PRACTICAL 1

Aim: Develop the presentation layer of Library Management software application with suitable menus.

Library.java

```
import java.awt.BorderLayout;
import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.GroupLayout;
import javax.swing.GroupLayout.Alignment;
import javax.swing.JLabel;
import java.awt.Color;
import java.awt.Font;
import javax.swing.JButton;
import javax.swing.LayoutStyle.ComponentPlacement;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;

public class Library extends JFrame {
	static Library frame;
	private JPanel contentPane;

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

	/**
	 * Create the frame.
	 */
	public Library() {
		setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		setBounds(100, 100, 450, 300);
		contentPane = new JPanel();
		contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
		setContentPane(contentPane);
```

```
        JLabel lblLibraryManagement = new JLabel("Library Management ");
        lblLibraryManagement.setFont(new Font("Tahoma", Font.PLAIN, 18));
        lblLibraryManagement.setForeground(Color.GRAY);

        JButton btnAdminLogin = new JButton("Admin Login");
        btnAdminLogin.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
            AdminLogin.main(new String[]{});
            frame.dispose();
            }
        });
        btnAdminLogin.setFont(new Font("Tahoma", Font.PLAIN, 15));

        JButton btnLibrarianLogin = new JButton("Librarian Login");
        btnLibrarianLogin.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                LibrarianLogin.main(new String[]{});
            }
        });
        btnLibrarianLogin.setFont(new Font("Tahoma", Font.PLAIN, 15));
        GroupLayout gl_contentPane = new GroupLayout(contentPane);
        gl_contentPane.setHorizontalGroup(
            gl_contentPane.createParallelGroup(Alignment.LEADING)
                .addGroup(gl_contentPane.createSequentialGroup()

    .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)

    .addGroup(gl_contentPane.createSequentialGroup()
                            .addGap(64)

    .addComponent(lblLibraryManagement))

    .addGroup(gl_contentPane.createSequentialGroup()
                            .addGap(140)

    .addGroup(gl_contentPane.createParallelGroup(Alignment.TRAILING, false)

    .addComponent(btnLibrarianLogin, Alignment.LEADING,
GroupLayout.DEFAULT_SIZE, GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                                .addComponent(btnAdminLogin,
Alignment.LEADING, GroupLayout.DEFAULT_SIZE, 135, Short.MAX_VALUE))))
                    .addContainerGap(95, Short.MAX_VALUE))
        );
        gl_contentPane.setVerticalGroup(
            gl_contentPane.createParallelGroup(Alignment.LEADING)
                .addGroup(gl_contentPane.createSequentialGroup()
                    .addContainerGap()
                    .addComponent(lblLibraryManagement)
                    .addGap(32)
                    .addComponent(btnAdminLogin,
GroupLayout.PREFERRED_SIZE, 52, GroupLayout.PREFERRED_SIZE)

    .addPreferredGap(ComponentPlacement.UNRELATED)
                    .addComponent(btnLibrarianLogin,
GroupLayout.PREFERRED_SIZE, 53, GroupLayout.PREFERRED_SIZE)
                    .addContainerGap(70, Short.MAX_VALUE))
        );
        contentPane.setLayout(gl_contentPane);
    }
}
```

AdminLogin.java

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

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.GroupLayout;
import javax.swing.GroupLayout.Alignment;
import javax.swing.JLabel;
import javax.swing.JOptionPane;

import java.awt.Font;
import java.awt.Color;
import javax.swing.JTextField;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.JPasswordField;

public class AdminLogin extends JFrame {
    static AdminLogin frame;
    private JPanel contentPane;
    private JTextField textField;
    private JPasswordField passwordField;

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

    /**
     * Create the frame.
     */
    public AdminLogin() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 450, 300);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);

        JLabel lblAdminLoginForm = new JLabel("Admin Login Form");
        lblAdminLoginForm.setForeground(Color.GRAY);
        lblAdminLoginForm.setFont(new Font("Tahoma", Font.PLAIN, 18));

        JLabel lblEnterName = new JLabel("Enter Name:");

        JLabel lblEnterPassword = new JLabel("Enter Password:");

        textField = new JTextField();
        textField.setColumns(10);

        JButton btnLogin = new JButton("Login");
        btnLogin.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
            String name=textField.getText();
            String password=String.valueOf(passwordField.getPassword());
            if(name.equals("admin")&&password.equals("admin123")){
                AdminSuccess.main(new String[]{});
                frame.dispose();
            }else{
                JOptionPane.showMessageDialog(AdminLogin.this, "Sorry,
Username or Password Error","Login Error!", JOptionPane.ERROR_MESSAGE);
                textField.setText("");
                passwordField.setText("");
            }
            }
        });

        passwordField = new JPasswordField();
        GroupLayout gl_contentPane = new GroupLayout(contentPane);
        gl_contentPane.setHorizontalGroup(
            gl_contentPane.createParallelGroup(Alignment.TRAILING)
                .addGroup(gl_contentPane.createSequentialGroup()

    .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)
    .addGroup(gl_contentPane.createSequentialGroup()
                            .addGap(124)
                            .addComponent(lblAdminLoginForm))

    .addGroup(gl_contentPane.createSequentialGroup()
                            .addGap(19)

    .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)
                                .addComponent(lblEnterName)

    .addComponent(lblEnterPassword))
                            .addGap(47)

    .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING, false)
                                .addComponent(passwordField)
                                .addComponent(textField,
GroupLayout.DEFAULT_SIZE, 172, Short.MAX_VALUE))))
                    .addContainerGap(107, Short.MAX_VALUE))
                .addGroup(gl_contentPane.createSequentialGroup()
                    .addContainerGap(187, Short.MAX_VALUE)
                    .addComponent(btnLogin,
GroupLayout.PREFERRED_SIZE, 86, GroupLayout.PREFERRED_SIZE)
                    .addGap(151))
        );
        gl_contentPane.setVerticalGroup(
            gl_contentPane.createParallelGroup(Alignment.LEADING)
                .addGroup(gl_contentPane.createSequentialGroup()
                    .addComponent(lblAdminLoginForm)
                    .addGap(26)

    .addGroup(gl_contentPane.createParallelGroup(Alignment.BASELINE)
                        .addComponent(lblEnterName)
                        .addComponent(textField,
GroupLayout.PREFERRED_SIZE, GroupLayout.DEFAULT_SIZE,
GroupLayout.PREFERRED_SIZE))
                    .addGap(28)

    .addGroup(gl_contentPane.createParallelGroup(Alignment.BASELINE)
                        .addComponent(lblEnterPassword)
                        .addComponent(passwordField,
GroupLayout.PREFERRED_SIZE, GroupLayout.DEFAULT_SIZE,
GroupLayout.PREFERRED_SIZE))
                    .addGap(18)
                    .addComponent(btnLogin,
GroupLayout.PREFERRED_SIZE, 37, GroupLayout.PREFERRED_SIZE)
                    .addContainerGap(80, Short.MAX_VALUE))
        );
        contentPane.setLayout(gl_contentPane);
    }
```

```
}
```

<u>AdminSuccess.java</u>

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

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.GroupLayout;
import javax.swing.GroupLayout.Alignment;
import javax.swing.JLabel;
import java.awt.Color;
import java.awt.Font;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.LayoutStyle.ComponentPlacement;
import java.sql.*;
public class AdminSuccess extends JFrame {
    static AdminSuccess frame;
    private JPanel contentPane;

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

    /**
     * Create the frame.
     */
    public AdminSuccess() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 450, 371);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);

        JLabel lblAdminSection = new JLabel("Admin Section");
```

```
		lblAdminSection.setFont(new Font("Tahoma", Font.PLAIN, 22));
		lblAdminSection.setForeground(Color.GRAY);

		JButton btnNewButton = new JButton("Add Librarian");
		btnNewButton.setFont(new Font("Tahoma", Font.PLAIN, 15));
		btnNewButton.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent e) {
			LibrarianForm.main(new String[]{});
			frame.dispose();
			}
		});

		JButton btnViewLibrarian = new JButton("View Librarian");
		btnViewLibrarian.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent arg0) {
			ViewLibrarian.main(new String[]{});
			}
		});
		btnViewLibrarian.setFont(new Font("Tahoma", Font.PLAIN, 15));

		JButton btnDeleteLibrarian = new JButton("Delete Librarian");
		btnDeleteLibrarian.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent e) {
			DeleteLibrarian.main(new String[]{});
			frame.dispose();
			}
		});
		btnDeleteLibrarian.setFont(new Font("Tahoma", Font.PLAIN, 15));

		JButton btnLogout = new JButton("Logout");
		btnLogout.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent arg0) {
				Library.main(new String[]{});
				frame.dispose();
			}
		});
		btnLogout.setFont(new Font("Tahoma", Font.PLAIN, 15));
		GroupLayout gl_contentPane = new GroupLayout(contentPane);
		gl_contentPane.setHorizontalGroup(
			gl_contentPane.createParallelGroup(Alignment.TRAILING)
				.addGroup(gl_contentPane.createSequentialGroup()
					.addContainerGap(150, Short.MAX_VALUE)
					.addComponent(lblAdminSection,
GroupLayout.PREFERRED_SIZE, 151, GroupLayout.PREFERRED_SIZE)
					.addGap(123))
				.addGroup(Alignment.LEADING,
gl_contentPane.createSequentialGroup()
					.addGap(134)

	.addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)
                                                .addComponent(btnLogout,
GroupLayout.PREFERRED_SIZE, 181, GroupLayout.PREFERRED_SIZE)
                                                .addComponent(btnDeleteLibrarian,
GroupLayout.PREFERRED_SIZE, 181, GroupLayout.PREFERRED_SIZE)
                                                .addComponent(btnViewLibrarian,
GroupLayout.PREFERRED_SIZE, 181, GroupLayout.PREFERRED_SIZE)
                                                .addComponent(btnNewButton,
GroupLayout.PREFERRED_SIZE, 181, GroupLayout.PREFERRED_SIZE))
                                        .addContainerGap(109, Short.MAX_VALUE))
                );
                gl_contentPane.setVerticalGroup(
                        gl_contentPane.createParallelGroup(Alignment.LEADING)
                                .addGroup(gl_contentPane.createSequentialGroup()
                                        .addComponent(lblAdminSection,
GroupLayout.PREFERRED_SIZE, 40, GroupLayout.PREFERRED_SIZE)
                                        .addGap(11)
                                        .addComponent(btnNewButton,
GroupLayout.PREFERRED_SIZE, 49, GroupLayout.PREFERRED_SIZE)
                                        .addGap(18)
                                        .addComponent(btnViewLibrarian,
GroupLayout.PREFERRED_SIZE, 49, GroupLayout.PREFERRED_SIZE)
                                        .addGap(18)
                                        .addComponent(btnDeleteLibrarian,
GroupLayout.PREFERRED_SIZE, 49, GroupLayout.PREFERRED_SIZE)
                                        .addGap(18)
                                        .addComponent(btnLogout,
GroupLayout.PREFERRED_SIZE, 49, GroupLayout.PREFERRED_SIZE)
                                        .addContainerGap(21, Short.MAX_VALUE))
                );
                contentPane.setLayout(gl_contentPane);
        }
}
```

LibrarianLogin.java

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

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.GroupLayout;
import javax.swing.GroupLayout.Alignment;
import javax.swing.JLabel;
import javax.swing.JOptionPane;

import java.awt.Font;
import java.awt.Color;
import javax.swing.JTextField;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.JPasswordField;

public class LibrarianLogin extends JFrame {
	static LibrarianLogin frame;
	private JPanel contentPane;
	private JTextField textField;
	private JPasswordField passwordField;

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

	/**
	 * Create the frame.
	 */
	public LibrarianLogin() {
		setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		setBounds(100, 100, 450, 300);
		contentPane = new JPanel();
		contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
		setContentPane(contentPane);

		JLabel lblAdminLoginForm = new JLabel("Librarian Login Form");
		lblAdminLoginForm.setForeground(Color.GRAY);
		lblAdminLoginForm.setFont(new Font("Tahoma", Font.PLAIN, 18));

		JLabel lblEnterName = new JLabel("Enter Name:");

		JLabel lblEnterPassword = new JLabel("Enter Password:");

		textField = new JTextField();
		textField.setColumns(10);

		JButton btnLogin = new JButton("Login");
		btnLogin.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent e) {
			String name=textField.getText();
```

```
            String password=String.valueOf(passwordField.getPassword());
            if(name.equals("raana")&&password.equals("raana123")){
                LibrarianSuccess.main(new String[]{});
                frame.dispose();
            }else{
                JOptionPane.showMessageDialog(LibrarianLogin.this, "Sorry,
Username or Password Error","Login Error!", JOptionPane.ERROR_MESSAGE);
                textField.setText("");
                passwordField.setText("");
            }
            }
        });

        passwordField = new JPasswordField();
        GroupLayout gl_contentPane = new GroupLayout(contentPane);
        gl_contentPane.setHorizontalGroup(
            gl_contentPane.createParallelGroup(Alignment.TRAILING)
                .addGroup(gl_contentPane.createSequentialGroup()

    .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)

    .addGroup(gl_contentPane.createSequentialGroup()
                            .addGap(124)
                            .addComponent(lblAdminLoginForm))

    .addGroup(gl_contentPane.createSequentialGroup()
                            .addGap(19)

    .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)
                                .addComponent(lblEnterName)

    .addComponent(lblEnterPassword))
                            .addGap(47)

    .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING, false)
                                .addComponent(passwordField)
                                .addComponent(textField,
GroupLayout.DEFAULT_SIZE, 172, Short.MAX_VALUE))))
                    .addContainerGap(107, Short.MAX_VALUE))
                .addGroup(gl_contentPane.createSequentialGroup()
                    .addContainerGap(187, Short.MAX_VALUE)
                    .addComponent(btnLogin,
GroupLayout.PREFERRED_SIZE, 86, GroupLayout.PREFERRED_SIZE)
                    .addGap(151))
        );
        gl_contentPane.setVerticalGroup(
            gl_contentPane.createParallelGroup(Alignment.LEADING)
                .addGroup(gl_contentPane.createSequentialGroup()
                    .addComponent(lblAdminLoginForm)
                    .addGap(26)
    .addGroup(gl_contentPane.createParallelGroup(Alignment.BASELINE)
                        .addComponent(lblEnterName)
                        .addComponent(textField,
GroupLayout.PREFERRED_SIZE, GroupLayout.DEFAULT_SIZE,
GroupLayout.PREFERRED_SIZE))
                    .addGap(28)

    .addGroup(gl_contentPane.createParallelGroup(Alignment.BASELINE)
                        .addComponent(lblEnterPassword)
                        .addComponent(passwordField,
GroupLayout.PREFERRED_SIZE, GroupLayout.DEFAULT_SIZE,
GroupLayout.PREFERRED_SIZE))
                    .addGap(18)
                    .addComponent(btnLogin,
GroupLayout.PREFERRED_SIZE, 37, GroupLayout.PREFERRED_SIZE)
                    .addContainerGap(80, Short.MAX_VALUE))
        );
        contentPane.setLayout(gl_contentPane);
    }
}
```

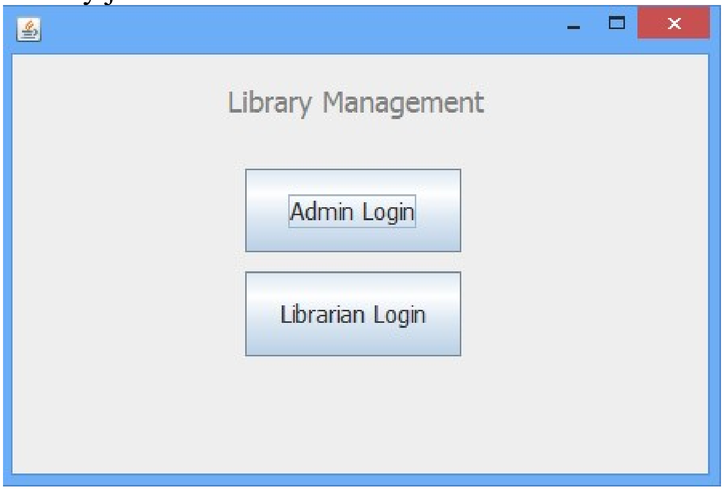
LibrarySuccess.java

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

import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.GroupLayout;
import javax.swing.GroupLayout.Alignment;
import javax.swing.JLabel;
import java.awt.Font;
import java.awt.Color;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;

public class LibrarianSuccess extends JFrame {
    static LibrarianSuccess frame;
    private JPanel contentPane;

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

	/**
	 * Create the frame.
	 */
	public LibrarianSuccess() {
		setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		setBounds(100, 100, 450, 433);
		contentPane = new JPanel();
		contentPane.setForeground(Color.GRAY);
		contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
		setContentPane(contentPane);

		JLabel lblLibrarianSection = new JLabel("Librarian Section - JavaTpoint");
		lblLibrarianSection.setFont(new Font("Tahoma", Font.PLAIN, 22));

		JButton btnNewButton = new JButton("Add Books");
		btnNewButton.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent e) {
			BooksForm.main(new String[]{});
			frame.dispose();
			}
		});
		btnNewButton.setFont(new Font("Tahoma", Font.PLAIN, 13));

		JButton btnViewBooks = new JButton("View Books");
		btnViewBooks.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent arg0) {
				ViewBooks.main(new String[]{});
			}
		});
		btnViewBooks.setFont(new Font("Tahoma", Font.PLAIN, 13));

		JButton btnIssueBook = new JButton("Issue Book");
		btnIssueBook.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent e) {
				IssueBookForm.main(new String[]{});
				frame.dispose();
			}
		});
		btnIssueBook.setFont(new Font("Tahoma", Font.PLAIN, 13));

		JButton btnViewIssuedBooks = new JButton("View Issued Books");
		btnViewIssuedBooks.addActionListener(new ActionListener() {
```

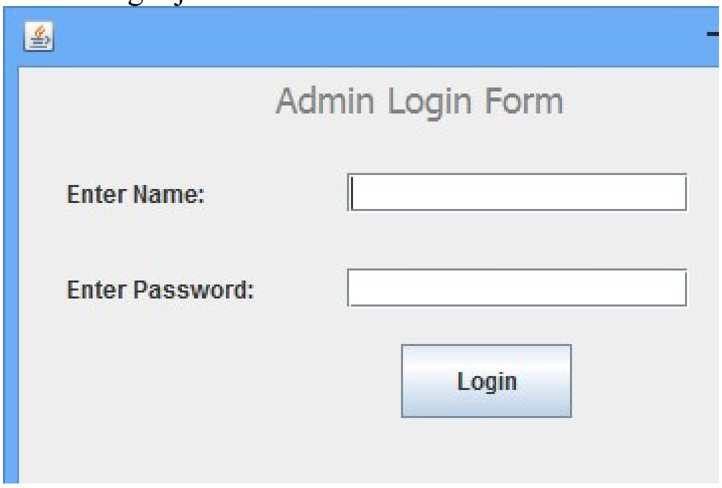
```
            public void actionPerformed(ActionEvent e) {
                ViewIssuedBooks.main(new String[]{});
            }
        });
        btnViewIssuedBooks.setFont(new Font("Tahoma", Font.PLAIN, 13));

        JButton btnReturnBook = new JButton("Return Book");
        btnReturnBook.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                ReturnBook.main(new String[]{});
                frame.dispose();
            }
        });
        btnReturnBook.setFont(new Font("Tahoma", Font.PLAIN, 13));

        JButton btnLogout = new JButton("Logout");
        btnLogout.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                Library.main(new String[]{});
                frame.dispose();
            }
        });
        btnLogout.setFont(new Font("Tahoma", Font.PLAIN, 13));
        GroupLayout gl_contentPane = new GroupLayout(contentPane);
        gl_contentPane.setHorizontalGroup(
            gl_contentPane.createParallelGroup(Alignment.LEADING)
                .addGroup(Alignment.TRAILING,
gl_contentPane.createSequentialGroup()
                    .addContainerGap(81, Short.MAX_VALUE)
                    .addComponent(lblLibrarianSection)
                    .addGap(54))
                .addGroup(gl_contentPane.createSequentialGroup()
                    .addGap(132)

    .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)
                        .addComponent(btnLogout,
GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE)
                        .addComponent(btnReturnBook,
GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE)
                        .addComponent(btnViewIssuedBooks,
GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE)
                        .addComponent(btnIssueBook,
GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE)
                        .addComponent(btnViewBooks,
GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE)
                        .addComponent(btnNewButton,
GroupLayout.PREFERRED_SIZE, 191, GroupLayout.PREFERRED_SIZE))
                    .addContainerGap(101, Short.MAX_VALUE))
        );
        gl_contentPane.setVerticalGroup(
            gl_contentPane.createParallelGroup(Alignment.LEADING)
                .addGroup(gl_contentPane.createSequentialGroup()
                    .addContainerGap()
                    .addComponent(lblLibrarianSection)
                    .addGap(18)
                    .addComponent(btnNewButton,
GroupLayout.PREFERRED_SIZE, 37, GroupLayout.PREFERRED_SIZE)
                    .addGap(18)
                    .addComponent(btnViewBooks,
GroupLayout.PREFERRED_SIZE, 37, GroupLayout.PREFERRED_SIZE)
                    .addGap(18)
                    .addComponent(btnIssueBook,
GroupLayout.PREFERRED_SIZE, 37, GroupLayout.PREFERRED_SIZE)
                    .addGap(18)
                    .addComponent(btnViewIssuedBooks,
GroupLayout.PREFERRED_SIZE, 37, GroupLayout.PREFERRED_SIZE)
                    .addGap(18)
                    .addComponent(btnReturnBook,
GroupLayout.PREFERRED_SIZE, 37, GroupLayout.PREFERRED_SIZE)
                    .addGap(18)
                    .addComponent(btnLogout,
GroupLayout.PREFERRED_SIZE, 37, GroupLayout.PREFERRED_SIZE)
                    .addContainerGap(16, Short.MAX_VALUE))
        );
        contentPane.setLayout(gl_contentPane);
    }

}
```

Outputs

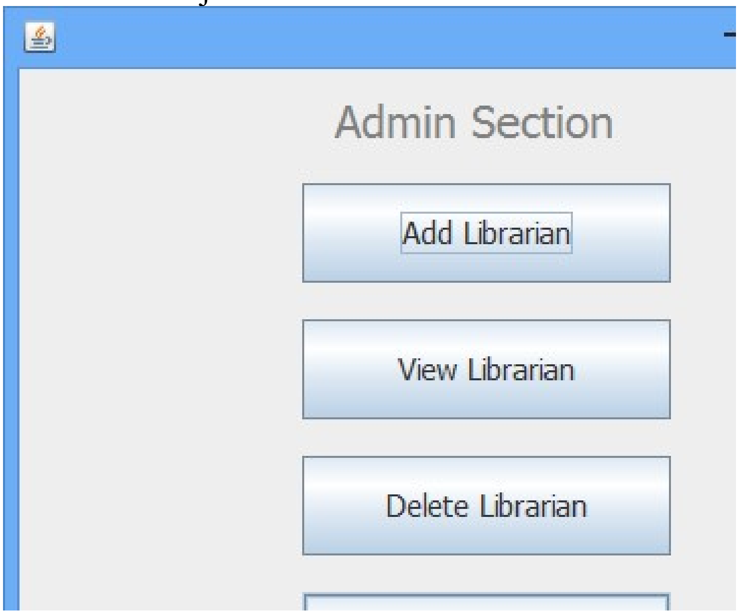
Library.java



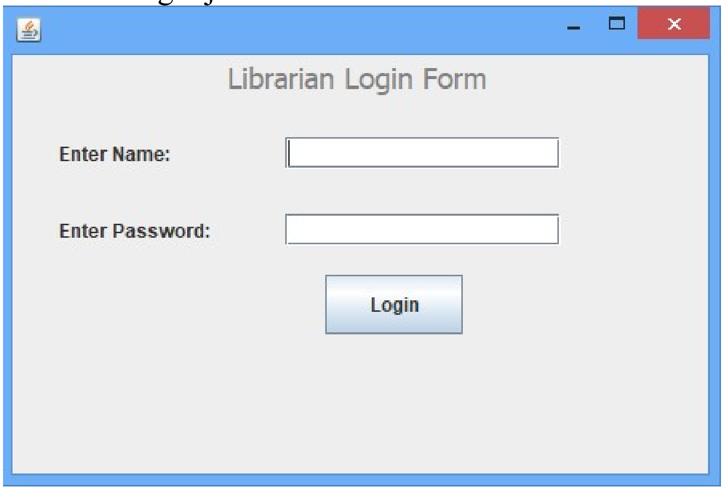
AdminLogin.java



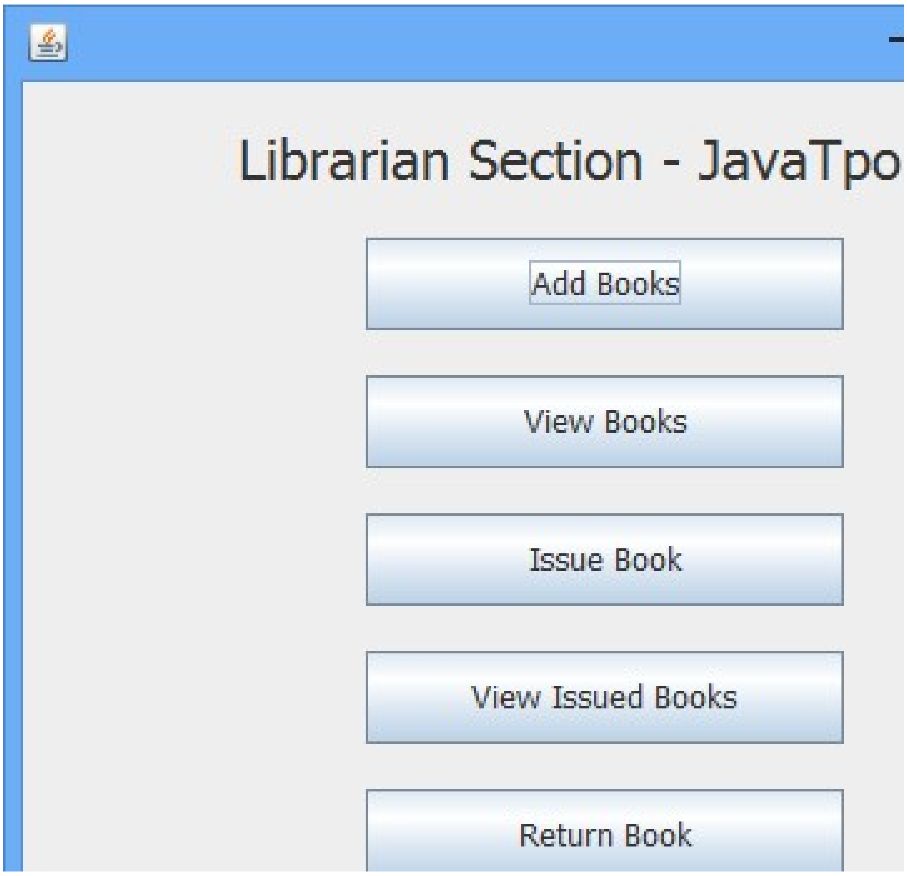
AdminSuccess.java



LibrarianLogin.java



LibrarianSuccess.java



Librarian Section - JavaTpo

Practical 4

Aim: Develop Java application to store image in a database as well as retrieve image from database.

Table

```
Create table myimages
(
        ID number(2),
        Name varchar2(20),
        Description varchar2(50),
        Image longblob
);
```

Work1.java:- Store

```
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import javax.swing.*;
import javax.swing.filechooser.FileNameExtensionFilter;
import java.io.File;
import java.io.FileInputStream;
import java.io.InputStream;


public class Work1 extends JFrame{
   JButton button ;
   JButton button2;
   JLabel label;
   JTextField textID;
   JTextField textNAME;
   JTextArea area;
   String s;

   public Work1(){

   super("insert image to database in java");

   button = new JButton("ADD");
   button.setBounds(200,300,100,40);

   button2 = new JButton("Browse");
   button2.setBounds(80, 300, 100, 40);

   textID = new JTextField("ID");
   textID.setBounds(320,290,100,20);
```

```
    textNAME = new JTextField("Name");
    textNAME.setBounds(320,330,100,20);

    area = new JTextArea("DESCRIPTION",100, 100);

    JScrollPane pane = new JScrollPane(area);
    pane.setBounds(450, 270, 200, 100);

    label = new JLabel();
    label.setBounds(10,10,670,250);

    //button to browse the image into jlabel
    button2.addActionListener(new ActionListener(){
        @Override
     public void actionPerformed(ActionEvent e){
         JFileChooser fileChooser = new JFileChooser();
         fileChooser.setCurrentDirectory(new File(System.getProperty("user.home")));
         FileNameExtensionFilter filter = new FileNameExtensionFilter("*.IMAGE",
"jpg","gif","png");
         fileChooser.addChoosableFileFilter(filter);
         int result = fileChooser.showSaveDialog(null);
         if(result == JFileChooser.APPROVE_OPTION){
             File selectedFile = fileChooser.getSelectedFile();
             String path = selectedFile.getAbsolutePath();
             label.setIcon(ResizeImage(path));
             s = path;
              }
         else if(result == JFileChooser.CANCEL_OPTION){
             System.out.println("No Data");
         }
     }
    });

    //button to insert image and some data into mysql database
    button.addActionListener(new ActionListener(){

       @Override
       public void actionPerformed(ActionEvent e){
           try{
               Connection con =
DriverManager.getConnection("jdbc:mysql://localhost/db_images","root","");
               PreparedStatement ps = con.prepareStatement("insert into
myimages(ID,Name,Description,Image) values(?,?,?,?)");
               InputStream is = new FileInputStream(new File(s));
               ps.setString(1, textID.getText());
               ps.setString(2, textNAME.getText());
               ps.setString(3, area.getText());
               ps.setBlob(4,is);
               ps.executeUpdate();
```

```
            JOptionPane.showMessageDialog(null, "Data Inserted");
          }catch(Exception ex){
            ex.printStackTrace();
          }
        }
      });

      add(label);
      add(textID);
      add(textNAME);
      add(pane);
      add(button);
      add(button2);
      setLayout(null);
      setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
      setSize(700,420);
      setVisible(true);
      }

      public ImageIcon ResizeImage(String imgPath){
        ImageIcon MyImage = new ImageIcon(imgPath);
        Image img = MyImage.getImage();
        Image newImage = img.getScaledInstance(label.getWidth(),
label.getHeight(),Image.SCALE_SMOOTH);
        ImageIcon image = new ImageIcon(newImage);
        return image;
      }

      public static void main(String[] args){
        new Work1();
      }
     }
```

Work.java Retrieve image

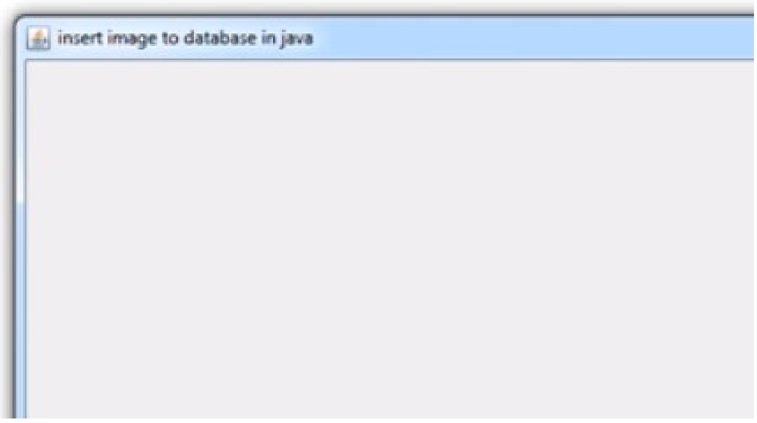
```
import java.awt.Color;
import java.awt.Image;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import javax.swing.*;

public class Work extends JFrame{
   JButton button ;
   JLabel label;
   JTextField jtf;

   public Work(){
   super("retrieve image from database in java");
```

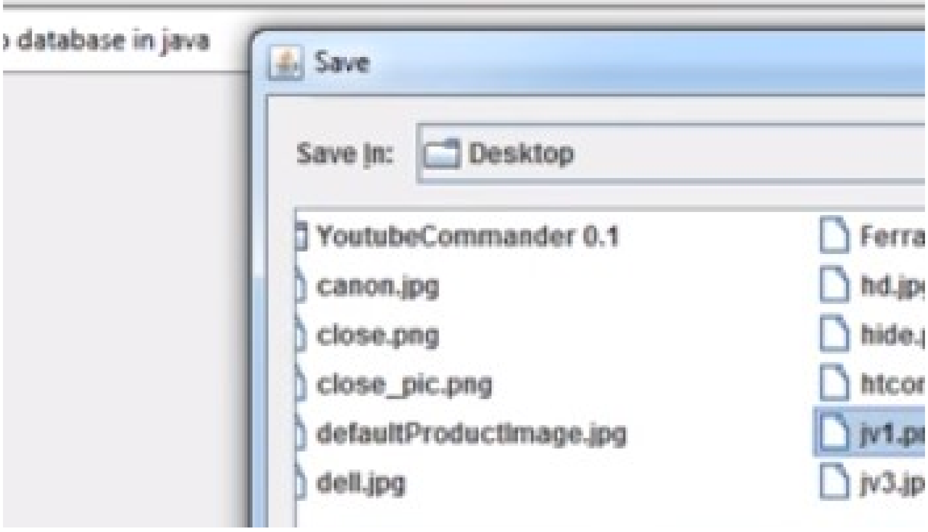
```
    button = new JButton("Retrieve");
    button.setBounds(250,300,100,40);

    jtf = new JTextField();
    jtf.setBounds(360,310,100,20);

    label = new JLabel();
    label.setBounds(10,10,670,250);

    add(button);
    add(label);
    add(jtf);

    button.addActionListener(new ActionListener() {

        @Override
        public void actionPerformed(ActionEvent e) {

            try{
                Connection con =
DriverManager.getConnection("jdbc:mysql://localhost/db_images","root","");
                Statement st = con.createStatement();
                ResultSet rs = st.executeQuery("select * from myimages where ID =
'"+jtf.getText()+"'");
                if(rs.next()){
                    byte[] img = rs.getBytes("Image");



                    //Resize The ImageIcon
                    ImageIcon image = new ImageIcon(img);
                    Image im = image.getImage();
                    Image myImg = im.getScaledInstance(label.getWidth(),
label.getHeight(),Image.SCALE_SMOOTH);
                    ImageIcon newImage = new ImageIcon(myImg);
                    label.setIcon(newImage);
                }

                else{
                    JOptionPane.showMessageDialog(null, "No Data");
                }
            }catch(Exception ex){
                ex.printStackTrace();
            }

        }
    });

    setLayout(null);
```

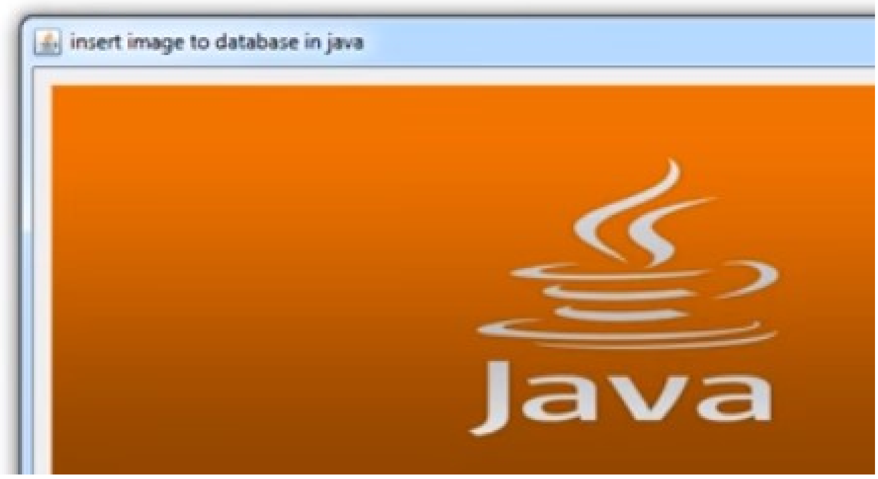
```
 setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
 getContentPane().setBackground(Color.decode("#bdb76b"));
 setLocationRelativeTo(null);
 setSize(700,400);
 setVisible(true);
 }

 public static void main(String[] args){
    new Work();
 }
}
```

Output

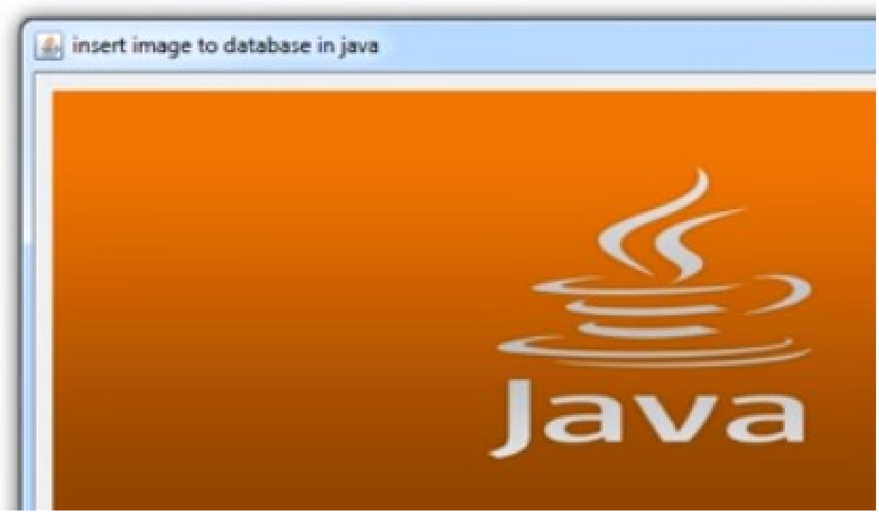
Insertion



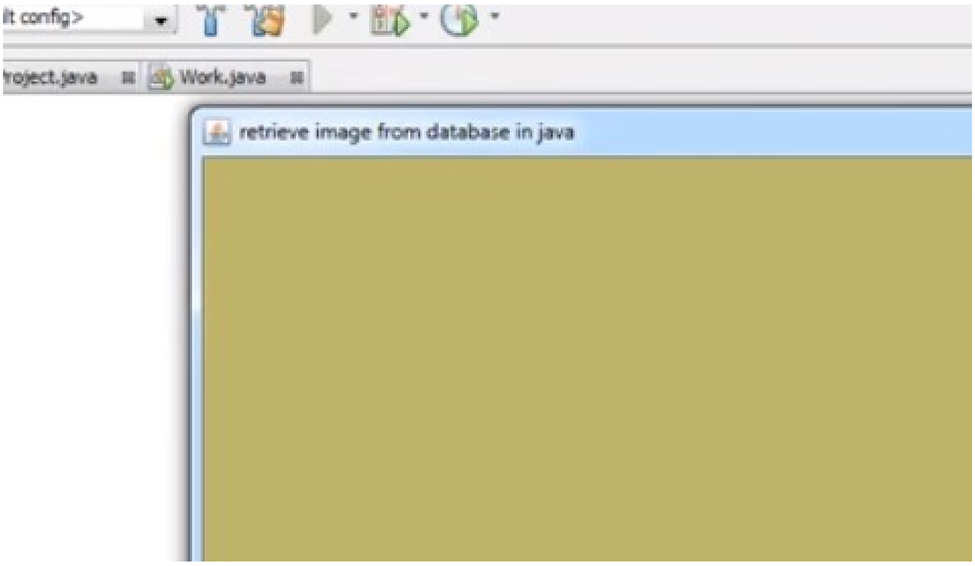


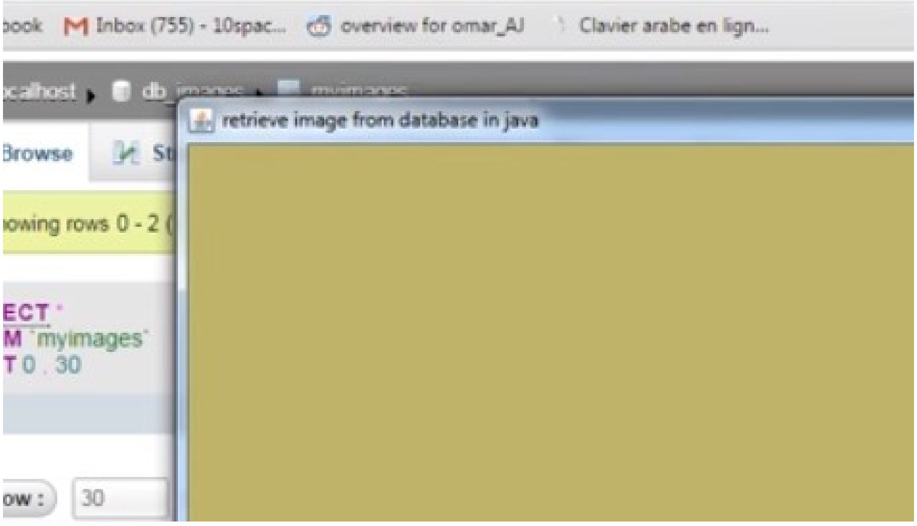

insert image to database in java
Java

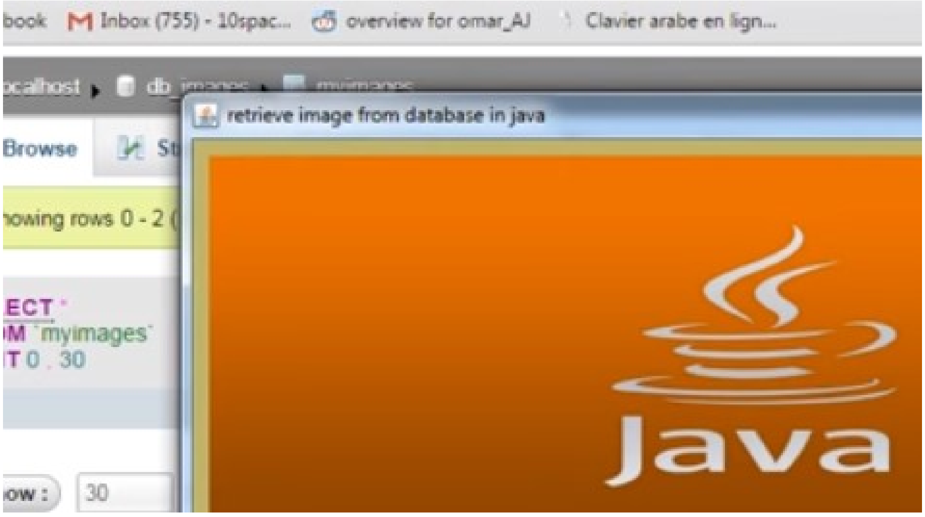
un)
Project.java
Work.java


insert image to database in java
Java

Retrieving


lt config>
roject.java
Work.java
retrieve image from database in java


book
Inbox (755) - 10spac...
overview for omar_AJ
Clavier arabe en lign...
ocalhost
db_images
myimages
retrieve image from database in java
Browse
St
owing rows 0 - 2 (
ECT *
M `myimages`
T 0 , 30
ow :
30


book
Inbox (755) - 10spac...
overview for omar_AJ
Clavier arabe en lign...
ocalhost
db_images
myimages
retrieve image from database in java
Browse
St
owing rows 0 - 2 (
ECT *
M `myimages`
T 0 , 30
ow :
30
Java

Practical 5

Aim: Write a Java application to demonstrate servlet life cycle.

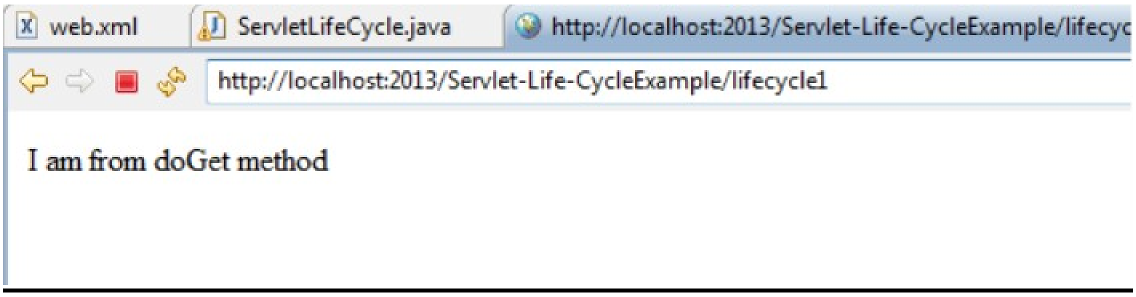
ServletLifeCycle.java

```
import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.ServletConfig;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class ServletLifeCycle extends HttpServlet
{

 public ServletLifeCycle()
 {
 System.out.println("Am from default constructor");
 }

 public void init(ServletConfig config)
 {
 System.out.println("Am from Init method...!");
 }

 public void doGet(HttpServletRequest req,HttpServletResponse res)throws
ServletException,IOException
 {
 res.setContentType("text/html");
 PrintWriter pw = res.getWriter();
 pw.println("I am from doGet method");
 pw.close();
 }

 public void destroy()
 {
 System.out.println("Am from Destroy methods");
 }

}
```

web.xml

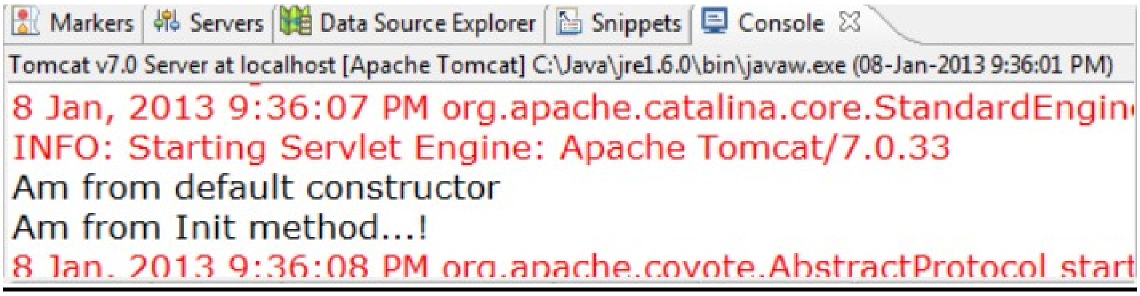
```
<web-app>
     <servlet>
          <servlet-name>second</servlet-name>
          <servlet-class>java4s.ServletLifeCycle</servlet-class>
        <load-on-startup>1</load-on-startup>
     </servlet>

     <servlet-mapping>
          <servlet-name>second</servlet-name>
        <url-pattern>/lifecycle1</url-pattern>
    </servlet-mapping>
</web-app>
```

Output



http://localhost:2013/Servlet-Life-CycleExample/lifecycle1

I am from doGet method



Tomcat v7.0 Server at localhost [Apache Tomcat] C:\Java\jre1.6.0\bin\javaw.exe (08-Jan-2013 9:36:01 PM)

```
8 Jan, 2013 9:36:07 PM org.apache.catalina.core.StandardEngin
INFO: Starting Servlet Engine: Apache Tomcat/7.0.33
Am from default constructor
Am from Init method...!
8 Jan, 2013 9:36:08 PM org.apache.coyote.AbstractProtocol start
```

Practical 6

Aim: Design database for employee administration. Develop servlet(s) to perform CRUD operations.

Index.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>

<h1>Add New Employee</h1>
<form action="SaveServlet" method="post">
<table>
<tr><td>Name:</td><td><input type="text" name="name"/></td></tr>
<tr><td>Password:</td><td><input type="password" name="password"/></td></tr>
<tr><td>Email:</td><td><input type="email" name="email"/></td></tr>
<tr><td>Country:</td><td>
<select name="country" style="width:150px">
<option>India</option>
<option>USA</option>
<option>UK</option>
<option>Other</option>
</select>
</td></tr>
<tr><td colspan="2"><input type="submit" value="Save Employee"/></td></tr>
</table>
</form>

<br/>
<a href="ViewServlet">view employees</a>

</body>
</html>
```

Emp.java

```
public class Emp {
private int id;
private String name,password,email,country;
public int getId() {
   return id;
}
public void setId(int id) {
   this.id = id;
}
public String getName() {
   return name;
}
public void setName(String name) {
    this.name = name;
}
public String getPassword() {
    return password;
}
public void setPassword(String password) {
    this.password = password;
}
public String getEmail() {
    return email;
}
public void setEmail(String email) {
    this.email = email;
}
public String getCountry() {
    return country;
}
public void setCountry(String country) {
    this.country = country;
}

}
```

EmpDao.java

```
import java.util.*;
import java.sql.*;

public class EmpDao {

    public static Connection getConnection(){
        Connection con=null;
        try{
            Class.forName("oracle.jdbc.driver.OracleDriver");
            con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","oracle");
        }catch(Exception e){System.out.println(e);}
        return con;
    }
    public static int save(Emp e){
        int status=0;
        try{
            Connection con=EmpDao.getConnection();
            PreparedStatement ps=con.prepareStatement(
                         "insert into Employee(name,password,email,country) values (?,?,?,?)");
            ps.setString(1,e.getName());
            ps.setString(2,e.getPassword());
            ps.setString(3,e.getEmail());
            ps.setString(4,e.getCountry());

            status=ps.executeUpdate();

            con.close();
        }catch(Exception ex){ex.printStackTrace();}

        return status;
    }
    public static int update(Emp e){
```

```java
        int status=0;
        try{
            Connection con=EmpDao.getConnection();
            PreparedStatement ps=con.prepareStatement(
                         "update Employee set name=?,password=?,email=?,country=? where id=?");
            ps.setString(1,e.getName());
            ps.setString(2,e.getPassword());
            ps.setString(3,e.getEmail());
            ps.setString(4,e.getCountry());
            ps.setInt(5,e.getId());

            status=ps.executeUpdate();

            con.close();
        }catch(Exception ex){ex.printStackTrace();}

        return status;
    }
    public static int delete(int id){
        int status=0;
        try{
            Connection con=EmpDao.getConnection();
            PreparedStatement ps=con.prepareStatement("delete from Employee where id=?");
            ps.setInt(1,id);
            status=ps.executeUpdate();

            con.close();
        }catch(Exception e){e.printStackTrace();}

        return status;
    }
    public static Emp getEmployeeById(int id){
        Emp e=new Emp();

        try{
            Connection con=EmpDao.getConnection();
            PreparedStatement ps=con.prepareStatement("select * from Employee where id=?");
            ps.setInt(1,id);
            ResultSet rs=ps.executeQuery();
            if(rs.next()){
                e.setId(rs.getInt(1));
                e.setName(rs.getString(2));
                e.setPassword(rs.getString(3));
                e.setEmail(rs.getString(4));
                e.setCountry(rs.getString(5));
            }
            con.close();
        }catch(Exception ex){ex.printStackTrace();}

        return e;
    }
    public static List<Emp> getAllEmployees(){
        List<Emp> list=new ArrayList<Emp>();

        try{
            Connection con=EmpDao.getConnection();
            PreparedStatement ps=con.prepareStatement("select * from Employee");
            ResultSet rs=ps.executeQuery();
            while(rs.next()){
                Emp e=new Emp();
                e.setId(rs.getInt(1));
                e.setName(rs.getString(2));
                e.setPassword(rs.getString(3));
                e.setEmail(rs.getString(4));
                e.setCountry(rs.getString(5));
                list.add(e);
            }
            con.close();
        }catch(Exception e){e.printStackTrace();}

        return list;
    }
}
```

SaveServelet.java

```
import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
@WebServlet("/SaveServlet")
public class SaveServlet extends HttpServlet {
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
         throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out=response.getWriter();

        String name=request.getParameter("name");
        String password=request.getParameter("password");
        String email=request.getParameter("email");
        String country=request.getParameter("country");

        Emp e=new Emp();
        e.setName(name);
        e.setPassword(password);
        e.setEmail(email);
        e.setCountry(country);

        int status=EmpDao.save(e);
        if(status>0){
            out.print("<p>Record saved successfully!</p>");
            request.getRequestDispatcher("index.html").include(request, response);
        }else{
            out.println("Sorry! unable to save record");
        }
        out.close();
    }
}
```

EditServlet.java

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
@WebServlet("/EditServlet")
public class EditServlet extends HttpServlet {
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
           throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out=response.getWriter();
        out.println("<h1>Update Employee</h1>");
        String sid=request.getParameter("id");
        int id=Integer.parseInt(sid);

        Emp e=EmpDao.getEmployeeById(id);

        out.print("<form action='EditServlet2' method='post'>");
        out.print("<table>");
        out.print("<tr><td></td><td><input type='hidden' name='id' value='"+e.getId()+"'/></td></tr>");

        out.print("<tr><td>Name:</td><td><input type='text' name='name' value='"+e.getName()+"'/></
td></tr>");
        out.print("<tr><td>Password:</td><td><input type='password' name='password' value='"+e.getP
assword()+"'/>
                </td></tr>");
        out.print("<tr><td>Email:</td><td><input type='email' name='email' value='"+e.getEmail()+"'/>
</td></tr>");
        out.print("<tr><td>Country:</td><td>");
        out.print("<select name='country' style='width:150px'>");
        out.print("<option>India</option>");
        out.print("<option>USA</option>");
        out.print("<option>UK</option>");
        out.print("<option>Other</option>");
        out.print("</select>");
        out.print("</td></tr>");
        out.print("<tr><td colspan='2'><input type='submit' value='Edit & Save '/></td></tr>");
        out.print("</table>");
        out.print("</form>");

        out.close();
    }
}
```

EditServlet2.java

```
import java.io.IOException;
import java.io.PrintWriter;

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

```
@WebServlet("/EditServlet2")
public class EditServlet2 extends HttpServlet {
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
          throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out=response.getWriter();

        String sid=request.getParameter("id");
        int id=Integer.parseInt(sid);
        String name=request.getParameter("name");
        String password=request.getParameter("password");
        String email=request.getParameter("email");
        String country=request.getParameter("country");

        Emp e=new Emp();
        e.setId(id);
        e.setName(name);
        e.setPassword(password);
        e.setEmail(email);
        e.setCountry(country);

        int status=EmpDao.update(e);
        if(status>0){
            response.sendRedirect("ViewServlet");
        }else{
            out.println("Sorry! unable to update record");
        }

        out.close();
    }

}
```

DeleteServlet.java

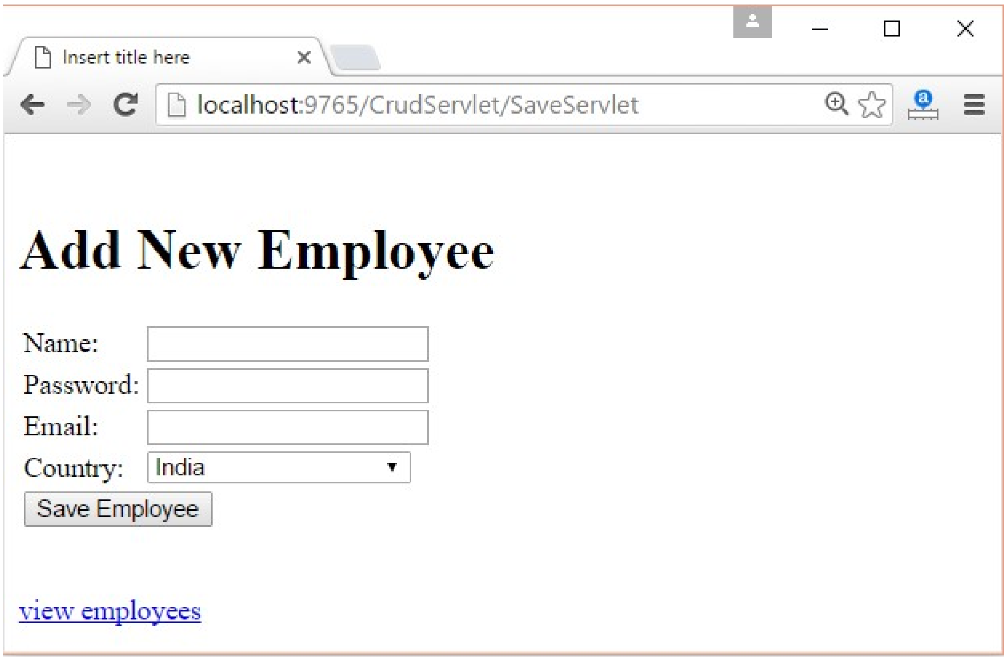
```
import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
@WebServlet("/DeleteServlet")
public class DeleteServlet extends HttpServlet {
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
             throws ServletException, IOException {
        String sid=request.getParameter("id");
        int id=Integer.parseInt(sid);
        EmpDao.delete(id);
        response.sendRedirect("ViewServlet");
    }
}
```

ViewServlet.java

```
import java.io.IOException;
import java.io.PrintWriter;
import java.util.List;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
@WebServlet("/ViewServlet")
public class ViewServlet extends HttpServlet {
   protected void doGet(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
       response.setContentType("text/html");
       PrintWriter out=response.getWriter();
       out.println("<a href='index.html'>Add New Employee</a>");
       out.println("<h1>Employees List</h1>");

       List<Emp> list=EmpDao.getAllEmployees();

       out.print("<table border='1' width='100%'");
       out.print("<tr><th>Id</th><th>Name</th><th>Password</th><th>Email</th><th>Country</th
>
            <th>Edit</th><th>Delete</th></tr>");
       for(Emp e:list){
        out.print("<tr><td>"+e.getId()+"</td><td>"+e.getName()+"</td><td>"+e.getPassword()+"</td
>
            <td>"+e.getEmail()+"</td><td>"+e.getCountry()+"</td><td><a href='EditServlet?id="+e.
getId()+"'>edit</a></td>
            <td><a href='DeleteServlet?id="+e.getId()+"'>delete</a></td></tr>");
       }
       out.print("</table>");

       out.close();
   }
}
```

Output

First page will look like this, fill the form and submit it.



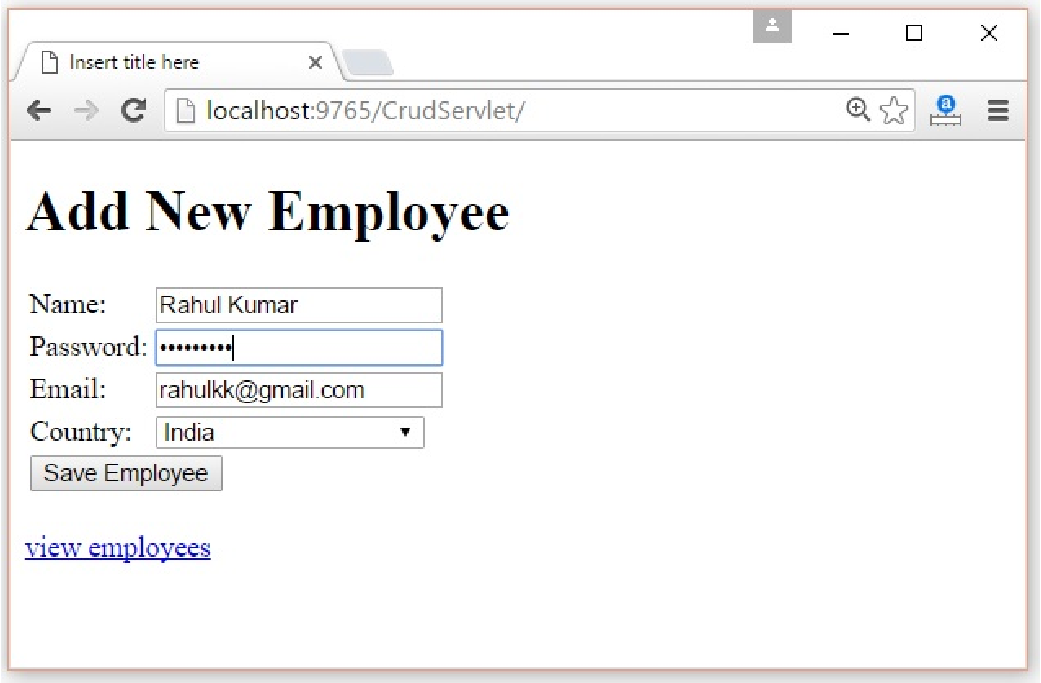
Add New Employee

Name:

Password:

Email:

Country: India



Add New Employee

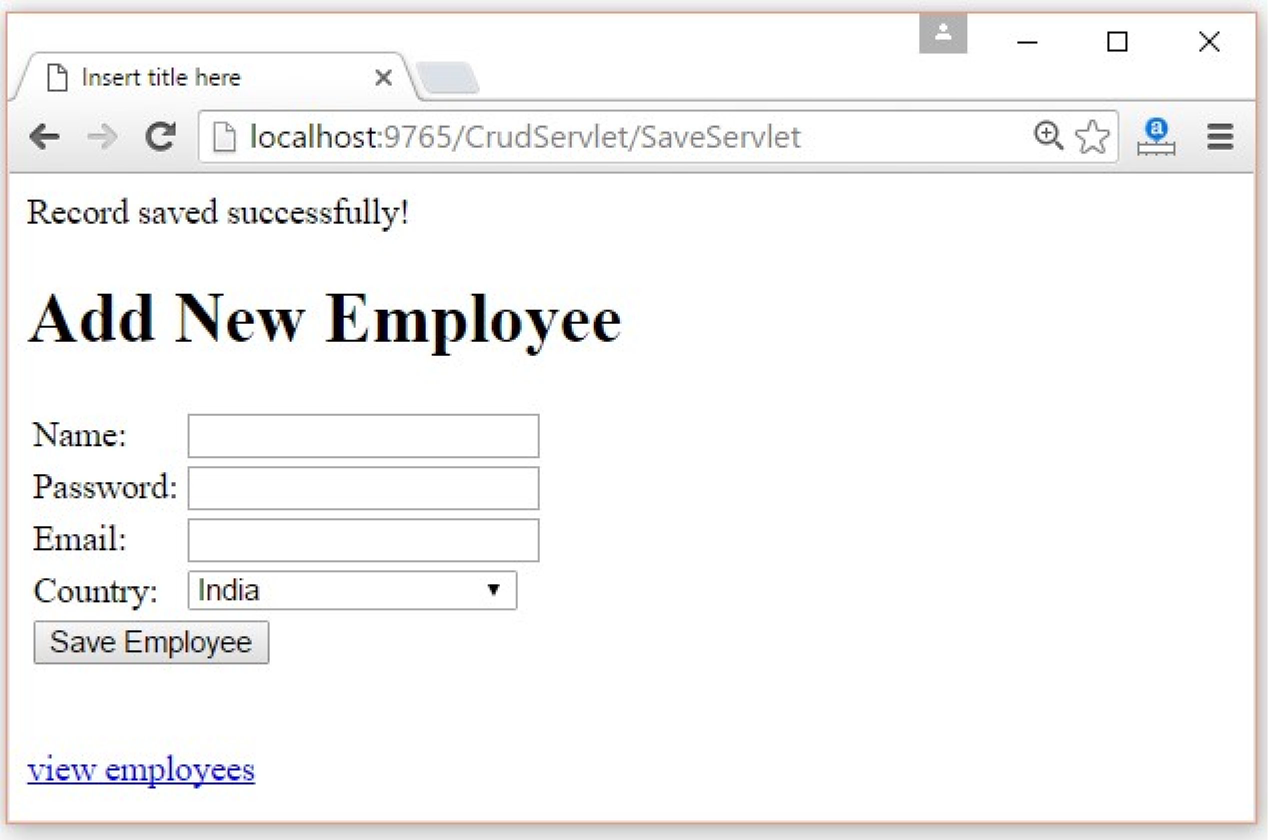
Name: Rahul Kumar

Password: •••••••••

Email: rahulkk@gmail.com

Country: India

You will get a message "Record successfully saved!".



Record saved successfully!

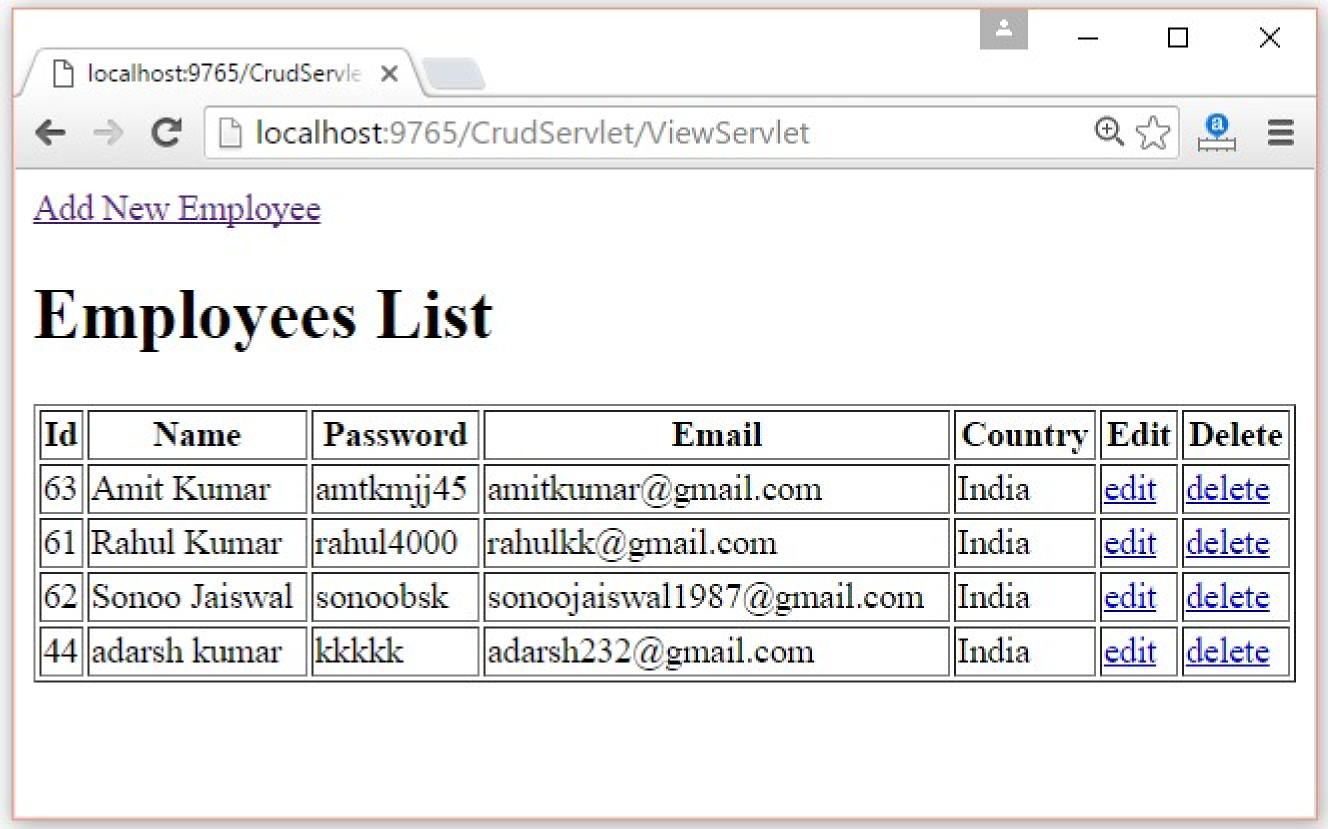
Add New Employee

Name:	
Password:	
Email:	
Country:	India ▾

Save Employee

view employees

Click on the View Employees link to see the total employees list.

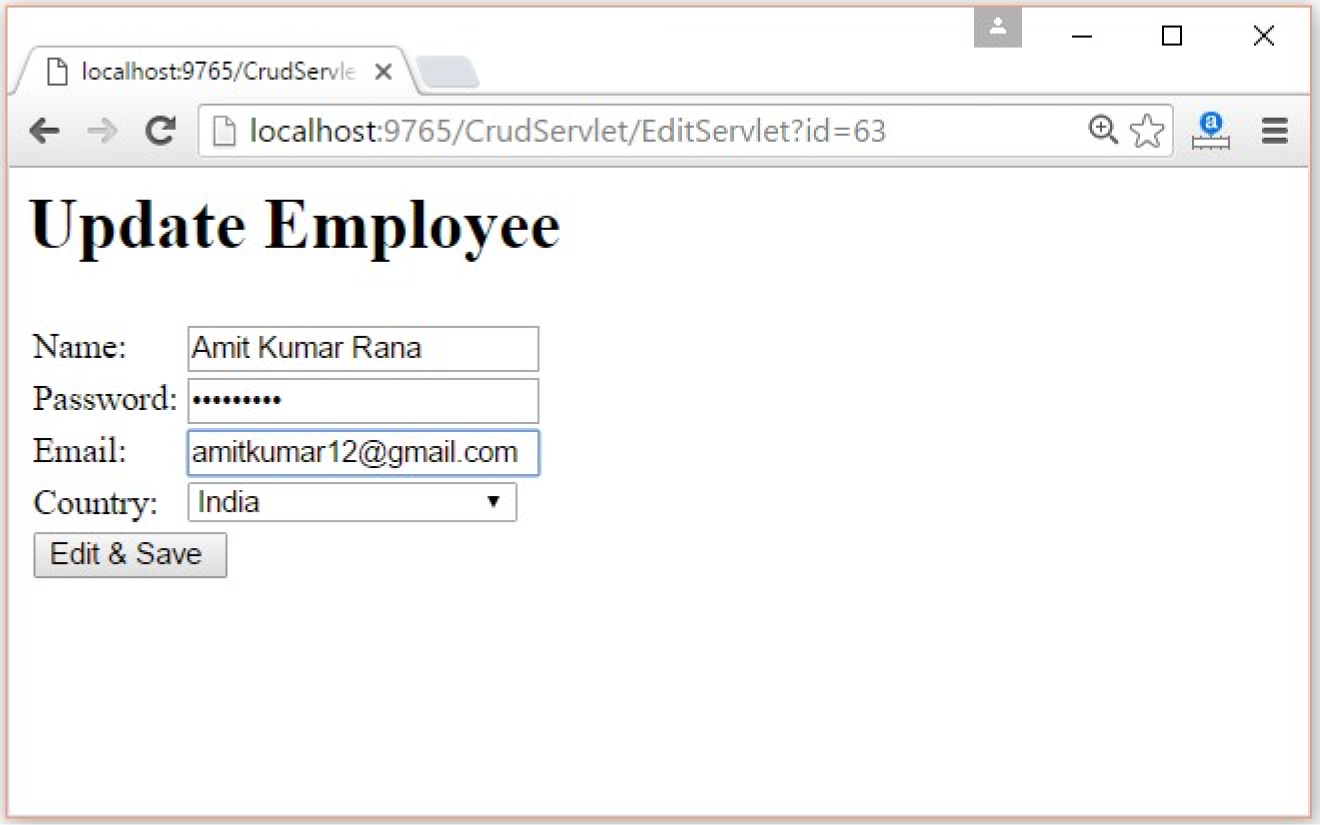

localhost:9765/CrudServle
localhost:9765/CrudServlet/ViewServlet

Add New Employee

Employees List

Id	Name	Password	Email	Country	Edit	Delete
63	Amit Kumar	amtkmjj45	amitkumar@gmail.com	India	edit	delete
61	Rahul Kumar	rahul4000	rahulkk@gmail.com	India	edit	delete
62	Sonoo Jaiswal	sonoobsk	sonoojaiswal1987@gmail.com	India	edit	delete
44	adarsh kumar	kkkkk	adarsh232@gmail.com	India	edit	delete

Click on the update link, to change the data.


localhost:9765/CrudServle
localhost:9765/CrudServlet/EditServlet?id=63

Update Employee

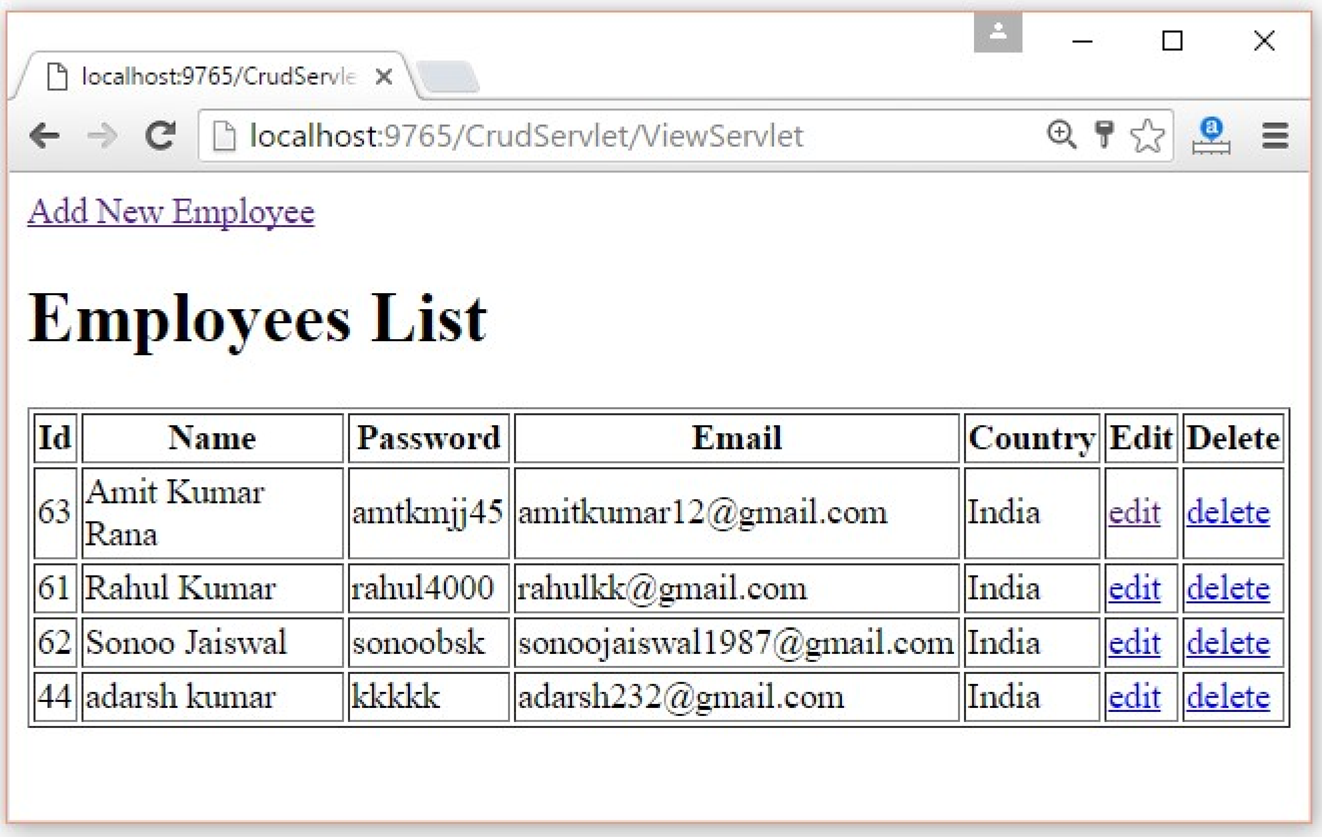
Name: Amit Kumar Rana

Password: •••••••••

Email: amitkumar12@gmail.com

Country: India

After changing information, submit button. You will see that information is changed.

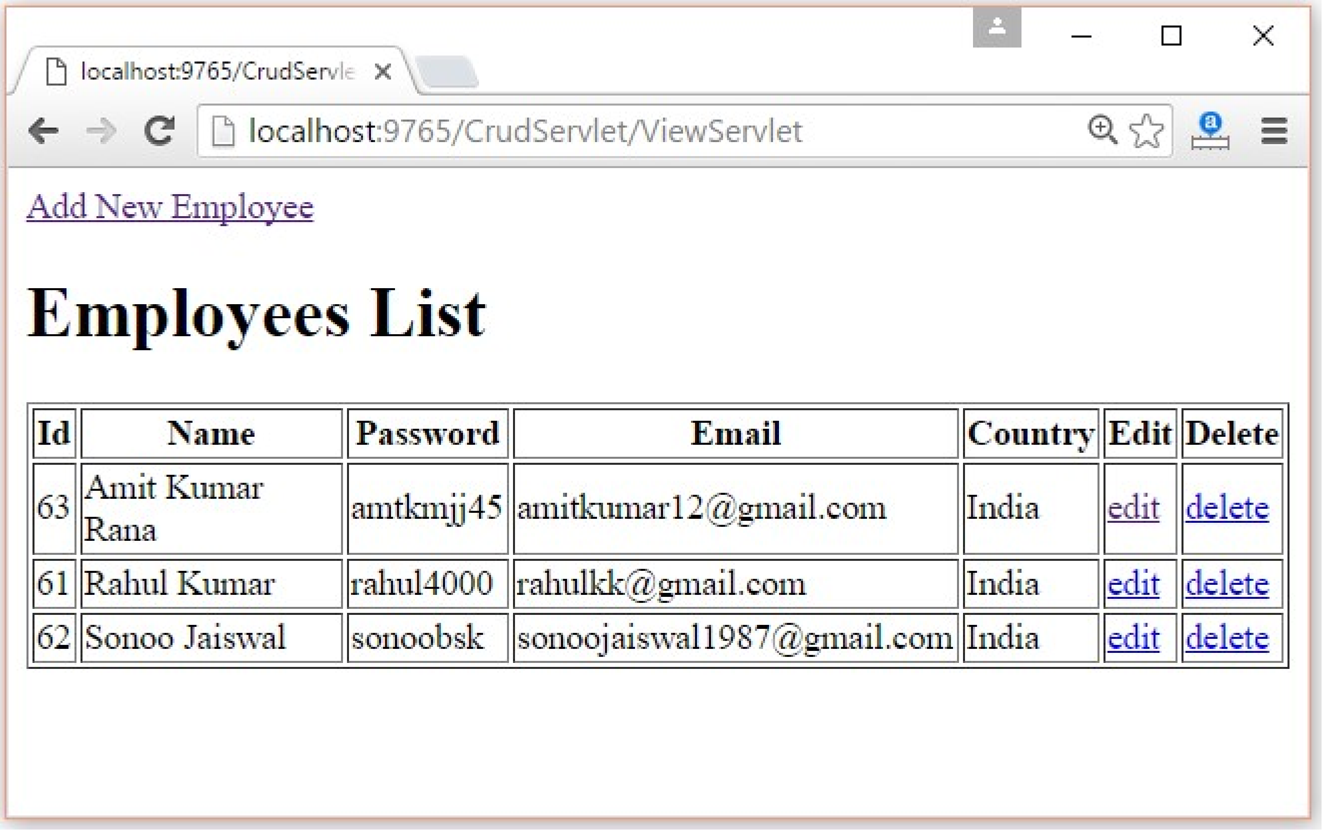

localhost:9765/CrudServle
localhost:9765/CrudServlet/ViewServlet

Add New Employee

Employees List

Id	Name	Password	Email	Country	Edit	Delete
63	Amit Kumar Rana	amtkmjj45	amitkumar12@gmail.com	India	edit	delete
61	Rahul Kumar	rahul4000	rahulkk@gmail.com	India	edit	delete
62	Sonoo Jaiswal	sonoobsk	sonoojaiswal1987@gmail.com	India	edit	delete
44	adarsh kumar	kkkkk	adarsh232@gmail.com	India	edit	delete

Now, click on the delete link to delete the record.



localhost:9765/CrudServle
localhost:9765/CrudServlet/ViewServlet

Add New Employee

Employees List

Id	Name	Password	Email	Country	Edit	Delete
63	Amit Kumar Rana	amtkmjj45	amitkumar12@gmail.com	India	edit	delete
61	Rahul Kumar	rahul4000	rahulkk@gmail.com	India	edit	delete
62	Sonoo Jaiswal	sonoobsk	sonoojaiswal1987@gmail.com	India	edit	delete

Practical 7

Aim: Create Employees table in EMP database. Perform select, insert, update, and delete operations on Employee table using JSP.

index.jsp

```
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>JSP CRUD Example</title>
</head>
<body>
<h1>JSP CRUD Example</h1>
<a href="adduserform.jsp">Add User</a>
<a href="viewusers.jsp">View Users</a>

</body>
</html>
```

adduserform.jsp

```
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Add User Form</title>
</head>
<body>

<jsp:include page="userform.html"></jsp:include>

</body>
</html>
```

userform.html

```
<a href="viewusers.jsp">View All Records</a><br/>

<h1>Add New User</h1>
<form action="adduser.jsp" method="post">
<table>
<tr><td>Name:</td><td><input type="text" name="name"/></td></tr>
<tr><td>Password:</td><td>
<input type="password" name="password"/></td></tr>
<tr><td>Email:</td><td><input type="email" name="email"/></td></tr>
<tr><td>Sex:</td><td>
<input type="radio" name="sex" value="male"/>Male
<input type="radio" name="sex" value="female"/>Female </td></tr>
<tr><td>Country:</td><td>
<select name="country" style="width:155px">
<option>India</option>
<option>Pakistan</option>
<option>Afghanistan</option>
```

```
<option>Berma</option>
<option>Other</option>
</select>
</td></tr>
<tr><td colspan="2"><input type="submit" value="Add User"/></td></tr>
</table>
</form>
```

adduser.jsp

```
<%@page import="com.javatpoint.dao.UserDao"%>
<jsp:useBean id="u" class="com.javatpoint.bean.User"></jsp:useBean>
<jsp:setProperty property="*" name="u"/>

<%
int i=UserDao.save(u);
if(i>0){
response.sendRedirect("adduser-success.jsp");
}else{
response.sendRedirect("adduser-error.jsp");
}
%>
```

User.java

```
package com.javatpoint.bean;
public class User {
private int id;
private String name,password,email,sex,country;
//generate getters and setters
}
```

UserDao.java

```
package com.javatpoint.dao;
import java.sql.*;
import java.util.ArrayList;
import java.util.List;
import com.javatpoint.bean.User;
public class UserDao {

public static Connection getConnection(){
    Connection con=null;
    try{
        Class.forName("com.mysql.jdbc.Driver");
        con=DriverManager.getConnection("jdbc:mysql://localhost:3306/test","","");
    }catch(Exception e){System.out.println(e);}
    return con;
}
public static int save(User u){
    int status=0;
    try{
        Connection con=getConnection();
        PreparedStatement ps=con.prepareStatement(
"insert into register(name,password,email,sex,country) values(?,?,?,?,?)");
        ps.setString(1,u.getName());
        ps.setString(2,u.getPassword());
        ps.setString(3,u.getEmail());
        ps.setString(4,u.getSex());
        ps.setString(5,u.getCountry());
        status=ps.executeUpdate();
    }catch(Exception e){System.out.println(e);}
    return status;
}
public static int update(User u){
    int status=0;
    try{
        Connection con=getConnection();
        PreparedStatement ps=con.prepareStatement(
"update register set name=?,password=?,email=?,sex=?,country=? where id=?");
        ps.setString(1,u.getName());
        ps.setString(2,u.getPassword());
        ps.setString(3,u.getEmail());
        ps.setString(4,u.getSex());
        ps.setString(5,u.getCountry());
        ps.setInt(6,u.getId());
        status=ps.executeUpdate();
    }catch(Exception e){System.out.println(e);}
    return status;
}
public static int delete(User u){
    int status=0;
    try{
        Connection con=getConnection();
        PreparedStatement ps=con.prepareStatement("delete from register where id=?");
        ps.setInt(1,u.getId());
        status=ps.executeUpdate();
    }catch(Exception e){System.out.println(e);}

    return status;
}
public static List<User> getAllRecords(){
    List<User> list=new ArrayList<User>();

    try{
        Connection con=getConnection();
        PreparedStatement ps=con.prepareStatement("select * from register");
        ResultSet rs=ps.executeQuery();
        while(rs.next()){
            User u=new User();
            u.setId(rs.getInt("id"));
            u.setName(rs.getString("name"));
            u.setPassword(rs.getString("password"));
            u.setEmail(rs.getString("email"));
            u.setSex(rs.getString("sex"));
            u.setCountry(rs.getString("country"));
            list.add(u);
        }
    }catch(Exception e){System.out.println(e);}
    return list;
}
public static User getRecordById(int id){
    User u=null;
    try{
        Connection con=getConnection();
        PreparedStatement ps=con.prepareStatement("select * from register where id=?");
        ps.setInt(1,id);
        ResultSet rs=ps.executeQuery();
        while(rs.next()){
            u=new User();
            u.setId(rs.getInt("id"));
            u.setName(rs.getString("name"));
            u.setPassword(rs.getString("password"));
            u.setEmail(rs.getString("email"));
            u.setSex(rs.getString("sex"));
            u.setCountry(rs.getString("country"));
        }
    }catch(Exception e){System.out.println(e);}
    return u;
}
}
```

adduser-success.jsp

```
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Add User Success</title>
</head>
<body>

<p>Record successfully saved!</p>
<jsp:include page="userform.html"></jsp:include>

</body>
</html>
```

adduser-error.jsp

```
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Add User Error</title>
```

```
</head>
<body>

<p>Sorry, an error occurred!</p>
<jsp:include page="userform.html"></jsp:include>

</body>
</html>
```

viewusers.jsp

```
<!DOCTYPE html>

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>View Users</title>
</head>
<body>

<%@page import="com.javatpoint.dao.UserDao,com.javatpoint.bean.*,java.util.*"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>

<h1>Users List</h1>

<%
List<User> list=UserDao.getAllRecords();
request.setAttribute("list",list);
%>

<table border="1" width="90%">
<tr><th>Id</th><th>Name</th><th>Password</th><th>Email</th>
<th>Sex</th><th>Country</th><th>Edit</th><th>Delete</th></tr>
<c:forEach items="${list}" var="u">
<tr><td>${u.getId()}</td><td>${u.getName()}</td><td>${u.getPassword()}</td>
<td>${u.getEmail()}</td><td>${u.getSex()}</td><td>${u.getCountry()}</td>
<td><a href="editform.jsp?id=${u.getId()}">Edit</a></td>
<td><a href="deleteuser.jsp?id=${u.getId()}">Delete</a></td></tr>
</c:forEach>
</table>
<br/><a href="adduserform.jsp">Add New User</a>

</body>
</html>
```

editform.jsp

```
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
```

```
<title>Edit Form</title>
</head>
<body>
<%@page import="com.javatpoint.dao.UserDao,com.javatpoint.bean.User"%>

<%
String id=request.getParameter("id");
User u=UserDao.getRecordById(Integer.parseInt(id));
%>

<h1>Edit Form</h1>
<form action="edituser.jsp" method="post">
<input type="hidden" name="id" value="<%=u.getId() %>"/>
<table>
<tr><td>Name:</td><td>
<input type="text" name="name" value="<%= u.getName()%>"/></td></tr>
<tr><td>Password:</td><td>
<input type="password" name="password" value="<%= u.getPassword()%>"/></td></tr>
<tr><td>Email:</td><td>
<input type="email" name="email" value="<%= u.getEmail()%>"/></td></tr>
<tr><td>Sex:</td><td>
<input type="radio" name="sex" value="male"/>Male
<input type="radio" name="sex" value="female"/>Female </td></tr>
<tr><td>Country:</td><td>
<select name="country">
<option>India</option>
<option>Pakistan</option>
<option>Afghanistan</option>
<option>Berma</option>
<option>Other</option>
</select>
</td></tr>
<tr><td colspan="2"><input type="submit" value="Edit User"/></td></tr>
</table>
</form>

</body>
</html>
```

edituser.jsp

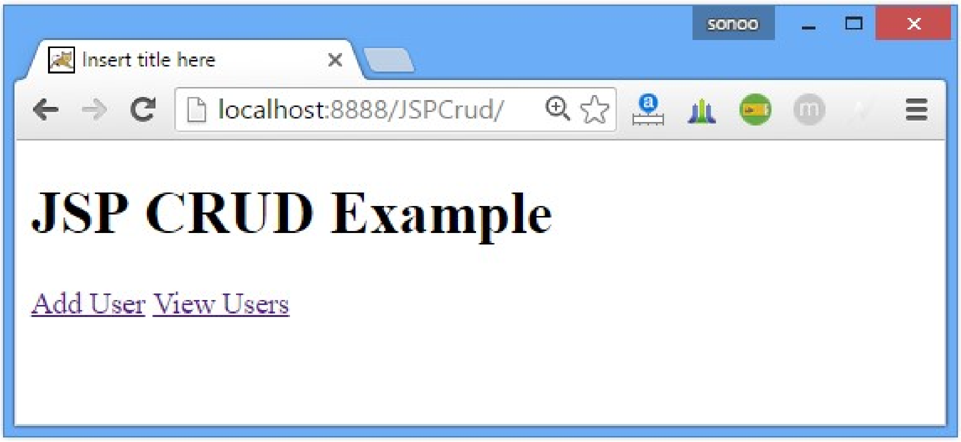
```
<%@page import="com.javatpoint.dao.UserDao"%>
<jsp:useBean id="u" class="com.javatpoint.bean.User"></jsp:useBean>
<jsp:setProperty property="*" name="u"/>
<%
int i=UserDao.update(u);
response.sendRedirect("viewusers.jsp");
%>
```

deleteuser.jsp

```
<%@page import="com.javatpoint.dao.UserDao"%>
```

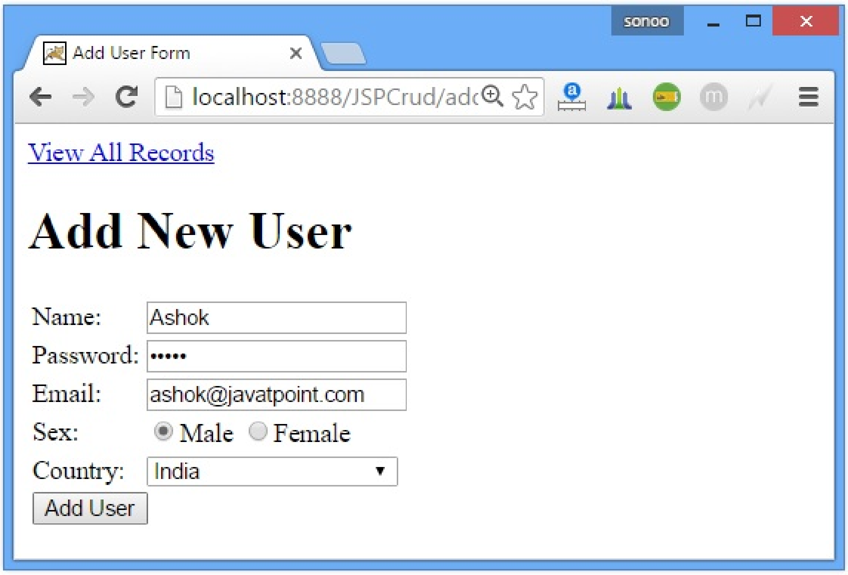
```
<jsp:useBean id="u" class="com.javatpoint.bean.User"></jsp:useBean>
<jsp:setProperty property="*" name="u"/>
<%
UserDao.delete(u);
response.sendRedirect("viewusers.jsp");
%>
```

Output



JSP CRUD Example

Add User View Users

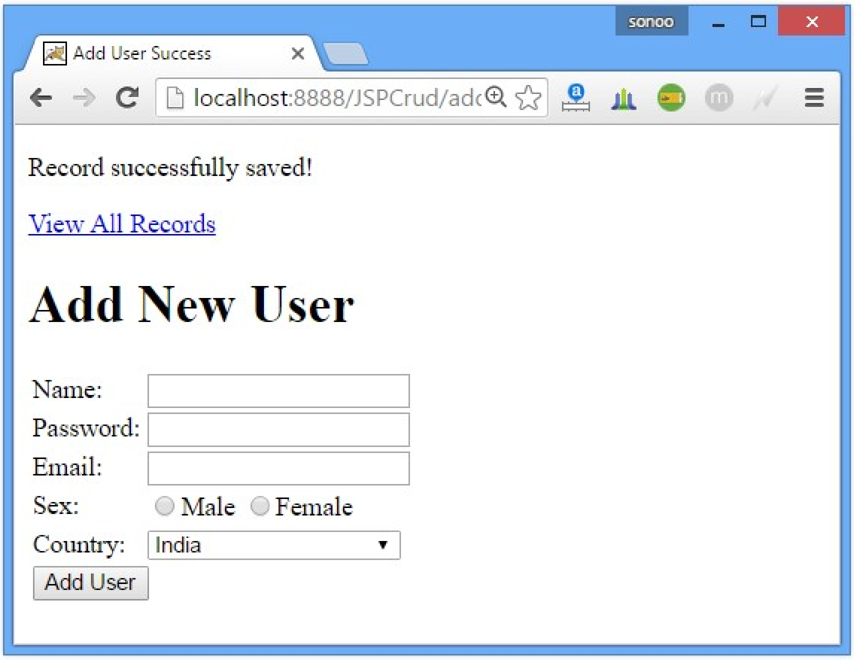


View All Records

Add New User

Name:	Ashok
Password:	•••••
Email:	ashok@javatpoint.com
Sex:	● Male ○ Female
Country:	India ▾

Add User

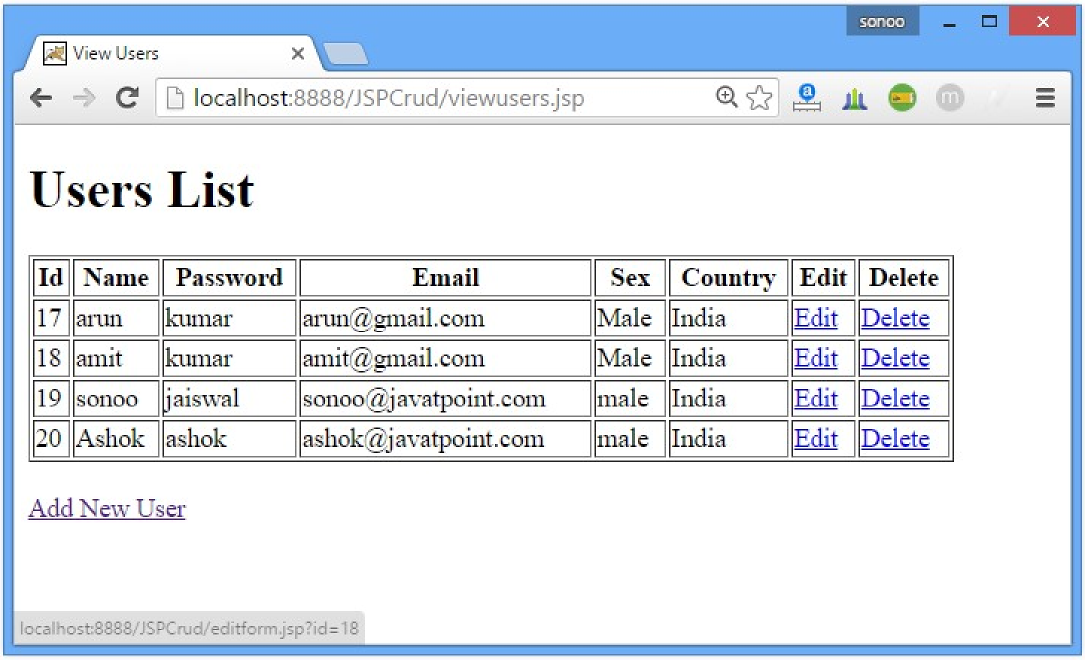


Record successfully saved!

View All Records

Add New User

Name:	
Password:	
Email:	
Sex:	○ Male ○ Female
Country:	India ▾
Add User	

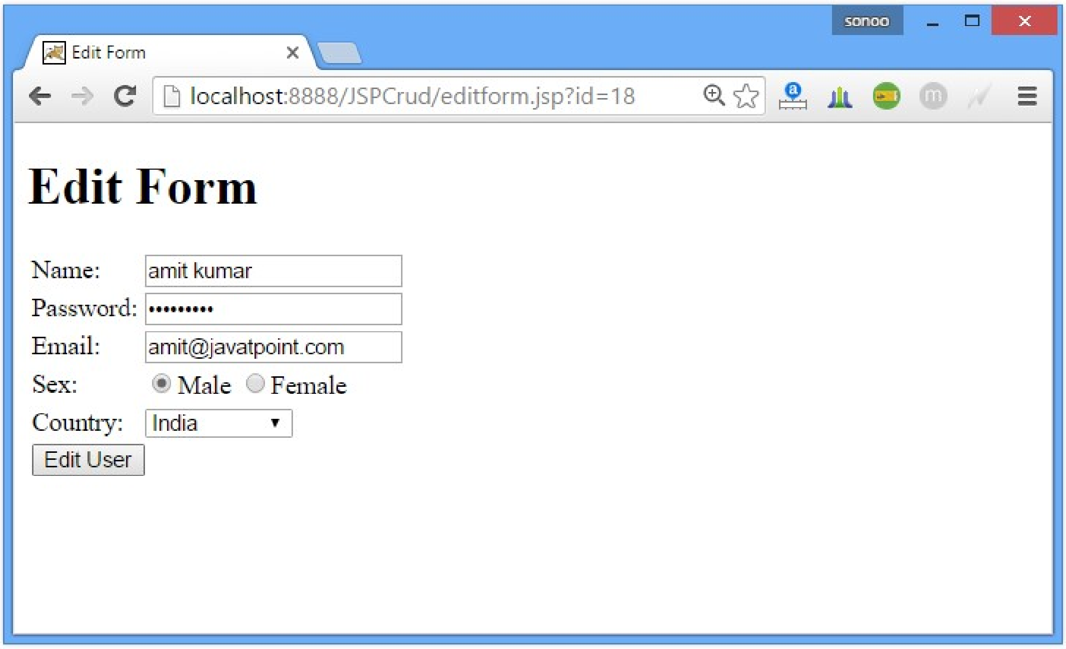

sonoo
View Users
localhost:8888/JSPCrud/viewusers.jsp

Users List

Id	Name	Password	Email	Sex	Country	Edit	Delete
17	arun	kumar	arun@gmail.com	Male	India	Edit	Delete
18	amit	kumar	amit@gmail.com	Male	India	Edit	Delete
19	sonoo	jaiswal	sonoo@javatpoint.com	male	India	Edit	Delete
20	Ashok	ashok	ashok@javatpoint.com	male	India	Edit	Delete

Add New User

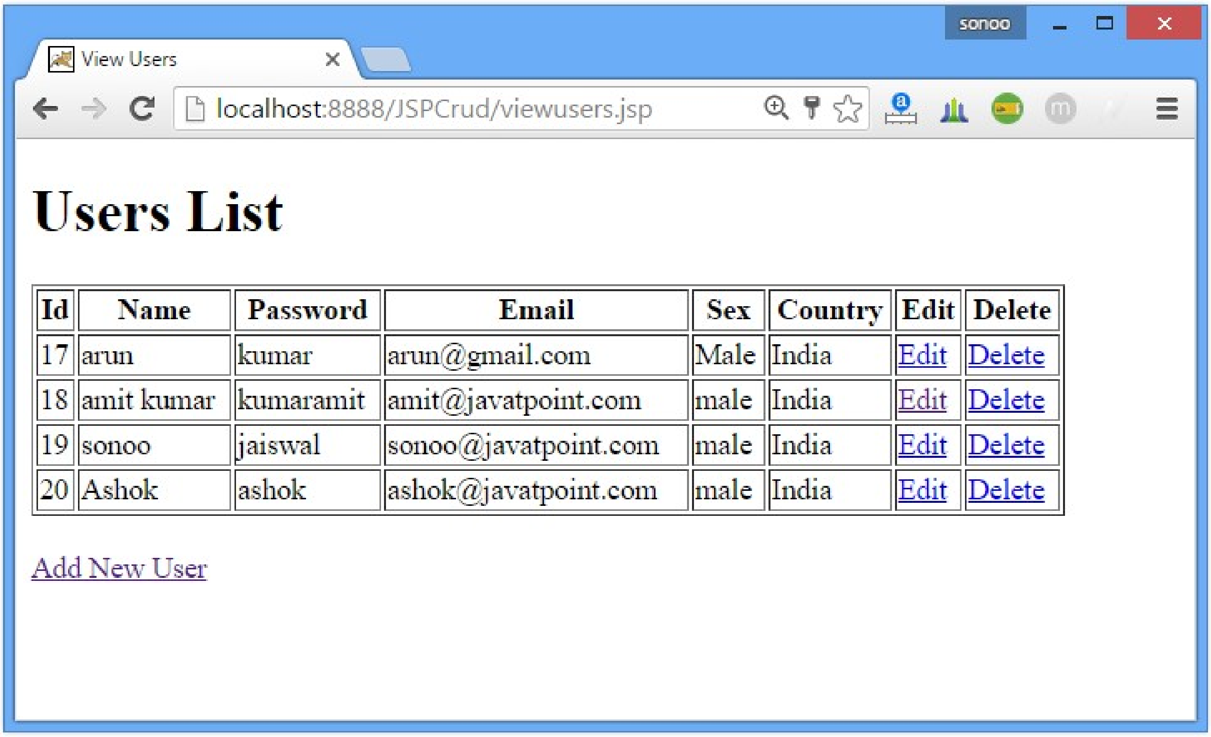
localhost:8888/JSPCrud/editform.jsp?id=18



Edit Form

Name:	amit kumar
Password:	•••••••••
Email:	amit@javatpoint.com
Sex:	● Male ○ Female
Country:	India ▾

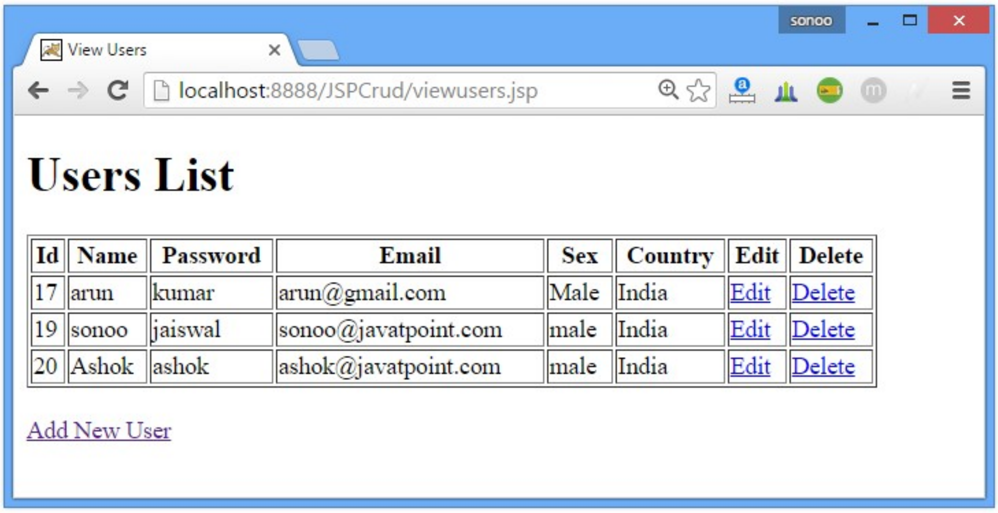
Edit User



Users List

Id	Name	Password	Email	Sex	Country	Edit	Delete
17	arun	kumar	arun@gmail.com	Male	India	Edit	Delete
18	amit kumar	kumaramit	amit@javatpoint.com	male	India	Edit	Delete
19	sonoo	jaiswal	sonoo@javatpoint.com	male	India	Edit	Delete
20	Ashok	ashok	ashok@javatpoint.com	male	India	Edit	Delete

Add New User



Users List

Id	Name	Password	Email	Sex	Country	Edit	Delete
17	arun	kumar	arun@gmail.com	Male	India	Edit	Delete
19	sonoo	jaiswal	sonoo@javatpoint.com	male	India	Edit	Delete
20	Ashok	ashok	ashok@javatpoint.com	male	India	Edit	Delete

Add New User