

***“MCQ TEST SFTWARE”***

**Mini Project Report**

**Submitted in partial fulfillment of the requirements of the subject Object Oriented**

**Programming Method lab by**

**Kush Gada (SE-14)**

**Meet Bhanushali (SE-05)**

Supervisor

**Prof.Amit Kukeraja**



**Department of Electronic and Tele-Communication**

**K.J. Somaiya Institute of Engineering and Information Technology**

**Ayurvihar, Sion Mumbai-400022**

**2019**

 **CERTIFICATE** 

*This is to certify that the project entitled* ***“MCQ TEST SFTWARE”*** *is a bonafide work of Student Kush Gada, Student Meet Bhanushali submitted as mini project in the subject of OOPJ* ***Lab*** *in* ***“Electroics and Tele Communication Enginering”.***

Prof. Amit Kukeraja

(Project Guide)

**DECLARATION**

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Kush Gada

Meet Bhanushali

Date :

Place :

Before presenting our project entitled “Ledger System”, we would like to convey our sincere thanks to many people who guided us throughout the course for this seminar work.

First, we would like to express our sincere thanks to our beloved Principal **Dr. SURESH UKARANDE** for providing various facilities to carry out this report.

We would like to express our sincere thanks to **Prof. AMIT KUKREJA** for her guidance, encouragement, co-operation and suggestions given to us at progressing stages of report.

Finally, we would like to thank our **H.O.D. Prof. JAYASHREE KHANAPURI** and all teaching, non-teaching staff of the college and friends for their moral support rendered during the course of the report work and for their direct and indirect involvement in the completion of our report work, which made our endeavor fruitful.

Kush Gada

Meet Bhanushali

**INDEX**

|  |  |  |
| --- | --- | --- |
| SERIAL NO | TITLE | PAGE NO |
| 1) | PROBLEM DEFINITION | 1 |
| 2) | THEORY | 1 |
| 3) | USER INTERFACE FLOW | 10 |
| 4) | SCREEN SHOTS | 11 |
| 5) | CONCLUSION | 18 |
| 6) | REFERENCE | 19 |

Create a Graphical User Interface(GUI) using Java to implement the MCQ TEST SFTWARE. You should be able to Register account , select subject ,give test on that particular subject and View score .

**THEORY**

# Java Swing Tutorial

**Java Swing tutorial** is a part of Java Foundation Classes (JFC) that is used to create window-based applications. It is built on the top of AWT (Abstract Windowing Toolkit) API and entirely written in java.

Unlike AWT, Java Swing provides platform-independent and lightweight components.

The javax.swing package provides classes for java swing API such as JButton, JTextField, JTextArea, JRadioButton, JCheckbox, JMenu, JColorChooser etc.

### Difference between AWT and Swing

There are many differences between java awt and swing that are given below.

|  |  |  |
| --- | --- | --- |
| **No.** | **Java AWT** | **Java Swing** |
| 1) | AWT components are **platform-dependent**. | Java swing components are **platform-independent**. |
| 2) | AWT components are **heavyweight**. | Swing components are **lightweight**. |
| 3) | AWT **doesn't support pluggable look and feel**. | Swing **supports pluggable look and feel**. |
| 4) | AWT provides **less components** than Swing. | Swing provides **more powerful components** such as tables, lists, scrollpanes, colorchooser, tabbedpane etc. |
| 5) | AWT **doesn't follows MVC**(Model View Controller) where model represents data, view represents presentation and controller acts as an interface between model and view. | Swing **follows MVC**. |

### What is JFC

The Java Foundation Classes (JFC) are a set of GUI components which simplify the development of desktop applications.

### Hierarchy of Java Swing classes

The hierarchy of java swing API is given below.



### Commonly used Methods of Component class

The methods of Component class are widely used in java swing that are given below.

|  |  |
| --- | --- |
| **Method** | **Description** |
| public void add(Component c) | add a component on another component. |
| public void setSize(int width,int height) | sets size of the component. |
| public void setLayout(LayoutManager m) | sets the layout manager for the component. |
| public void setVisible(boolean b) | sets the visibility of the component. It is by default false. |

# Java JButton

The JButton class is used to create a labeled button that has platform independent implementation. The application result in some action when the button is pushed. It inherits AbstractButton class.

# Java JLabel

The object of JLabel class is a component for placing text in a container. It is used to display a single line of read only text. The text can be changed by an application but a user cannot edit it directly. It inherits JComponent class.

# Java JTextField

The object of a JTextField class is a text component that allows the editing of a single line text. It inherits JTextComponent class.

# Java JPasswordField

The object of a JPasswordField class is a text component specialized for password entry. It allows the editing of a single line of text. It inherits JTextField class.

# Java JRadioButton

The JRadioButton class is used to create a radio button. It is used to choose one option from multiple options. It is widely used in exam systems or quiz.

It should be added in ButtonGroup to select one radio button only.

# Java JOptionPane

The JOptionPane class is used to provide standard dialog boxes such as message dialog box, confirm dialog box and input dialog box. These dialog boxes are used to display information or get input from the user. The JOptionPane class inherits JComponent class.

# Java JDBC Tutorial

JDBC stands for Java Database Connectivity. JDBC is a Java API to connect and execute the query with the database. It is a part of JavaSE (Java Standard Edition). JDBC API uses JDBC drivers to connect with the database.

**WHAT IS A GUI?**

A graphical user interface (GUI) is a human-computer interface (i.e., a way for humans to interact with computers) that uses [windows,](http://www.linfo.org/window.html) [icons and](http://www.linfo.org/icon.html) menus and which can be manipulated by a mouse (and often to a limited extent by a keyboard as well).

**Advantages of GUIs :**

1)A major advantage of GUIs is that they make computer operation more intuitive, and thus easier to learn and use. For example, it is much easier for a new user to move a file from one directory to another by dragging its icon with the mouse than by having to remember and type seemingly arcane commands to accomplish the same task.

2)Adding to this intuitiveness of operation is the fact that GUIs generally provide users with immediate, visual feedback about the effect of each action. For example, when a user deletes an icon representing a file, the icon immediately disappears, confirming that the file has been deleted (or at least sent to the trash can). This contrasts with the situation for a CLI, in which the user types a delete command (inclusive of the name of the file to be deleted) but receives no automatic feedback indicating that the file has actually been removed.

3)In addition, GUIs allow users to take full advantage of the powerful [multitasking (the](http://www.linfo.org/multitasking.html) ability for multiple programs and/or multiple instances

of single programs to run simultaneously) capabilities of modern [operating](http://www.linfo.org/operating_systems_list.html) [systems by](http://www.linfo.org/operating_systems_list.html) allowing such multiple programs and/or instances to be displayed simultaneously. The result is a large increase in the flexibility of computer use and a consequent rise in user productivity.

4)But the GUI has became much more than a mere convenience. It has also become the standard in human-computer interaction, and it has influenced the work of a generation of computer users. Moreover, it has led to the development of new types of applications and entire new industries. An example is desktop publishing, which has revolutionized (and partly wiped out) the traditional printing and typesetting industry.

**DATABASE**

A database is an organized collection of data. A database-management system (DBMS) is a computer-software application that interacts with end-users, other applications, and the database itself to capture and analyze data. A general-purpose DBMS allows the definition, creation, querying, update, and administration of databases. Well-known DBMSs include MySQL, PostgreSQL, EnterpriseDB

MongoDB, MariaDB, Microsoft SQL Server, Oracle, Sybase, SAP HANA, MemSQL, SQLite and IBM DB2.

The DBMS provides various functions that allow entry, storage and retrieval of large quantities of information and provides ways to manage how that information is organized.

Existing DBMSs provide various functions that allow management of a database and its data which can be classified into four main functional groups:

**Data definition –**

Creation, modification and removal of definitions that define the organization of the data.

**Update –**

Insertion, modification, and deletion of the actual data.

**Retrieval –**

Providing information in a form directly usable or for further processing by other applications. The retrieved data may be made available in a form basically the same as it is stored in the database or in a new form obtained by altering or combining existing data from the database.

**Administration –**

Registering and monitoring users, enforcing data security, monitoring performance, maintaining data integrity, dealing with concurrency control, and recovering information that has been corrupted by some event such as an unexpected system failure.

**MySQL :**

What is MySQL?

1)MySQL is a database system, used for developing web based software

applications.

MySQL used for both small and large applications

2)MySQL is a relational database management system (RDBMS).

3)MySQL is fast reliable and flexible and easy to use.

4)MySQL supports standard SQL (Structured Query Language).

**Main Features of MySQL**

1)MySQL server design is multi-layered with independent modules

2)MySQL is fully multithreaded by using kernel threads. It can use multiple CPUs if they are available.

3)MySQL provides transactional and non-transactional storage engines.

4)MySQL has very fast thread-based memory allocation system.

5)MySQL supports in-memory heap table.

6)MySQL Handles large databases.

7)MySQL Server works in client/server or embedded systems.

8)MySQL Works on many different platforms.

**Creating a database**

1)mysql -u root -p

Enter the password: \*\*\*\*

2)create database user\_defined\_name;

3)use database\_name;

4)create table table\_name(variable\_name data\_type)

**Inserting into database**

insert into table\_name values(parameter\_list);

**USER INTERFACE FLOW**

When program starts form with main mcq test software the registered user has to login into his account after logging he will be moved on to the menu page where he has to select the topic on which he wants to give the test after selectinf the topic he will be moved on to the test page from where the test begins .

**View Score :** On clicking of this option the user can see the test results of the previous exam he/she gave .

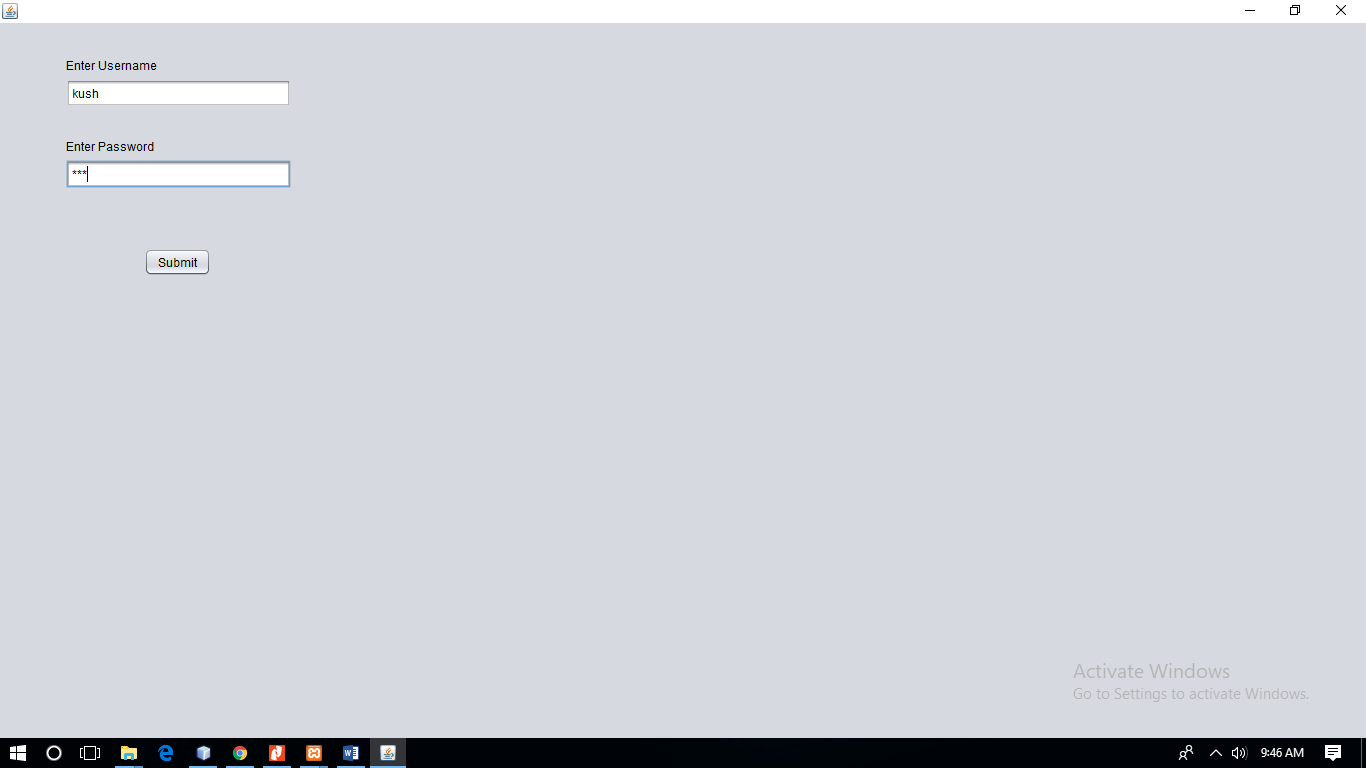
**Give Test:** On clicking of this button the user can give the test .

**DropDown Box:** On clicking of this option user will be able to choose the topic on which he wants to attempt the test.

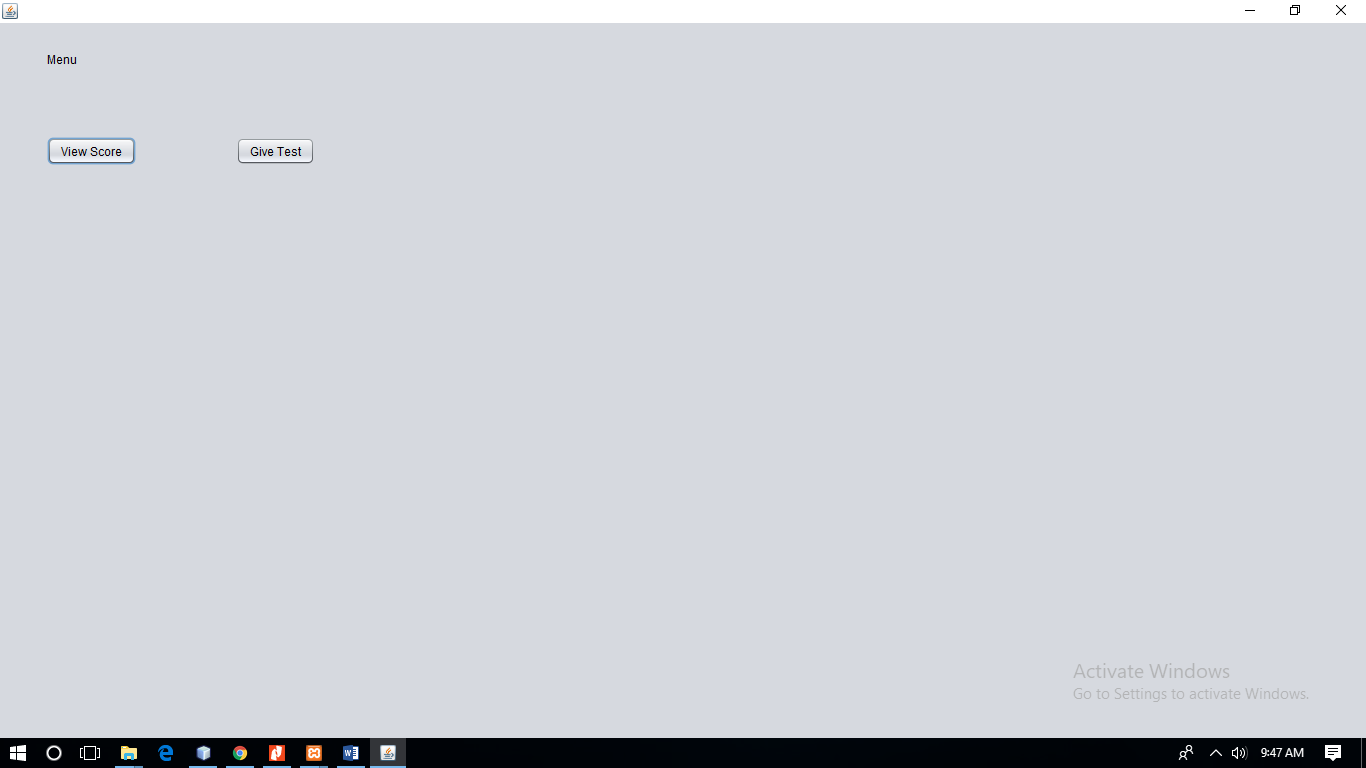
**Submit:** On clicking of this option the user verifies the selected topic of test and proceed to give the test .

**SCREENSHOTS**

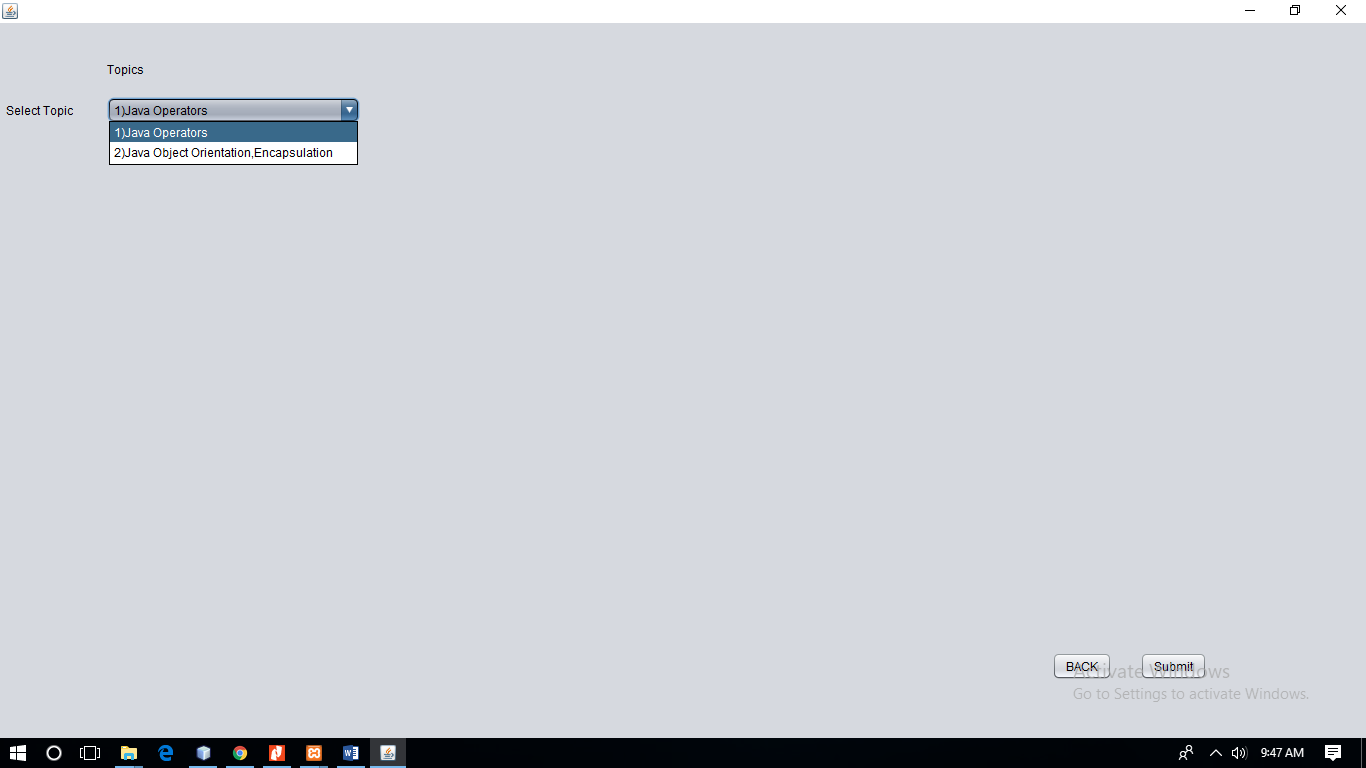
**LOGIN PAGE**



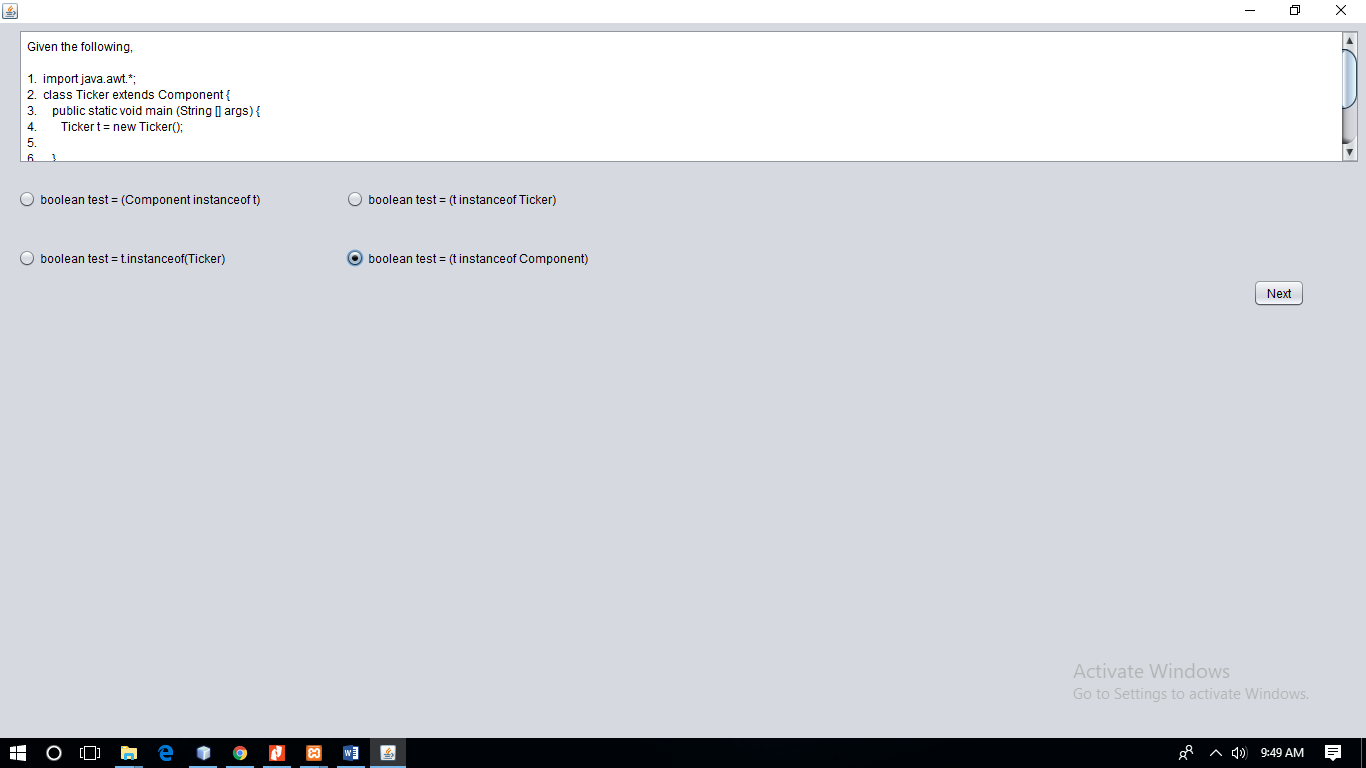
**MENU**



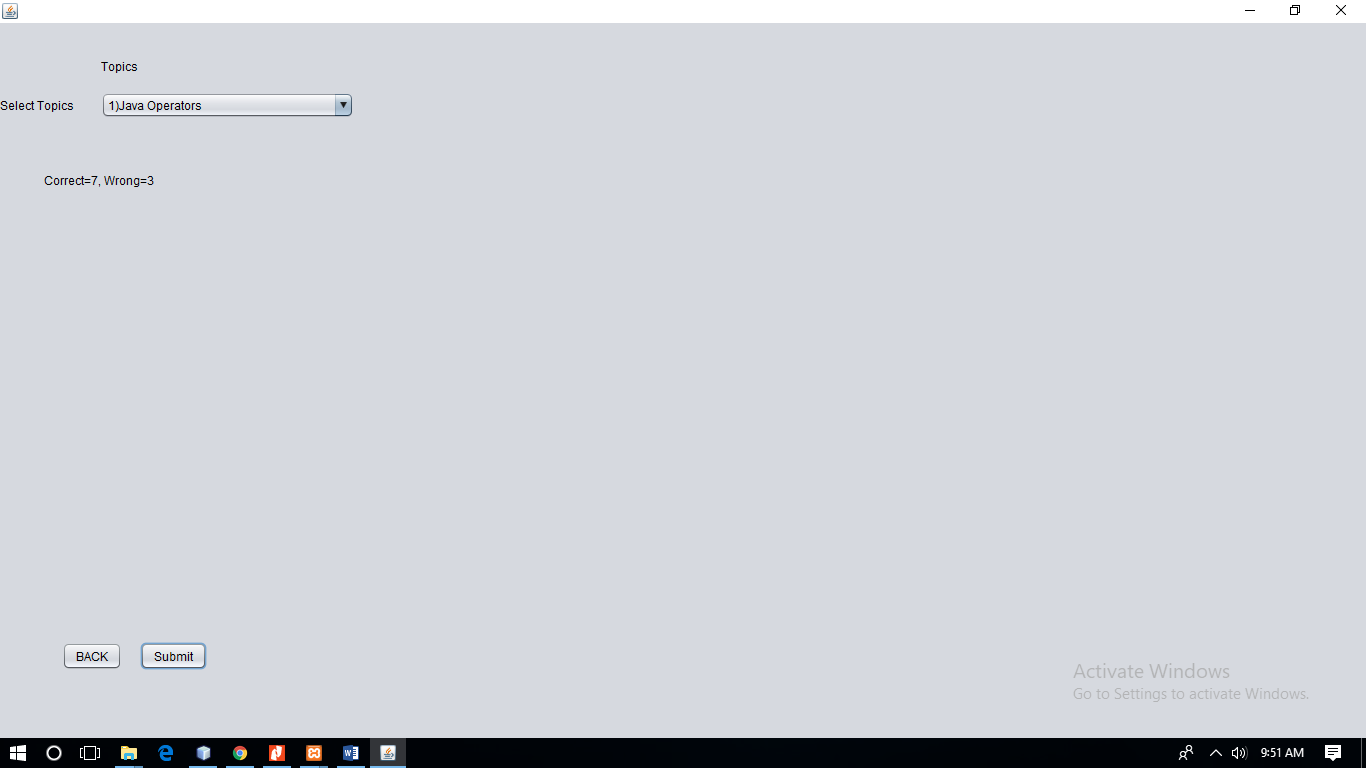
**TEST MENU**



**QUESTIONS**



**SCORE**



**CONCLUSION**

Implementing this project helped us learn a lot about java and its uses. We learnt how to create a java program, use various types of layouts, make use of text fields, labels, buttons, etc., how to create an event and making use of action listener for event handling to use the applet as a GUI, make use of JDBC to link applet to the Mysql database etc. The Ledger System is backbone of any company. This project helped us to understand nuances of ledger system used by almost all the companies. Thus this mini project by implementing java program as front end and Mysql(database) as back end was achieved and tested accordingly.

**REFERENCE**

**Text books:**

● Herbert Schildt, ‘JAVA: The Complete Reference’, Ninth Edition, Oracle

Press.

**●** Sachin Malhotra and Saurabh Chaudhary, “Programming in Java”, Oxford

University Press, 2010

**Reference Books:**

● Ivor Horton, ‘Beginning JAVA’, Wiley India.

● DietalandDietal, ‘Java: How to Program’, 8/e,PHI

● ‘JAVA Programming’, Black Book, Dreamtech Press.

**●** ‘Learn to Master Java Programming’, Stardusolutions

**Digital Material :**

● www.nptelvideos.in

● www.w3schools.com

● http://spoken-tutorial.org

● www.staredusolutions.org