

	public String getQualification() {
		return qualification;
	}

	public void setQualification(String qualification) {
		this.qualification = qualification;
	}

	public String getCid() {
		return cid;
	}

	public void setCid(String cid) {
		this.cid = cid;
	}
}
```

## 8. Creating Service interfaces

- Create a package ‘mypackage.services’ in Java Resources.
- Create interfaces AdminService.java, ClazzService.java, StudentService.java, SubjectService.java and TeacherService.java.
- Declare methods that define the operations to be performed on the corresponding models.

### AdminService.java

```
package mypackage.services;

import mypackage.entities.Admin;
import mypackage.models.AdminModel;
import mypackage.models.LoginModel;
```

```
public interface AdminService {

	void register(AdminModel adminModel);
	Admin getAdmin(LoginModel loginModel);
}
```

## ClazzService.java

```
package mypackage.services;

import mypackage.entities.Admin;
import mypackage.models.ClazzModel;

public interface ClazzService {

	void addClazz(Admin admin, ClazzModel clazzModel);
}
```

## StudentService.java

```
package mypackage.services;

import mypackage.entities.Admin;
import mypackage.models.StudentModel;

public interface StudentService {

	void addStudent(Admin admin, StudentModel studentModel);
}
```

## SubjectService.java

```
package mypackage.services;

import mypackage.entities.Admin;
import mypackage.models.SubjectModel;

public interface SubjectService {

	void addSubject(Admin admin, SubjectModel subjectModel);

}
```

### TeacherService.java

```
package mypackage.services;

import mypackage.entities.Admin;
import mypackage.models.TeacherModel;

public interface TeacherService {

	void addTeacher(Admin admin, TeacherModel teacherModel);
}
```

## 9. Creating ServiceImpl classes

- Create a package ‘mypackage.serviceimpl’ in Java Resources.
- Create java classes AdminServiceImpl.java, ClazzServiceImpl.java, StudentServiceImpl.java, SubjectServiceImpl.java and TeacherServiceImpl.java.
- Implement the methods declared in each service interface.

### AdminServiceImpl.java

```
package mypackage.serviceimpl;

import mypackage.dao.AdminDao;
import mypackage.daoimpl.AdminDaoImpl;
import mypackage.entities.Admin;
import mypackage.models.AdminModel;
import mypackage.models.LoginModel;
import mypackage.services.AdminService;

public class AdminServiceImpl implements AdminService{

	private AdminDao dao = new AdminDaoImpl();

	@Override
	public void register(AdminModel adminModel) {

		Admin admin = new Admin();
		admin.setUsername(adminModel.getUsername());
		admin.setPassword(adminModel.getPassword());
		dao.insert(admin);
```

```
	}

	@Override
	public Admin getAdmin(LoginModel loginModel) {

		return dao.getAdmin(loginModel.getUsername(),
loginModel.getPassword());
	}

}
```

## ClazzServiceImpl.java

```
package mypackage.serviceimpl;

import mypackage.dao.AdminDao;
import mypackage.dao.ClazzDao;
import mypackage.daoimpl.AdminDaoImpl;
import mypackage.daoimpl.ClazzDaoImpl;
import mypackage.entities.Admin;
import mypackage.entities.Clazz;
import mypackage.models.ClazzModel;
import mypackage.services.ClazzService;

public class ClazzServiceImpl implements ClazzService {

	private ClazzDao clazzDao = new ClazzDaoImpl();
	private AdminDao adminDao = new AdminDaoImpl();

	@Override
	public void addClazz(Admin admin, ClazzModel clazzModel) {

		try {
		Admin adm = adminDao.getAdmin(admin.getUsername(),
admin.getPassword());

		Clazz clazz = new Clazz();
		clazz.setClazz_name(clazzModel.getName());
		clazz.setAdmin(adm);
```

```
                adm.addClazz(clazz);
                clazzDao.add(clazz);
                } catch (Exception e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
                }
        }
}
```

## StudentServiceImpl.java

```
package mypackage.serviceimpl;

import mypackage.dao.AdminDao;
import mypackage.dao.ClazzDao;
import mypackage.dao.StudentDao;
import mypackage.daoimpl.AdminDaoImpl;
import mypackage.daoimpl.ClazzDaoImpl;
import mypackage.daoimpl.StudentDaoImpl;
import mypackage.entities.Admin;
import mypackage.entities.Clazz;
import mypackage.entities.Student;
import mypackage.models.StudentModel;
import mypackage.services.StudentService;

public class StudentServiceImpl implements StudentService{

        private StudentDao studentDao = new StudentDaoImpl();
        private AdminDao adminDao = new AdminDaoImpl();
        private ClazzDao clazzDao = new ClazzDaoImpl();

        @Override
        public void addStudent(Admin admin, StudentModel studentModel) {

                try {
                Admin adm = adminDao.getAdmin(admin.getUsername(),
admin.getPassword());

                Clazz clazz =
clazzDao.getClazzById(Integer.parseInt(studentModel.getCid()));
```

```
                Student student = new Student();
                student.setStudent_name(studentModel.getName());
                student.setAddress(studentModel.getAddress());
                student.setAdmin(adm);
                student.setClazz(clazz);

                adm.addStudent(student);
                studentDao.add(student);
                } catch (Exception e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
                }
        }
}
```

## SubjectServiceImpl.java

```
package mypackage.serviceimpl;

import mypackage.dao.AdminDao;
import mypackage.dao.ClazzDao;
import mypackage.dao.SubjectDao;
import mypackage.daoimpl.AdminDaoImpl;
import mypackage.daoimpl.ClazzDaoImpl;
import mypackage.daoimpl.SubjectDaoImpl;
import mypackage.entities.Admin;
import mypackage.entities.Clazz;
import mypackage.entities.Subject;
import mypackage.models.SubjectModel;
import mypackage.services.SubjectService;

public class SubjectServiceImpl implements SubjectService {

        private SubjectDao subjectDao = new SubjectDaoImpl();
        private AdminDao adminDao = new AdminDaoImpl();
        private ClazzDao clazzDao = new ClazzDaoImpl();

        @Override
        public void addSubject(Admin admin, SubjectModel subjectModel) {

                try {
```

```
                    Admin adm = adminDao.getAdmin(admin.getUsername(),
admin.getPassword());

                    Clazz clazz =
clazzDao.getClazzById(Integer.parseInt(subjectModel.getCid()));
                    Subject subject = new Subject();
                    subject.setSubject_name(subjectModel.getName());
                    subject.setSubject_level(subjectModel.getLevel());
                    subject.setAdmin(adm);
                    subject.setClazz(clazz);

                    adm.addSubject(subject);
                    subjectDao.add(subject);
                } catch (Exception e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
                }
        }

}
```

## TeacherServiceImpl.java

```
package mypackage.serviceimpl;

import java.util.HashSet;
import java.util.Set;

import mypackage.dao.AdminDao;
import mypackage.dao.ClazzDao;
import mypackage.dao.TeacherDao;
import mypackage.daoimpl.AdminDaoImpl;
import mypackage.daoimpl.ClazzDaoImpl;
import mypackage.daoimpl.TeacherDaoImpl;
import mypackage.entities.Admin;
import mypackage.entities.Clazz;
import mypackage.entities.Teacher;
import mypackage.models.TeacherModel;
import mypackage.services.TeacherService;

public class TeacherServiceImpl implements TeacherService {
```

```
	private AdminDao adminDao = new AdminDaoImpl();
	private TeacherDao teacherDao = new TeacherDaoImpl();
	private ClazzDao clazzDao = new ClazzDaoImpl();

	@Override
	public void addTeacher(Admin admin, TeacherModel teacherModel) {

		try {
			Admin adm = adminDao.getAdmin(admin.getUsername(),
admin.getPassword());

			Clazz clazz =
clazzDao.getClazzById(Integer.parseInt(teacherModel.getCid()));
			Teacher teacher = new Teacher();
			teacher.setTeacher_name(teacherModel.getName());
			teacher.setTeacher_qual(teacherModel.getQualification());
			teacher.setAdmin(adm);

			Set<Clazz> clazzSet = new HashSet<>();
			clazzSet.add(clazz);
			teacher.setClazzes(clazzSet);

			adm.addTeacher(teacher);
			teacherDao.add(teacher);
		} catch (Exception e) {
			// TODO Auto-generated catch block
			e.printStackTrace();
		}
	}
}
```

## 10. Creating Filter

- Create a package ‘mypackage.filters’ in Java Resources.
- Create a filter HeaderHomeFilter.java which blocks anyone from accessing the website through home button from header without logging in.

### HeaderHomeFilter.java

```
package mypackage.filters;
```

```
import javax.servlet.*;
import javax.servlet.annotation.WebFilter;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import java.io.IOException;

@WebFilter("/*")
public class HeaderHomeFilter implements Filter {

    public void init(FilterConfig fConfig) throws ServletException {
    }

    public void doFilter(ServletRequest request, ServletResponse response, FilterChain
chain) throws IOException, ServletException {
        HttpServletRequest httpRequest = (HttpServletRequest) request;
        HttpServletResponse httpResponse = (HttpServletResponse) response;

        HttpSession session = httpRequest.getSession(false);

        // Check if the session is null or invalid
        if (session == null || session.isNew()) {
            // Redirect to index.jsp unless it is already the requested page
            String requestedPage = httpRequest.getRequestURI();
            if (!requestedPage.endsWith("index.jsp")) {
                httpResponse.sendRedirect("index.jsp");
                return;
            }
        }

        chain.doFilter(request, response);
    }

    public void destroy() {
    }
}
```