Question 2

Create a login Java application that authenticate users by checking their details in a database with an interface given below. The application should allow the user to enter their username and password. On clicking login button, if user credentials are in the database the application should display a message “Log in Successful” else it displays message “Wrong password or username”. On clicking reset button the application should clear the username and password textboxes. Create an Ms Access database with 3 user details which the application works with for user authentication.

import javax.swing.\*;

import java.awt.event.\*;

import java.sql.\*;

public class LoginApp implements ActionListener {

JFrame frame;

JTextField userText;

JPasswordField passText;

JButton loginButton, resetButton;

public static void main(String[] args) {

new LoginApp();

}

public LoginApp() {

frame = new JFrame("Login Application");

userText = new JTextField(20);

passText = new JPasswordField(20);

loginButton = new JButton("Login");

resetButton = new JButton("Reset");

loginButton.setActionCommand("LOGIN");

resetButton.setActionCommand("RESET");

loginButton.addActionListener(this);

resetButton.addActionListener(this);

JPanel panel = new JPanel();

panel.add(new JLabel("Username:"));

panel.add(userText);

panel.add(new JLabel("Password:"));

panel.add(passText);

panel.add(loginButton);

panel.add(resetButton);

frame.add(panel);

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setSize(300, 200);

frame.setVisible(true);

}

public void actionPerformed(ActionEvent e) {

if(e.getActionCommand().equals("LOGIN")) {

try {

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

Connection con = DriverManager.getConnection("jdbc:odbc:datasource");

PreparedStatement ps = con.prepareStatement("select \* from users where username=? and password=?");

ps.setString(1, userText.getText());

ps.setString(2, passText.getText());

ResultSet rs = ps.executeQuery();

if(rs.next()) {

JOptionPane.showMessageDialog(null, "Login Successful");

}

else {

JOptionPane.showMessageDialog(null, "Wrong password or username");

}

}catch(Exception ex) {

ex.printStackTrace();

}

}

else if(e.getActionCommand().equals("RESET")) {

userText.setText("");

passText.setText("");

}

}

}

Title: Enhancing Proposal Writing Skills: A Research Design

1. Introduction:

Proposal writing plays a critical role in various academic, professional, and research endeavors. The ability to effectively communicate research ideas, secure funding, and gain approval is crucial for researchers and professionals across disciplines. This research aims to investigate strategies and techniques to enhance proposal writing skills, ultimately improving the success rate of research proposals.

2. Research Objectives:

The main objectives of this research are as follows:

a) To identify common challenges faced by researchers and professionals in writing effective proposals.

b) To explore best practices and techniques employed by successful proposal writers.

c) To develop and evaluate a training program aimed at enhancing proposal writing skills.

d) To assess the impact of improved proposal writing skills on research proposal success rates.

3. Research Questions:

a) What are the common challenges encountered by researchers and professionals in writing research proposals?

b) What are the best practices and strategies employed by successful proposal writers?

c) How effective is a targeted training program in enhancing proposal writing skills?

d) What is the relationship between improved proposal writing skills and research proposal success rates?

4. Research Methodology:

a) Literature Review: A comprehensive review of existing literature will be conducted to identify common challenges and best practices in proposal writing. This will involve examining relevant research articles, books, and resources from reputable sources.

b) Surveys and Interviews: Surveys and interviews will be conducted among researchers and professionals with experience in proposal writing. These data collection methods will help identify common challenges, effective strategies, and training needs.

c) Training Program Development: Based on the findings from the literature review, surveys, and interviews, a targeted training program will be developed. The program will focus on key areas such as proposal structure, writing style, persuasive techniques, and grant writing guidelines.

d) Training Program Evaluation: The effectiveness of the training program will be evaluated through pre- and post-training assessments to measure participants' proposal writing skills. Participants' feedback and satisfaction with the program will also be collected.

e) Proposal Success Rate Analysis: To assess the impact of improved proposal writing skills on success rates, a comparative analysis will be conducted. Proposal success rates before and after the training program will be compared, and statistical analysis will be performed to determine the significance of any observed changes.

5. Ethical Considerations:

Ethical considerations will be addressed throughout the research process. Participants' privacy and confidentiality will be ensured, and informed consent will be obtained. The research will adhere to ethical guidelines regarding data collection, analysis, and reporting.

6. Expected Outcomes:

The research outcomes will contribute to a better understanding of the challenges faced by researchers and professionals in proposal writing. The identification of best practices and the development of a training program will provide valuable resources for improving proposal writing skills. The evaluation of the program's effectiveness and its impact on proposal success rates will further enhance the body of knowledge in this area.

7. Timeline:

The research will be conducted over a period of 12 months, including literature review, data collection, training program development, implementation, and evaluation.

8. Conclusion:

This research aims to address the need for improved proposal writing skills by investigating common challenges, identifying best practices, and developing a targeted training program. By enhancing proposal writing skills, researchers and professionals can increase their chances of securing funding and gaining approval for their research projects, thereby contributing to the advancement of knowledge in their respective fields.

Certainly! Here are some additional details on each section of the research proposal:

1. Introduction:

In the introduction, you can provide a brief overview of the importance of proposal writing and its significance in securing funding and gaining approval for research projects. You can highlight the challenges faced by researchers and professionals in this process and emphasize the need for improved proposal writing skills.

2. Research Objectives:

The research objectives outline the specific goals of the study. You can expand on each objective to provide more context and detail. For example, under objective (c), you can explain that the training program will be designed to address the identified challenges and incorporate the best practices and strategies identified through the research.

3. Research Questions:

The research questions guide the investigation and provide a framework for data collection and analysis. You can elaborate on each research question by discussing the significance and relevance of exploring those specific areas. For example, under research question (d), you can explain that understanding the relationship between improved proposal writing skills and success rates will provide valuable insights into the impact of skill development on proposal outcomes.

4. Research Methodology:

In the research methodology section, you can provide more details on how each method will be implemented. For the literature review, you can explain the criteria for selecting relevant literature sources and the process of synthesizing the findings. In the surveys and interviews section, you can discuss the sample size, selection criteria, and the type of questions that will be asked to gather insights from researchers and professionals. For the training program development, you can describe the process of translating the identified best practices into a structured training curriculum.

5. Ethical Considerations:

Here, you can expand on the ethical considerations that will be addressed in the research. For example, you can discuss how participant privacy and confidentiality will be ensured, how informed consent will be obtained, and how the data will be securely stored and analyzed. You can also mention any ethical guidelines or institutional review board requirements that will be followed.

6. Expected Outcomes:

In this section, you can explain the significance of the expected outcomes. For instance, you can discuss how the identification of common challenges will help researchers and professionals anticipate and address potential obstacles in their proposal writing process. You can also emphasize the practical value of the training program in improving skills and increasing the success rates of research proposals.

7. Timeline:

Provide a high-level timeline for the research activities, indicating the estimated duration for each phase of the study, such as literature review, data collection, program development, implementation, and evaluation. This will give a clear understanding of the research's overall timeline and milestones.

8. Conclusion:

Summarize the main points of the proposal, reiterating the importance of enhancing proposal writing skills and the potential impact of the research outcomes. Emphasize how the findings will contribute to the body of knowledge in proposal writing and ultimately benefit researchers and professionals in various fields.

Remember, the level of detail in each section may vary depending on the specific requirements of your research proposal and the guidelines provided by the funding agency or institution.

Here is the Java program to create a login application that authenticates users with an MS Access database:

```java

import javax.swing.\*;

import java.awt.event.\*;

import java.sql.\*;

public class LoginApp implements ActionListener {

JFrame frame;

JTextField userText;

JPasswordField passText;

JButton loginButton, resetButton;

public static void main(String[] args) {

new LoginApp();

}

public LoginApp() {

frame = new JFrame("Login Application");

userText = new JTextField(20);

passText = new JPasswordField(20);

loginButton = new JButton("Login");

resetButton = new JButton("Reset");

loginButton.setActionCommand("LOGIN");

resetButton.setActionCommand("RESET");

loginButton.addActionListener(this);

resetButton.addActionListener(this);

JPanel panel = new JPanel();

panel.add(new JLabel("Username:"));

panel.add(userText);

panel.add(new JLabel("Password:"));

panel.add(passText);

panel.add(loginButton);

panel.add(resetButton);

frame.add(panel);

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setSize(300, 200);

frame.setVisible(true);

}

public void actionPerformed(ActionEvent e) {

if(e.getActionCommand().equals("LOGIN")) {

try {

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

Connection con = DriverManager.getConnection("jdbc:odbc:datasource");

PreparedStatement ps = con.prepareStatement("select \* from users where username=? and password=?");

ps.setString(1, userText.getText());

ps.setString(2, passText.getText());

ResultSet rs = ps.executeQuery();

if(rs.next()) {

JOptionPane.showMessageDialog(null, "Login Successful");

}

else {

JOptionPane.showMessageDialog(null, "Wrong password or username");

}

}catch(Exception ex) {

ex.printStackTrace();

}

}

else if(e.getActionCommand().equals("RESET")) {

userText.setText("");

passText.setText("");

}

}

}

```

This program authenticates the credentials against data in an MS Access database and displays appropriate messages.

1. Write a program to find the sum of elements in an array.

2. Write a program to find the largest element in an array.

3. Write a program to sort an array in ascending order.

4. Write a program to remove duplicates from an array.

5. Write a program to reverse a string using an array.

Sure! Here's the implementation of the requested programs in Java:

1. To find the sum of elements in an array:

```java

public class ArraySum {

public static int findSum(int[] arr) {

int sum = 0;

for (int i = 0; i < arr.length; i++) {

sum += arr[i];

}

return sum;

}

public static void main(String[] args) {

int[] arr = {1, 2, 3, 4, 5};

int sum = findSum(arr);

System.out.println("Sum of elements in the array: " + sum);

}

}

```

2. To find the largest element in an array:

```java

public class LargestElement {

public static int findLargest(int[] arr) {

int largest = arr[0];

for (int i = 1; i < arr.length; i++) {

if (arr[i] > largest) {

largest = arr[i];

}

}

return largest;

}

public static void main(String[] args) {

int[] arr = {5, 3, 9, 1, 6};

int largest = findLargest(arr);

System.out.println("Largest element in the array: " + largest);

}

}

```

3. To sort an array in ascending order:

```java

import java.util.Arrays;

public class ArraySort {

public static void sortArray(int[] arr) {

Arrays.sort(arr);

}

public static void main(String[] args) {

int[] arr = {5, 3, 9, 1, 6};

sortArray(arr);

System.out.println("Sorted array: " + Arrays.toString(arr));

}

}

```

4. To remove duplicates from an array:

```java

import java.util.Arrays;

import java.util.HashSet;

public class RemoveDuplicates {

public static int[] removeDuplicates(int[] arr) {

HashSet<Integer> set = new HashSet<>();

for (int i = 0; i < arr.length; i++) {

set.add(arr[i]);

}

int[] result = new int[set.size()];

int index = 0;

for (int num : set) {

result[index++] = num;

}

return result;

}

public static void main(String[] args) {

int[] arr = {1, 2, 3, 2, 4, 1, 5};

int[] result = removeDuplicates(arr);

System.out.println("Array after removing duplicates: " + Arrays.toString(result));

}

}

```

5. To reverse a string using an array:

```java

public class ReverseString {

public static String reverse(String str) {

char[] arr = str.toCharArray();

int left = 0;

int right = arr.length - 1;

while (left < right) {

char temp = arr[left];

arr[left] = arr[right];

arr[right] = temp;

left++;

right--;

}

return new String(arr);

}

public static void main(String[] args) {

String str = "Hello, World!";

String reversedStr = reverse(str);

System.out.println("Reversed string: " + reversedStr);

}

}

```

These programs perform the desired operations efficiently. You can execute each program individually to see the results.