

PAWN MANAGEMENT SYSTEM

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Degree in Computer Science

By

Dilli Kumaran.S(Reg.no 39290020)

Nitish Kumar.R(Reg.no 39290075)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING SCHOOL OF
COMPUTING**

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

JEPPIAAR NAGAR, RAJIV GANDHI SALAI,

CHENNAI – 600119, TAMILNADU

APRIL 2022



**SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade “A” by NAAC

established under Section 3 of UGC Act, 1956)

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI- 600119

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of DILLI KUMARAN.S(Reg. No. 39290020) and NITHISH KUMAR.S (Reg. No. 38290104) who carried out the project entitled “PAWN MANAGEMENT SYSTEM” under my supervision from November 2021 to April 2022.

Internal Guide

Dr.Asha

Head of the Department

Dr. L. Lakshmanan, M.E., Ph.D.

Submitted for Viva voce Examination held on

DECLARATION

I, DILLI KUMARAN (Reg. No. 39290020) hereby declare that the Project Report entitled "PAWN MANAGEMENT SYSTEM" done by me under the guidance of Dr.ASHA is submitted in partial fulfillment of the requirements for the award of Bachelor of Science degree in Computer Science.

DATE:

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to Board of Management of SATHYABAMA for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to Dr. L. LAKSHMANAN, M.E., Ph.D., Head of the Department, Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide Dr.Asha for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the Department of Computer Science and Engineering who were helpful in many ways For the Completion of the project.

ABSTRACT

Pawn management system project is implemented in java platform using android app application.

- Main aim of this project is to develop a online application for pawn brokers company to atomize work procedure, using this system agents and customers can know details about present pledged item, release history, auction procedure,, terms and conditions and pledging registration by the customers.
- In existing system manual procedure is followed where records are used to maintain data which is a time taking process and require more man power and calculating commissions dues..etc are done manually. The proposed system where staffs send sms alert to the customer to remind before and exceeds the data of due.
- In present system there is no need of human interference in calculating any details. Total work is done using management system which will save time and less paper work and even human resource.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE
	ABSTRACT	5
1	CHAPTER - 1	
1.1	INTRODUCTION	8
1.2	ADVANTAGES	9
1.3	DISADVANTAGES	9
	CHAPTER - 2	
2	LITERATURE SURVEY	10
	CHAPTER - 3	
3	METHODOLOGY	
3.1	HARDWARE REQUIREMENTS	13
3.2	SOFTWARE REQUIREMENTS	13
3.3	FRONT END(look and feel)	13
3.4	DATA ACCESS LAYER	22
3.5	BACK END(data control)	24
	CHAPTER - 4	
4.1	MODULES	25
5	APPLICATION PERFORMANCE	26
	CHAPTER - 6	
6.1	CONCLUSION	30
6.2	SCOPE OF DEVELOPEMENT	30
6.3	REFERENCES	30
	CHAPTER - 7	
7.1	APPENDIX	35

LIST OF FIGURES

FIGURE NO NO.	FIGURE NAME	PAGE
5.1	LOGIN PAGE	
5.2	HOME PAGE	
5.3	NEW BILL	
5.4	NEW CUSTOMER REGISTRATION	
5.5	VIEW DETAILS BY BILL	
5.6	VIEW DETAILS BY NAME	
5.7	PLEDGE BOOK	
5.8	DAY VIEW	

CHAPTER 1

1.1 INTRODUCTION

OVERVIEW OF PROJECT

EXISTING SYSTEM :

In the existing pawn Management System, the work is done by hand.

All the details for the pawn broking such as cash information and history of customers and confidential biodata information was collected into the hard copy and by chance if any of the document get missed up or get harmed then whole of the information will be missed, resulted into the major loss for the business people to ask for the details again or knowing the customer's healthy transaction with the broking company.

Also adding all the details manually will take a lot of time and also a lot more chances of entering the information wrong.

And also sending details from one place to another will not be any task because in case while sending one important document from one place to another, it get lost, then also it will proved to a great loss both for the user as well as for the business.

PROPOSED SYSTEM:

In the proposed pawn Management System, all the work will be digitalized and is done via computers and internet.

All the details regarding the customers and pawn tickets will be added via computer and the information data is being saved in servers.

Backup should be there in case if by chance any of the information will be lost. Time consume will be reduced and customer will be knowing thier pledging related informations in fraction of seconds.

Owner just have to click on the button and just have to wait for some moments and they get an easy access to their information.

1.2 ADVANTAGES

Time consume will be reduce

Users will get any easy way to access their pledging related information

.

Users just have to click on the button and just have to wait for some moments and they get an easy access to their information.

1.4 DISADVANTAGES

Also adding all the details manually will take a lot of time

also a lot more chances of entering the information wrong.

while sending one important document from one place to another, it get lost, then also it will proved to a great loss both for the user as well as for the organization.

CHAPTER 2

2. LITERATURE SURVEY

1.) RepairQ by service central technologies:

RepairQ is an end-to-end business management platform built for repair shops. With powerful features, integrations, and POS functionality, RepairQ is the go to choice for growing repair shops around the globe.

Initially built for the retail repair industry, and focused on mobile device and electronic repair, RepairQ has recently undergone major changes to make our full-stop business management platform an effective tool for ANY repair-oriented organization.

RepairQ boasts all native, english-speaking support team members, customized onboarding options, and scalable solutions for any level of business; so we can help get you up and running quickly and efficiently.

With numerous, unique integrations, industry leading customer engagement tools, and more features than you could shake a stick at, RepairQ will help take your repair shop to the next level.

Features:

- Accounting
- CRM
- Inventory Management
- Commission management
- Inventory control
- Work order management
- Customer DataBase
- Barcode Scanning
- Employee Management
- Point of Sale (POS)
- Electronic Signature
- Multi-Location
- Supplier Management
- Billing & Invoicing

- Returns Management

2.) Pawnbroker by ppss:

Computer software, hardware and IT support for pawn shops. Pawn, buy outright, consign and rent. Tiered in features and price. No prepayment for customer support. Print tickets on plain paper with any ink printer.

Features:

- Inventory Management
- Point of Sale (POS)
- Equipment Management
- Data Import & Export
- Client Management

3.) ShopKeep by lightspeed:

Shopkeep has well-designed POS solution that manages your inventory, with handy Bluetooth barcode scanners. The POS system for the retail store has portable barcode scanners and barcode printer. It imports the bulk inventory in your system just by taking few minutes and keeps it up-to-date.

Features :

- Email Marketing / SMS Marketing
- Inventory Management
- Payment Handling
- Table Management
- Stock Management
- Loyalty Management
- Cash Management
- Channel Management
- Billing & Invoicing

- Dashboard
- Billing Management
- Reporting
- Ordering Automation
- Pricing Management
- Restaurant POS

CHAPTER 3

3.1 HARDWARE REQUIREMENTS:

Minimum hardware requirement to download Java on your Windows operating system as follows:

- Minimum Windows 95 software
- IBM-compatible 486 system
- Hard Drive and Minimum of 8 MB memory
- A CD-ROM drive
- Mouse, keyboard and sound card, if required

3.2 SOFTWARE REQUIREMENTS :

Java is supported by almost every operating systems. whether it is a Windows, Macintosh and Unix all supports the Java application development. So you can download any of the operating system on your personal computer. Here are the minimum requirement.

- Operating System
- Java SDK or JRE 1.6 or higher
- Java Servlet Container (Free Servlet Container available)
- Supported Database and library that supports the database connection with Java.

3.3 FRONT END(look and feel):

We use java swing for the front end works

Java is a **programming language** and a **platform**. Java is a high level, robust, object-oriented and secure programming language.

Java was developed by Sun Microsystems (which is now the subsidiary of Oracle) in the year 1995. James Gosling is known as the father of Java. Before Java, its name was Oak. Since Oak was already a registered company, so James Gosling and his team changed the name from Oak to Java.

Platform: Any hardware or software environment in which a program runs, is known as a platform. Since Java has a runtime environment (JRE) and API, it is called a platform.

According to Sun, 3 billion devices run Java. There are many devices where Java is currently used. Some of them are as follows:

1. Desktop Applications such as acrobat reader, media player, antivirus, etc.
2. Web Applications such as irctc.co.in, javatpoint.com, etc.
3. Enterprise Applications such as banking applications.
4. Mobile
5. Embedded System
6. Smart Card
7. Robotics
8. Games, etc.

Types of Java Applications

There are mainly 4 types of applications that can be created using Java programming:

1) Standalone Application

Standalone applications are also known as desktop applications or window-based applications. These are traditional software that we need to install on every machine. Examples of standalone application are Media player, antivirus, etc. AWT and Swing are used in Java for creating standalone applications.

2) Web Application

An application that runs on the server side and creates a dynamic page is called a web application. Currently, [Servlet](#), [JSP](#), [Struts](#), [Spring](#), [Hibernate](#), [JSF](#), etc. technologies are used for creating web applications in Java.

3) Enterprise Application

An application that is distributed in nature, such as banking applications, etc. is called an enterprise application. It has advantages like high-level security, load balancing, and clustering. In Java, [EJB](#) is used for creating enterprise applications.

4) Mobile Application

An application which is created for mobile devices is called a mobile application. Currently, Android and Java ME are used for creating mobile applications.

Java Platforms / Editions

There are 4 platforms or editions of Java:

1) Java SE (Java Standard Edition)

It is a Java programming platform. It includes Java programming APIs such as java.lang, java.io, java.net, java.util, java.sql, java.math etc. It includes core topics like OOPs, [String](#), [Regex](#), [Exception](#), [Inner classes](#), [Multithreading](#), [I/O Stream](#), [Networking](#), [AWT](#), [Swing](#), [Reflection](#), [Collection](#), etc.

2) Java EE (Java Enterprise Edition)

It is an enterprise platform that is mainly used to develop web and enterprise applications. It is built on top of the Java SE platform. It includes topics like Servlet, JSP, Web Services, EJB, JPA, etc.

3) Java ME (Java Micro Edition)

It is a micro platform that is dedicated to mobile applications.

4) JavaFX

It is used to develop rich internet applications. It uses a lightweight user interface API. the basics of OOPs. Object-Oriented Programming is a paradigm that provides many concepts, such as inheritance, data binding, polymorphism, etc.

Simula is considered the first object-oriented programming language. The programming paradigm where everything is represented as an object is known as a truly object-oriented programming language.

Smalltalk is considered the first truly object-oriented programming language.

The popular object-oriented languages are Java

, C#, PHP, Python, C++, etc.

The main aim of object-oriented programming is to implement real-world entities, for example, object, classes, abstraction, inheritance, polymorphism, etc.

OOPs (Object-Oriented Programming System)

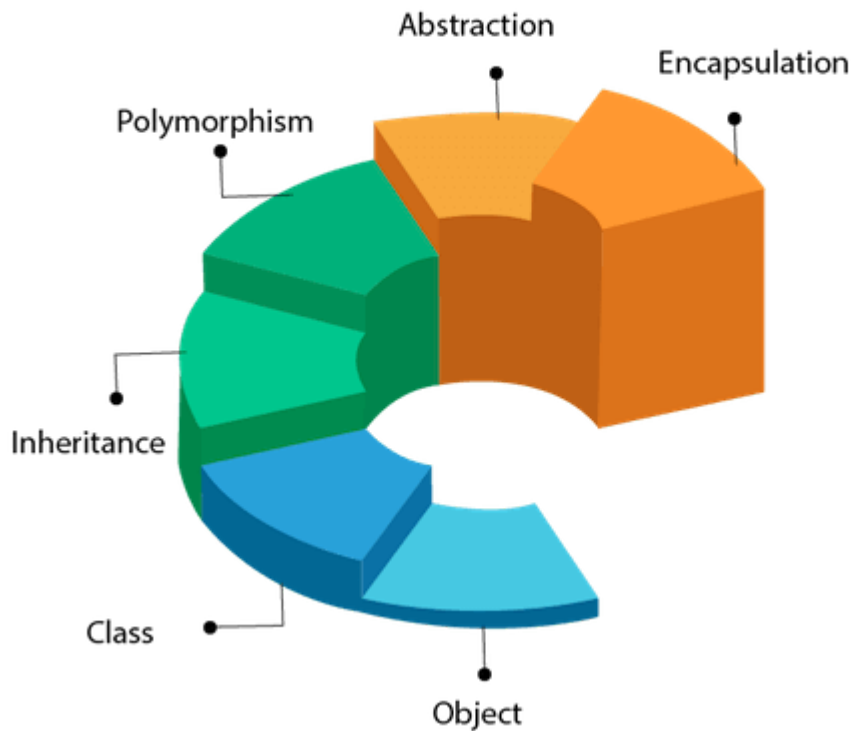
Object means a real-world entity such as a pen, chair, table, computer, watch, etc. **Object-Oriented Programming** is a methodology or paradigm to design a program using classes and objects. It simplifies software development and maintenance by providing some concepts:

- Object
- Class
- Inheritance
- Polymorphism
- Abstraction
- Encapsulation

Apart from these concepts, there are some other terms which are used in Object-Oriented design:

- Coupling
- Cohesion
- Association
- Aggregation
- Composition

OOPs (Object-Oriented Programming System)



Object



Any entity that has state and behavior is known as an object. For example, a chair, pen, table, keyboard, bike, etc. It can be physical or logical.

An Object can be defined as an instance of a class. An object contains an address and takes up some space in memory. Objects can communicate without knowing the

details of each other's data or code. The only necessary thing is the type of message accepted and the type of response returned by the objects.

Example: A dog is an object because it has states like color, name, breed, etc. as well as behaviors like wagging the tail, barking, eating, etc.

Class

Collection of objects is called class. It is a logical entity.

A class can also be defined as a blueprint from which you can create an individual object. Class doesn't consume any space.

Inheritance

When one object acquires all the properties and behaviors of a parent object, it is known as inheritance. It provides code reusability. It is used to achieve runtime polymorphism.



Polymorphism

If one task is performed in different ways, it is known as polymorphism. For example: to convince the customer differently, to draw something, for example, shape, triangle, rectangle, etc.

In Java, we use method overloading and method overriding to achieve polymorphism. Another example can be to speak something; for example, a cat speaks meow, dog barks woof, etc.

Abstraction

Hiding internal details and showing functionality is known as abstraction. For example phone call, we don't know the internal processing.

In Java, we use abstract class and interface to achieve abstraction.



Capsule

Encapsulation

Binding (or wrapping) code and data together into a single unit are known as encapsulation. For example, a capsule, it is wrapped with different medicines.

A java class is the example of encapsulation. Java bean is the fully encapsulated class because all the data members are private here.

Coupling

Coupling refers to the knowledge or information or dependency of another class. It arises when classes are aware of each other. If a class has the details information of another class, there is strong coupling. In Java, we use private, protected, and public modifiers to display the visibility level of a class, method, and field. You can use interfaces for the weaker coupling because there is no concrete implementation.

Cohesion

Cohesion refers to the level of a component which performs a single well-defined task. A single well-defined task is done by a highly cohesive method. The weakly cohesive method will split the task into separate parts. The java.io package is a highly cohesive package because it has I/O related classes and interface. However, the java.util package is a weakly cohesive package because it has unrelated classes and interfaces.

Association

Association represents the relationship between the objects. Here, one object can be associated with one object or many objects. There can be four types of association between the objects:

- One to One
- One to Many
- Many to One, and
- Many to Many

Let's understand the relationship with real-time examples. For example, One country can have one prime minister (one to one), and a prime minister can have many ministers (one to many). Also, many MP's can have one prime minister (many to one), and many ministers can have many departments (many to many).

Association can be undirectional or bidirectional.

Aggregation

Aggregation is a way to achieve Association. Aggregation represents the relationship where one object contains other objects as a part of its state. It represents the weak relationship between objects. It is also termed as a has-a relationship in Java. Like, inheritance represents the is-a relationship. It is another way to reuse objects.

Composition

The composition is also a way to achieve Association. The composition represents the relationship where one object contains other objects as a part of its state. There is a strong relationship between the containing object and the dependent object. It is the state where containing objects do not have an independent existence. If you delete the parent object, all the child objects will be deleted automatically.

Advantage of OOPs over Procedure-oriented programming language

- 1) OOPs makes development and maintenance easier, whereas, in a procedure-oriented programming language, it is not easy to manage if code grows as project size increases.
- 2) OOPs provides data hiding, whereas, in a procedure-oriented programming language, global data can be accessed from anywhere.

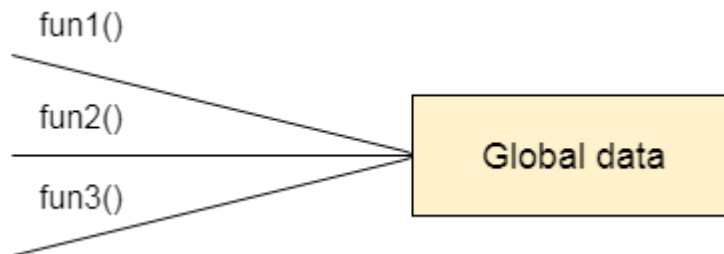


Figure: Data Representation in Procedure-Oriented Programming

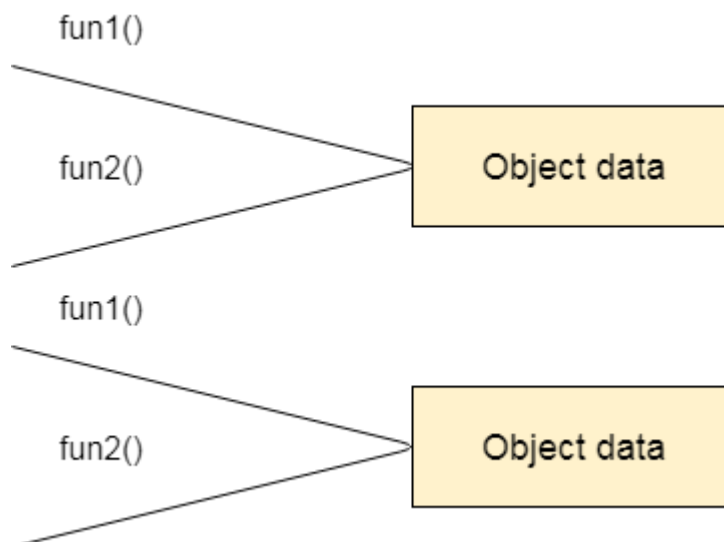


Figure: Data Representation in Object-Oriented Programming

- 3) OOPs provides the ability to simulate real-world event much more effectively. We can provide the solution of real word problem if we are using the Object-Oriented Programming language.
-

What is the difference between an object-oriented programming language and object-based programming language?

Object-based programming language follows all the features of OOPs except Inheritance. JavaScript and VBScript are examples of object-based programming languages.

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 compone 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 fe
4)	AWT provides less components than Swing.	Swing provides more power components such as tables, lis scrollpanes, colorchooser, tabbedpa 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.

3.4 DATA ACCESS LAYER :

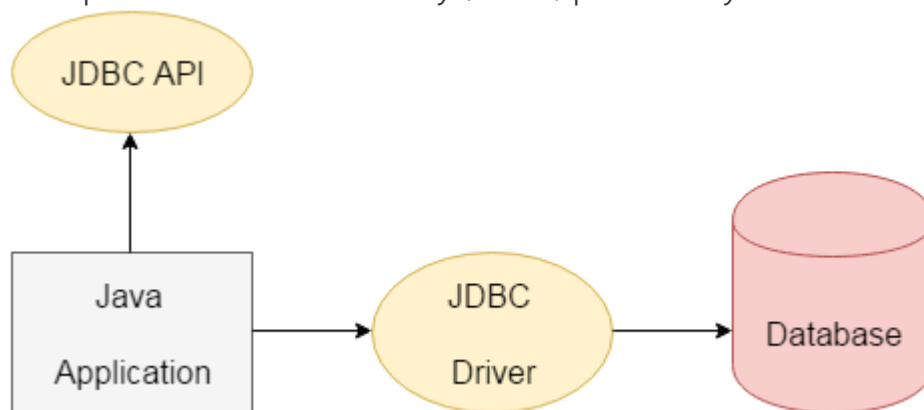
WE USE java data base connectivity for data access and control functionality

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. There are four types of JDBC drivers:

- JDBC-ODBC Bridge Driver,
- Native Driver,
- Network Protocol Driver, and
- Thin Driver

We have discussed the above four drivers in the next chapter.

We can use JDBC API to access tabular data stored in any relational database. By the help of JDBC API, we can save, update, delete and fetch data from the database. It is like Open Database Connectivity (ODBC) provided by Microsoft.



The current version of JDBC is 4.3. It is the stable release since 21st September, 2017. It is based on the X/Open SQL Call Level Interface. The **java.sql** package contains classes and interfaces for JDBC API. A list of popular interfaces of JDBC API are given below:

- Driver interface
- Connection interface
- Statement interface
- PreparedStatement interface
- CallableStatement interface
- ResultSet interface
- ResultSetMetaData interface
- DatabaseMetaData interface
- RowSet interface

A list of popular classes of JDBC API are given below:

- DriverManager class
- Blob class
- Clob class

- Types class

Why Should We Use JDBC

Before JDBC, ODBC API was the database API to connect and execute the query with the database. But, ODBC API uses ODBC driver which is written in C language (i.e. platform dependent and unsecured). That is why Java has defined its own API (JDBC API) that uses JDBC drivers (written in Java language).

We can use JDBC API to handle database using Java program and can perform the following activities:

1. Connect to the database
2. Execute queries and update statements to the database
3. Retrieve the result received from the database.

Do You Know

- How to connect Java application with Oracle and Mysql database using JDBC?
- What is the difference between Statement and PreparedStatement interface?
- How to print total numbers of tables and views of a database using JDBC?
- How to store and retrieve images from Oracle database using JDBC?
- How to store and retrieve files from Oracle database using JDBC?

What is API

API (Application programming interface) is a document that contains a description of all the features of a product or software. It represents classes and interfaces that software programs can follow to communicate with each other. An API can be created for applications, libraries, operating systems, etc.

3.5 BACK END:

We have used MYSQL for database functionality working with tables and procedures

MySQL is a very popular open-source relational database management system (RDBMS).

What is MySQL?

- MySQL is a relational database management system
- MySQL is open-source
- MySQL is free
- MySQL is ideal for both small and large applications
- MySQL is very fast, reliable, scalable, and easy to use
- MySQL is cross-platform

- MySQL is compliant with the ANSI SQL standard
 - MySQL was first released in 1995
 - MySQL is developed, distributed, and supported by Oracle Corporation
 - MySQL is named after co-founder Monty Widenius's daughter: My
-

Who Uses MySQL?

- Huge websites like Facebook, Twitter, Airbnb, Booking.com, Uber, GitHub, YouTube, etc.
 - Content Management Systems like WordPress, Drupal, Joomla!, Contao, etc.
 - A very large number of web developers around the world
-

Show Data On Your Web Site

To build a web site that shows data from a database, you will need:

- An RDBMS database program (like MySQL)
- A server-side scripting language, like PHP
- To use SQL to get the data you want
- To use HTML / CSS to style the page

CHAPTER - 4

4.1 MODULES:

Admin

Login

Pledge book maintainance

View all customer details.

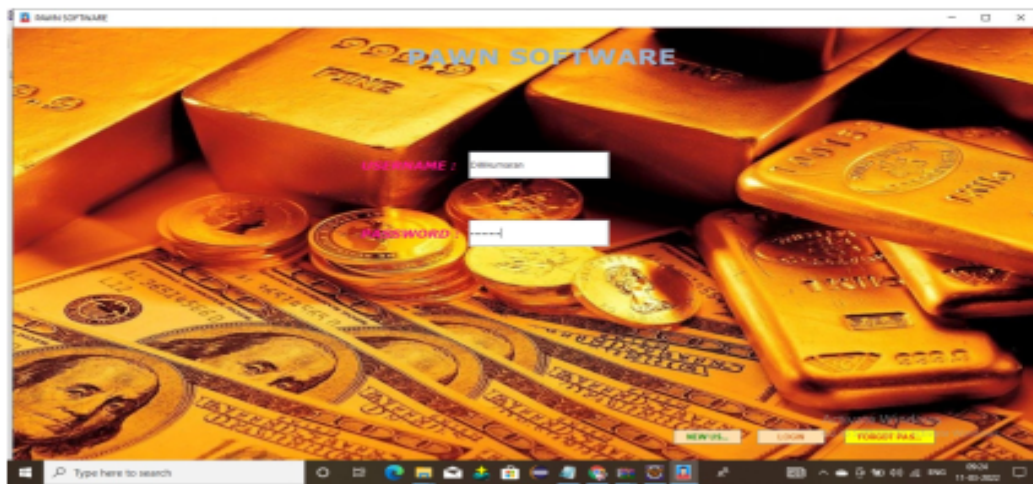
Day view (net or gross total of the days

transaction)

*CustomerRegister Login Create CustomerManager Customer DetailsManage
pledging DetailsSend SMS Alert Upcoming auction details Get sms alert*

5 APPLICATION PERFORMANCE:

Application Snapshot

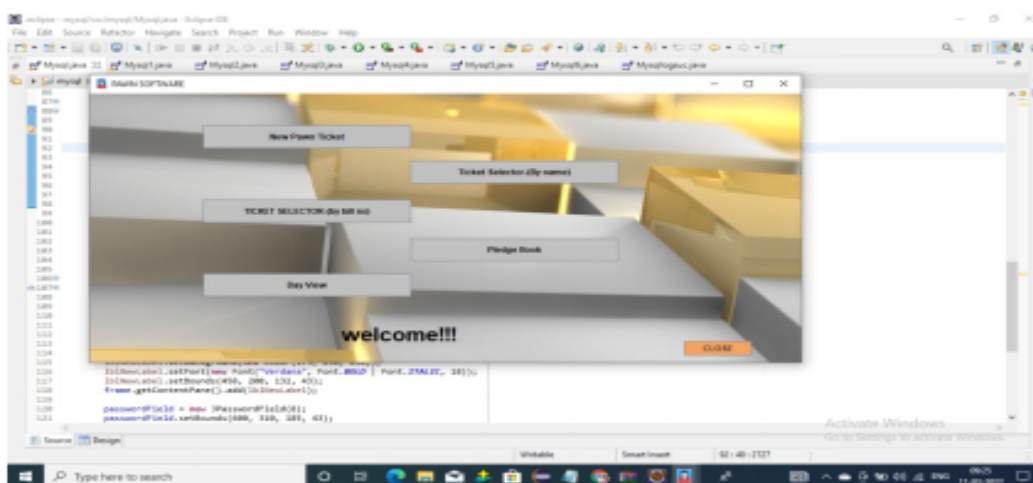


9 May 2022

Department of CSE

1

Application Snapshot

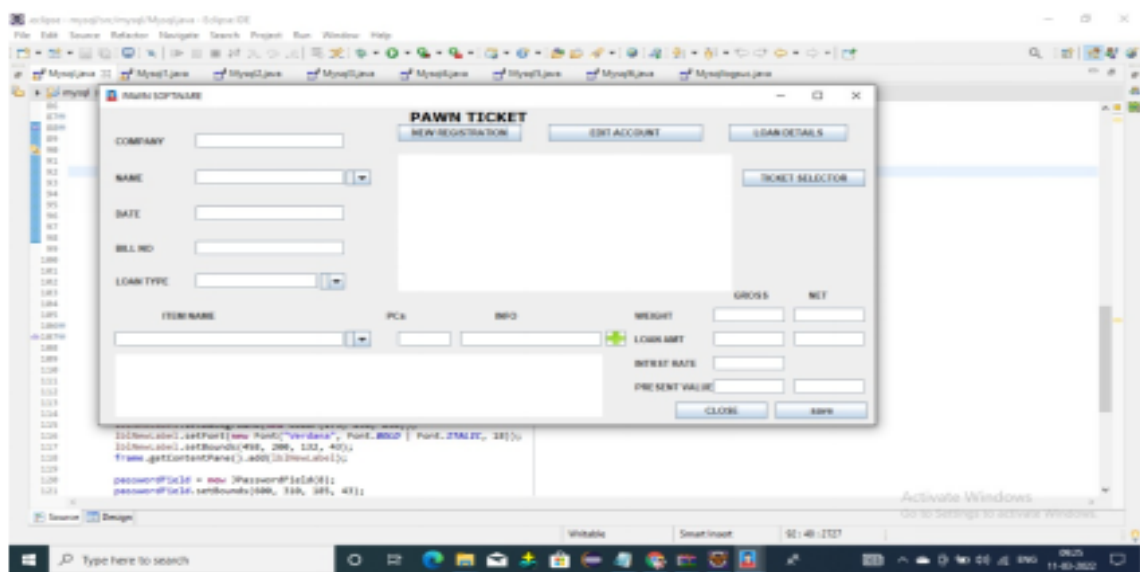


9 May 2022

Department of CSE

1

Application Snapshot

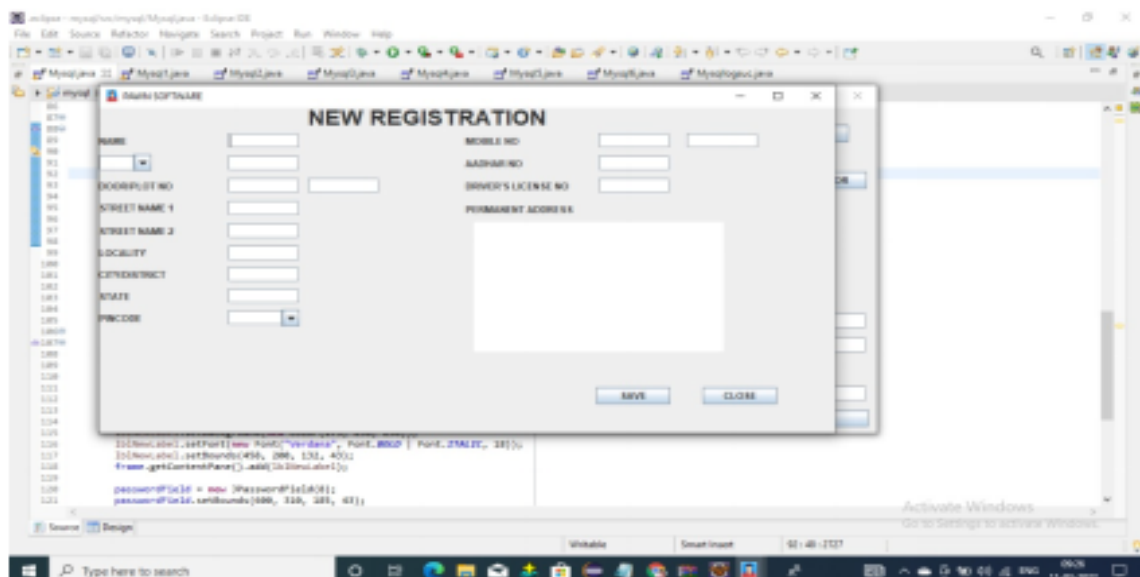


9 May 2022

Department of CSE

1

Application Snapshot

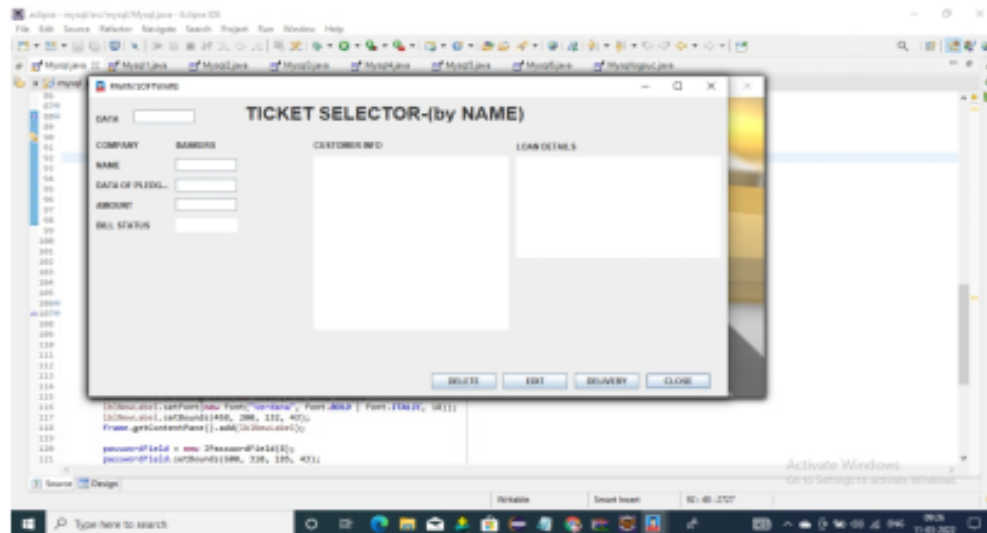


9 May 2022

Department of CSE

1

Application Snapshot

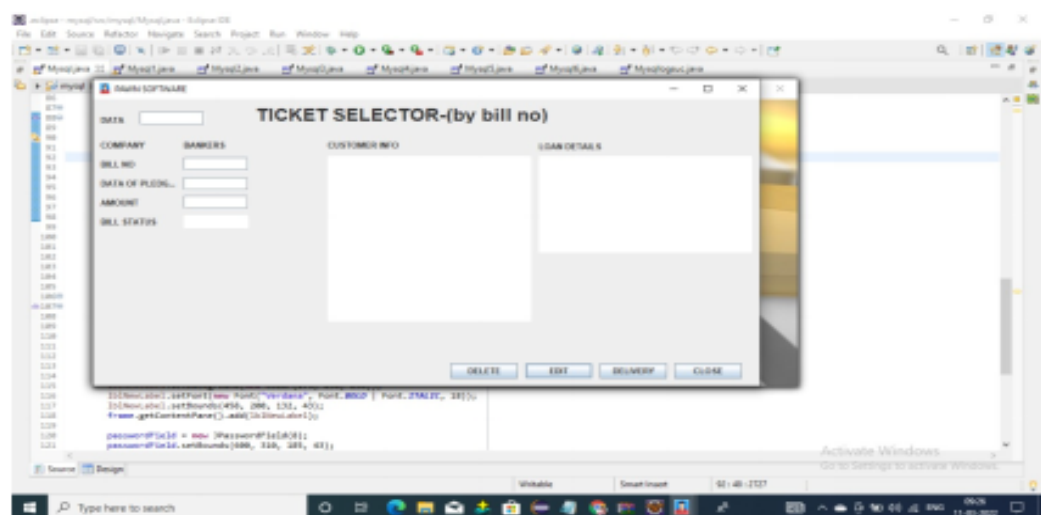


9 May 2022

Department of CSE

1

Application Snapshot



9 May 2022

Department of CSE

1

CHAPTER - 6

6.1 CONCLUSION:

A computerized pawn management system has been developed and the system was tested with sample data. The system results in regular timely preparations of required outputs.

In comparison with manual system the benefits under a computer system are considerable in the saving of man power working hours and Effort. Provision for addition , updation and deletion of customers is there in the system .

It is observed that proper filing system has been adopted for future refernce . The entire project runs on windows environments. The system can be used to make better management described at appropriate time. The user gets amount and timely information system.

6.2 SCOPE OF DEVELOPEMENT:

The system may be further updated or modified that will owing to its simple structure.

We can further add a transaction entity which will look after the payments made by the customer towards their policy.

Depending on future requirements more changes can be made owing to the organization's need.

6.3 REFERENCES:

WEBSITES

<http://www.stackoverflow.com>

<http://www.w3schools.com>

<http://www.tutorialspoint.com>

[en.wikipedia.org/wiki/ PHP](http://en.wikipedia.org/wiki/PHP)

5.www.hotscripts.com/category/php/6.www.apache.org/

7.www.mysql.com/click.php

CHAPTER - 7

7.1 APPENDIX:

A.) SOURCE CODE:

LANDING PAGE (LOGIN)

```
package mysql;

import java.awt.EventQueue;
import java.awt.Font;

import javax.swing.JFrame;
import javax.swing.JTextField;
import java.awt.BorderLayout;
import java.awt.Color;
import java.awt.Dimension;

import javax.swing.ImageIcon;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.JTextPane;
import javax.swing.JLabel;
import java.sql.DriverManager;
import javax.swing.JPasswordField;
import java.beans.PropertyChangeListener;
import java.sql.Connection;
import java.sql.SQLException;
import java.sql.Statement;
import java.beans.PropertyChangeEvent;
import java.awt.Toolkit;
import java.awt.SystemColor;
import javax.swing.SwingConstants;
```

```

public class Mysql {

    JFrame frame;
    private JTextField textField;
    private JPasswordField passwordField;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Mysql window = new Mysql();
                    window.frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the application.
     */
    public Mysql() {

```

```

        initialize();
    }

    /**
     * Initialize the contents of the frame.
     */
    private void initialize() {
        frame = new JFrame("PAWN SOFTWARE");

        frame.setIconImage(Toolkit.getDefaultToolkit().getImage("C:\\Users\\DIL
        LI KUMARAN\\Pictures\\Saved Pictures\\icon.png"));

        Dimension screenSize =
        Toolkit.getDefaultToolkit().getScreenSize();
        frame.setSize(screenSize.width, screenSize.height);

        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.getContentPane().setLayout(null);

        textField = new JTextField(15);
        textField.setBounds(600, 200, 185, 43);
        frame.getContentPane().add(textField);
        textField.setColumns(10);
        frame.setFont(new Font("verdana", Font.BOLD,9));
        frame.setForeground(Color.black);

        JButton btnNewButton = new JButton("NEW USER");
        btnNewButton.setForeground(new Color(0, 128, 0));
        btnNewButton.setBackground(new Color(255, 222, 173));
        btnNewButton.addActionListener(new ActionListener()

```

```

    {
        public void actionPerformed(ActionEvent e) {
String userValue = textField.getText();
            @SuppressWarnings("deprecation")
            String passValue = passwordField.getText();
        try {
            Connection connection =
DriverManager.getConnection("jdbc:mysql://localhost:3306/log_cred",
"root","Dilli21s@");

            String query = "INSERT INTO loginform values('" +
userValue + "','" + passValue + "')";

            Statement sta = (Statement) connection.createStatement();
            int x = ((java.sql.Statement) sta).executeUpdate(query);
            connection.close();
        }

        catch (Exception exception) {
            exception.printStackTrace();
        }

    }

});
btnNewButton.setBounds(870, 648, 89, 23);
frame.getContentPane().add(btnNewButton);

```

```

JButton btnNewButton_1 = new JButton("LOGIN");
btnNewButton_1.setForeground(new Color(210, 105, 30));
btnNewButton_1.setBackground(new Color(255, 222, 173));
btnNewButton_1.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        String userValue = textField.getText();
        String passValue = passwordField.getText();
        if (userValue.equals("Dillikumaran")&&
passValue.equals("Dilli20"))
        {
            Mysql1 page = new Mysql1();
            page.frame.setVisible(true);
        }
        else {

Mysqllogsuc page = new
Mysqllogsuc();

            page.frame.setVisible(true);
        }
    }
});
btnNewButton_1.setBounds(980, 648, 89, 23);
frame.getContentPane().add(btnNewButton_1);

JButton btnNewButton_2 = new JButton("FORGOT
PASSWORD");
btnNewButton_2.setForeground(Color.red);
btnNewButton_2.setBackground(Color.yellow);

```

```

btnNewButton_2.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
    }
});
btnNewButton_2.setBounds(1095, 648, 119, 23);
frame.getContentPane().add(btnNewButton_2);

JLabel lblNewLabel = new JLabel("USERNAME :");
lblNewLabel.setForeground(new Color(255, 20, 147));
lblNewLabel.setBackground(new Color(175, 238, 238));
lblNewLabel.setFont(new Font("Verdana", Font.BOLD |
Font.ITALIC, 18));
lblNewLabel.setBounds(458, 200, 132, 43);
frame.getContentPane().add(lblNewLabel);

passwordField = new JPasswordField(8);
passwordField.setBounds(600, 310, 185, 43);
frame.getContentPane().add(passwordField);

JLabel lblNewLabel_1 = new JLabel("PASSWORD :");
lblNewLabel_1.setForeground(new Color(255, 20, 147));
lblNewLabel_1.setBackground(new Color(175, 238, 238));
lblNewLabel_1.setFont(new Font("Verdana", Font.BOLD |
Font.ITALIC, 18));
lblNewLabel_1.setBounds(458, 310, 132, 43);
frame.getContentPane().add(lblNewLabel_1);

JLabel lblNewLabel_2 = new JLabel("PAWN SOFTWARE");

```

```

lblNewLabel_2.setHorizontalAlignment(SwingConstants.CENTER);

lblNewLabel_2.setBackground(SystemColor.textHighlightText);
        lblNewLabel_2.setBounds(485, 22, 420, 48);
        lblNewLabel_2.setFont(new Font("Verdana", Font.BOLD, 35));

        lblNewLabel_2.setForeground(SystemColor.activeCaption);
        frame.getContentPane().add(lblNewLabel_2);

        JLabel lblNewLabel_3 = new JLabel("New label");
        lblNewLabel_3.setIcon(new ImageIcon("C:\\Users\\DILLI
KUMARAN\\Pictures\\Saved Pictures\\gold3.jpg"));
        lblNewLabel_3.setBounds(0, 0, 1350, 729);
        frame.getContentPane().add(lblNewLabel_3);
    }
}

```

HOME PAGE(MENU)

```
package mysql;
```

```
import java.awt.EventQueue;
```

```
import java.awt.Font;
```

```
import java.awt.Toolkit;
```

```
import javax.swing.JFrame;
```

```
import javax.swing.JToolBar;
```

```
import javax.swing.SwingConstants;
```

```
import java.awt.BorderLayout;
```

```

import java.awt.Color;
import java.awt.Dimension;

import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.JLabel;
import javax.swing.ImageIcon;

public class Mysql1 {

    public JFrame frame;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Mysql1 window = new Mysql1();
                    window.frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }
}

```

```

    }

    /**
     * Create the application.
     */
    public Mysql1() {
        initialize();
    }

    /**
     * Initialize the contents of the frame.
     */
    public void initialize() {
        frame = new JFrame("PAWN SOFTWARE");
        Dimension screenSize =
Toolkit.getDefaultToolkit().getScreenSize();
        frame.setSize(screenSize.width, screenSize.height);
        frame.setFont(new Font("verdana", Font.BOLD,9));
        frame.setForeground(Color.black);

frame.setIconImage(Toolkit.getDefaultToolkit().getImage("C:\\Users\\DIL
LI KUMARAN\\Pictures\\Saved Pictures\\icon.png"));
        frame.getContentPane().setLayout(null);

        JButton btnNewButton = new JButton("New Pawn Ticket");
        btnNewButton.setForeground(Color.black);
        btnNewButton.setBackground(Color.lightGray);
        btnNewButton.setBounds(327, 54, 273, 37);

```

```

btnNewButton.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e) {
        Mysql2 page = new Mysql2();
        page.frame.setVisible(true);
    }
});
frame.getContentPane().add(btnNewButton);

JButton btnNewButton_1 = new JButton("Ticket Selector-(By
name)");
btnNewButton_1.setForeground(Color.black);
btnNewButton_1.setBackground(Color.lightGray);
btnNewButton_1.setBounds(648, 133, 273, 37);
btnNewButton_1.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e) {
        Mysql5 page = new Mysql5();
        page.frame.setVisible(true);
    }
});
frame.getContentPane().add(btnNewButton_1);

JButton btnNewButton_2 = new JButton("TICKET
SELECTOR-(by bill no)");
btnNewButton_2.setForeground(Color.black);
btnNewButton_2.setBackground(Color.lightGray);
btnNewButton_2.setBounds(327, 250, 273, 37);

```

```

btnNewButton_2.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e) {
        Mysql4 page = new Mysql4();
        page.frame.setVisible(true);
    }
});

```

```

frame.getContentPane().add(btnNewButton_2);

```

```

JButton btnNewButton_3 = new JButton("Pledge Book");
btnNewButton_3.setBounds(648, 346, 273, 37);
btnNewButton_3.setForeground(Color.black);
btnNewButton_3.setBackground(Color.lightGray);
btnNewButton_3.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e) {
    }
});

```

```

frame.getContentPane().add(btnNewButton_3);

```

```

JButton btnNewButton_4 = new JButton("Day View");
btnNewButton_4.setForeground(Color.black);
btnNewButton_4.setBackground(Color.lightGray);
btnNewButton_4.setBounds(327, 440, 273, 37);
btnNewButton_4.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)

```

```

        {
            Mysqldayv page = new Mysqldayv();
            page.frame.setVisible(true);
        }
    });
    frame.getContentPane().add(btnNewButton_4);

    JLabel lblNewLabel_1 = new JLabel(" welcome!!!");

    lblNewLabel_1.setHorizontalAlignment(SwingConstants.CENTER);
    lblNewLabel_1.setFont(new Font("ostwald", Font.BOLD,
50));

    lblNewLabel_1.setForeground(Color.black);
    lblNewLabel_1.setBounds(425,533, 500, 100);
    frame.getContentPane().add(lblNewLabel_1);

    JButton btnNewButton_5 = new JButton("CLOSE");
    btnNewButton_5.setBackground(new Color(244, 164, 96));
    btnNewButton_5.setBounds(1181, 651, 89, 23);
    btnNewButton_5.addActionListener(new ActionListener()
    {
        public void actionPerformed(ActionEvent e) {
            Mysql page = new Mysql();
            page.frame.setVisible(true);
        }
    });
    frame.getContentPane().add(btnNewButton_5);

```

```

        JLabel lblNewLabel = new JLabel("New label");
        lblNewLabel.setIcon(new ImageIcon("C:\\Users\\DILLI
KUMARAN\\Pictures\\Saved Pictures\\gold2.jpeg"));
        lblNewLabel.setBounds(0, 0, 1350, 729);
        frame.getContentPane().add(lblNewLabel);

    }
}

```

NEW BILL(EXISTING CUSTOMER)

```
package mysql;
```

```

import java.awt.Color;
import java.awt.Dimension;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.GradientPaint;
import java.awt.Graphics;
import java.awt.Graphics2D;
import java.awt.Panel;
import java.awt.RenderingHints;
import java.awt.Toolkit;

```

```

import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JComboBox;
import javax.swing.JTextField;

```

```
import javax.swing.SwingUtilities;
import javax.swing.JTextArea;
import javax.swing.DropMode;
import javax.swing.Icon;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.ImageIcon;
import javax.swing.SwingConstants;
import javax.swing.JTable;
```

```
public class Mysql2 {
```

```
    public JFrame frame;
    public JTextField textField;
    public JTextField textField_1;
    public JTextField textField_2;
    public JTextField textField_3;
    public JTextField textField_4;
    public JTextField textField_5;
    public JTextField textField_6;
    public JTextField textField_7;
    public JTextField textField_8;
    public JTextField textField_9;
    public JTextField textField_10;
    public JTextField textField_11;
    public JTextField textField_12;
```

```

public JTextField textField_13;
public JTextField textField_14;
public JTable table;

/**
 * Launch the application.
 */
public static void main(String[] args) {
    EventQueue.invokeLater(new Runnable() {
        public void run() {
            try {
                Mysql2 window = new Mysql2();
                window.frame.setVisible(true);
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    });
}

/**
 * Create the application.
 */
public Mysql2() {
    initialize();
}

```

```

/**
 * Initialize the contents of the frame.
 */

    public void initialize() {
        frame = new JFrame("PAWN SOFTWARE");
        frame.setSize(950,500);
        frame.setFont(new Font("verdana", Font.BOLD,9));
        frame.setForeground(Color.black);

frame.setIconImage(Toolkit.getDefaultToolkit().getImage("C:\\Users\\DIL
LI KUMARAN\\Pictures\\Saved Pictures\\icon.png"));

        frame.getContentPane().setLayout(null);

        JLabel lblNewLabel = new JLabel("COMPANY");
        lblNewLabel.setBounds(20, 39, 87, 14);
        frame.getContentPane().add(lblNewLabel);

        JLabel lblNewLabel_1 = new JLabel("NAME");
        lblNewLabel_1.setBounds(20, 89, 87, 14);
        frame.getContentPane().add(lblNewLabel_1);

        JComboBox comboBox = new JComboBox();
        comboBox.setBounds(297, 86, 30, 20);
        frame.getContentPane().add(comboBox);

        JLabel lblNewLabel_2 = new JLabel("PAWN TICKET");
        lblNewLabel_2.setFont(new Font("verdana",
Font.BOLD,18));

```

```
lblNewLabel_2.setForeground(Color.black);  
lblNewLabel_2.setBounds(374, 0, 185, 25);  
frame.getContentPane().add(lblNewLabel_2);
```

```
textField = new JTextField();  
textField.setForeground(Color.DARK_GRAY);  
textField.setBounds(117, 86, 179, 20);  
frame.getContentPane().add(textField);  
textField.setColumns(10);
```

```
textField_1 = new JTextField();  
textField_1.setBounds(117, 36, 179, 20);  
frame.getContentPane().add(textField_1);  
textField_1.setColumns(10);
```

```
JLabel lblNewLabel_3 = new JLabel("DATE");  
lblNewLabel_3.setBounds(20, 138, 87, 14);  
frame.getContentPane().add(lblNewLabel_3);
```

```
textField_2 = new JTextField();  
textField_2.setBounds(117, 135, 179, 20);  
frame.getContentPane().add(textField_2);  
textField_2.setColumns(10);
```

```
JLabel lblNewLabel_4 = new JLabel("BILL NO");  
lblNewLabel_4.setBounds(20, 185, 87, 14);  
frame.getContentPane().add(lblNewLabel_4);
```

```
textField_3 = new JTextField();  
textField_3.setBounds(117, 182, 179, 20);  
frame.getContentPane().add(textField_3);  
textField_3.setColumns(10);
```

```
JLabel lblNewLabel_5 = new JLabel("LOAN TYPE");  
lblNewLabel_5.setBounds(20, 230, 87, 14);  
frame.getContentPane().add(lblNewLabel_5);
```

```
textField_4 = new JTextField();  
textField_4.setBounds(117, 227, 147, 20);  
frame.getContentPane().add(textField_4);  
textField_4.setColumns(10);
```

```
JComboBox comboBox_1 = new JComboBox();  
comboBox_1.setBounds(266, 226, 30, 22);  
frame.getContentPane().add(comboBox_1);
```

```
JTextArea textArea = new JTextArea();  
textArea.setBounds(360, 64, 402, 187);  
frame.getContentPane().add(textArea);
```

```
JLabel lblNewLabel_6 = new JLabel("ITEM NAME");  
lblNewLabel_6.setBounds(77, 277, 110, 14);  
frame.getContentPane().add(lblNewLabel_6);
```

```
textField_5 = new JTextField();  
textField_5.setBounds(20, 306, 276, 20);  
frame.getContentPane().add(textField_5);  
textField_5.setColumns(10);
```

```
JLabel lblNewLabel_7 = new JLabel("PCs");  
lblNewLabel_7.setBounds(346, 277, 75, 14);  
frame.getContentPane().add(lblNewLabel_7);
```

```
textField_6 = new JTextField();  
textField_6.setBounds(360, 306, 65, 20);  
frame.getContentPane().add(textField_6);  
textField_6.setColumns(10);
```

```
JComboBox comboBox_2 = new JComboBox();  
comboBox_2.setBounds(297, 305, 30, 22);  
frame.getContentPane().add(comboBox_2);
```

```
JLabel lblNewLabel_8 = new JLabel("INFO");  
lblNewLabel_8.setBounds(477, 277, 75, 14);  
frame.getContentPane().add(lblNewLabel_8);
```

```
textField_7 = new JTextField();  
textField_7.setBounds(435, 306, 171, 20);  
frame.getContentPane().add(textField_7);  
textField_7.setColumns(10);
```

```
JTextArea textArea_1 = new JTextArea();
textArea_1.setBounds(20, 337, 586, 85);
frame.getContentPane().add(textArea_1);
```

```
Icon icon = new
ImageIcon("C:\\\\Users\\\\DINESH\\\\Downloads\\\\images.jpg");

JButton btnNewButton = new JButton(new
ImageIcon("C:\\Users\\DINESH\\Pictures\\images2.jpg"));
btnNewButton.setBounds(610, 305, 24, 20);
btnNewButton.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e) {
    }
});
frame.getContentPane().add(btnNewButton);
```

```
JLabel lblNewLabel_9 = new JLabel("WEIGHT");
lblNewLabel_9.setBounds(645, 277, 96, 14);
frame.getContentPane().add(lblNewLabel_9);
```

```
JLabel lblNewLabel_9_1 = new JLabel("LOAN AMT");
lblNewLabel_9_1.setBounds(644, 309, 95, 14);
frame.getContentPane().add(lblNewLabel_9_1);
```

```
JLabel lblNewLabel_9_2 = new JLabel("INTRST RATE");
lblNewLabel_9_2.setBounds(645, 342, 96, 14);
frame.getContentPane().add(lblNewLabel_9_2);
```

```
JLabel lblNewLabel_9_3 = new JLabel("PRESENT VALUE");  
lblNewLabel_9_3.setBounds(645, 374, 96, 14);  
frame.getContentPane().add(lblNewLabel_9_3);
```

```
textField_8 = new JTextField();  
textField_8.setBounds(740, 274, 86, 20);  
frame.getContentPane().add(textField_8);  
textField_8.setColumns(10);
```

```
textField_9 = new JTextField();  
textField_9.setColumns(10);  
textField_9.setBounds(740, 306, 86, 20);  
frame.getContentPane().add(textField_9);
```

```
textField_10 = new JTextField();  
textField_10.setColumns(10);  
textField_10.setBounds(740, 339, 86, 20);  
frame.getContentPane().add(textField_10);
```

```
textField_11 = new JTextField();  
textField_11.setColumns(10);  
textField_11.setBounds(740, 371, 86, 20);  
frame.getContentPane().add(textField_11);
```

```
textField_12 = new JTextField();  
textField_12.setColumns(10);  
textField_12.setBounds(836, 274, 86, 20);
```

```
frame.getContentPane().add(textField_12);
```

```
textField_13 = new JTextField();
```

```
textField_13.setColumns(10);
```

```
textField_13.setBounds(836, 306, 86, 20);
```

```
frame.getContentPane().add(textField_13);
```

```
textField_14 = new JTextField();
```

```
textField_14.setColumns(10);
```

```
textField_14.setBounds(836, 371, 86, 20);
```

```
frame.getContentPane().add(textField_14);
```

```
        JButton btnNewButton_1 = new JButton("TICKET  
SELECTOR");
```

```
        btnNewButton_1.setBounds(775, 85, 147, 23);
```

```
        btnNewButton_1.addActionListener(new ActionListener()
```

```
{
```

```
            public void actionPerformed(ActionEvent e) {
```

```
            }
```

```
});
```

```
frame.getContentPane().add(btnNewButton_1);
```

```
JButton btnNewButton_2 = new JButton("save");
```

```
btnNewButton_2.setBounds(814, 402, 110, 23);
```

```
btnNewButton_2.addActionListener(new ActionListener()
```

```
{
```

```
            public void actionPerformed(ActionEvent e) {
```

```
            }
```

```

});
frame.getContentPane().add(btnNewButton_2);

JLabel lblNewLabel_10 = new JLabel("GROSS");
lblNewLabel_10.setBounds(764, 249, 46, 14);
frame.getContentPane().add(lblNewLabel_10);

JLabel lblNewLabel_10_1 = new JLabel("NET");
lblNewLabel_10_1.setBounds(854, 249, 46, 14);
frame.getContentPane().add(lblNewLabel_10_1);

JButton btnNewButton_3 = new JButton("NEW
REGISTRATION");
btnNewButton_3.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e)
{
        Mysql3 page = new Mysql3();
        page.frame.setVisible(true);
    }
});
btnNewButton_3.setBounds(360, 23, 149, 23);
btnNewButton_3.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e) {
        Mysql3 page = new Mysql3();
        page.frame.setVisible(true);
    }
});

```

```
frame.getContentPane().add(btnNewButton_3);
```

```
JButton btnNewButton_4 = new JButton("EDIT ACCOUNT");
```

```
btnNewButton_4.setBounds(541, 23, 185, 23);
```

```
btnNewButton_4.addActionListener(new ActionListener()
```

```
{
```

```
    public void actionPerformed(ActionEvent e) {
```

```
    }
```

```
});
```

```
frame.getContentPane().add(btnNewButton_4);
```

```
JButton btnNewButton_5 = new JButton("LOAN DETAILS");
```

```
btnNewButton_5.addActionListener(new ActionListener() {
```

```
    public void actionPerformed(ActionEvent e) {
```

```
    }
```

```
});
```

```
btnNewButton_5.setBounds(758, 23, 142, 23);
```

```
frame.getContentPane().add(btnNewButton_5);
```

```
table = new JTable();
```

```
table.setBounds(20, 420, 586, -82);
```

```
frame.getContentPane().add(table);
```

```
JButton btnNewButton_2_1 = new JButton("CLOSE");
```

```
btnNewButton_2_1.addActionListener(new ActionListener() {
```

```
    public void actionPerformed(ActionEvent e) {
```

```
        Mysql1 page = new Mysql1();
```

```

        page.frame.setVisible(true);
    }
});
btnNewButton_2_1.setBounds(708, 402, 96, 23);
frame.getContentPane().add(btnNewButton_2_1);

JButton btnNewButton_2_1_1 = new JButton("PRINT");
btnNewButton_2_1_1.addActionListener(new
ActionListener() {
    public void actionPerformed(ActionEvent e) {

    }

});
btnNewButton_2_1_1.setBounds(616, 399, 75, 23);
frame.getContentPane().add(btnNewButton_2_1_1);
}
}

```

B.)DB CONNECTION:

```
package mysql;
```

```

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.Statement;

```

```

public class DB
{
    public static Connection getConnection()
    {
        Connection con=null;
        try
        {
            Class.forName("com.mysql.jdbc.Driver");

            con=DriverManager.getConnection("jdbc:mysql://localhost:3306/log_cre
            d", "root","Dilli21s@");

            Statement st=con.createStatement();
            String s1="create table loginform(Username
            varchar(20),Password varchar(20)";
            st.executeUpdate(s1);
        }
        catch (Exception e)
        {
            e.printStackTrace();
        }
        return con;
    }
}

```

THANK YOU

