wrote a program in python3.6 that the user creates a username and password. then the program checks if that username is stored if not it creates the username and password permanently.I am having trouble linking username and password lists as 'jason' has the password 'oero'.

so that the

```
if sentence == stored_username[0:] and
sentence2 == stored_password[0:]:
    print(Aceppted)
```

runs as 'jason' is the the username and oero is the password.

the other problem is when I ran the program it tried to run the entire list

so you cant choose just one value of the lists. This is what I have so far. the program is also set to quit if the user gets their username and password wrong 3 times. which works fine. Thank you! The code runs as is, the way its supposed too.

```
username= input('Create Username')
password= input('Create Password')
stored_username =['jason' , 'nicole',username]
stored_password =['oeros', 'chance',password]

print(stored_username[0:])

trials =0
def sign_in():
```

```
    global username
    global password
    global stored_username

    global stored_password



    sentence= input('Enter Username')
    print(sentence)
    sentence2 = input('Enter Password')
    print(sentence2)
    global trials
    Aceppted= 'Welcome to Bacall Land'
    wrong=('Wrong Username or Password
')

    if sentence == stored_username[0:]
and sentence2 == stored_password[0:]:
        print(Aceppted)
```

```
    else:
        print(wrong)
    while sentence != stored_username[0:]
and sentence2!= stored_password[0:]:
        trials += 1
        print(trials)
        (trials <=3 and sign_in())
        if trials >= 3:
            break
    if sentence== stored_username[0:] and
sentence2 == stored_password[0:]:
        print(Aceppted)
    else:
        quit()
```

```
{                    updateStatementE
    static
    {
        //STEP 1 : Registerin
        try
        {
            Class.forName("or
        }
        catch (ClassNotFoundE
        {
            System.out.printl
        }
    }

    public static void main(S
    {
        Connection con = null
        Statement stmt = null
        try
        {
            //Database Creden
            String URL = "jdb
            String username =
            String password =
```

```
//Database Credentials

String URL = "jdbc:oracle:thi

String username = "username";

String password = "password";

//STEP 2 : Creating The Conne

con = DriverManager.getConnec

//STEP 3 : Creating The State

stmt = con.createStatement();

//Constructing The SQL Query

String sql = "DELETE FROM EMF

//Step 4 : Executing The Quer

//We are using executeUpdate(

int i = stmt.executeUpdate(sc

if(i != 0)
{
    System.out.println("Recor
}
else
{
    System.out.println("Recor
}

:h (SQLException e)
```

```
atch (SQLException e)
    e.printStackTrace();
inally
    //STEP 5 : Closing The DB I
    //Closing the Statement ob
    try
    {
        if(stmt!=null)
        {
            stmt.close();
            stmt=null;
        }
    }
    catch (SQLException e)
    {
        e.printStackTrace();
    }
    //Closing the Connection ol
    try
    {
        if(con!=null)
        {
            con.close();
            con=null;
        }
    }
    catch (SQLException e)
    {
        e.printStackTrace();
    }
```

```
        if(stmt!=null)
        {
            stmt.close();
            stmt=null;
        }
    }
    catch (SQLException e)
    {
        e.printStackTrace();
    }

    //Closing the Connection obje

    try
    {
        if(con!=null)
        {
            con.close();
            con=null;
        }
    }
    catch (SQLException e)
    {
        e.printStackTrace();
    }
```

```
'/Database Credentials
String URL = "jdbc:oracle:th
String username = "username"
String password = "password"
'/STEP 2 : Creating The Conr
on = DriverManager.getConne
'/STEP 3 : Creating The Stat
tmt = con.createStatement()
/Constructing The SQL Query
tring sql = "UPDATE EMPLOYE
          "LAST_NAME='Malhotra
/Step 4 : Executing The Que
/We are using executeUpdate
nt i = stmt.executeUpdate(s
f(i != 0)
   System.out.println("Recc
lse
   System.out.println("Recc

 (SQLException e)
```

```
import java.sql.*;

public class CreateTableExamp
{
    static
    {
        //STEP 1 : Registerin

        try
        {
            Class.forName("or
        }
        catch (ClassNotFoundE
        {
            System.out.println
        }
    }

    public static void main(S
    {
        Connection con = null

        Statement stmt = null

        try
        {
            //Database Creden

            String URL = "jdb

            String username =

            String password =

            //STEP 2 : Creati

            con = DriverManag

            //STEP 3 : Creati
```

```
ch (SQLException e)

 e.printStackTrace();

ally

 //STEP 5 : Closing The DB Re

 //Closing the Statement obje

 try
 {
     if(stmt!=null)
     {
         stmt.close();
         stmt=null;
     }
 }
 catch (SQLException e)
 {
     e.printStackTrace();
 }

 //Closing the Connection obj

 try
 {
     if(con!=null)
     {
         con.close();
         con=null;
     }
 }
 catch (SQLException e)
 {
     e.printStackTrace();
 }
```

```
//Database Credentials

String URL = "jdbc:oracle:

String username = "usernam

String password = "passwor

//STEP 2 : Creating The Co

con = DriverManager.getCon

//STEP 3 : Creating The St

stmt = con.createStatement

//Constructing The SQL Que

String sql = "CREATE TABLE
             "ID NUMBER NOT NUL
             "FIRST_NAME VARCHA
             "LAST_NAME VARCHAR
             "DISIGNATION VARCH

//Step 4 : Executing The Q

//We are using executeUpda

int i = stmt.executeUpdate

if(i == 0)
{
    System.out.println("Tal
}
else
{
    System.out.println("Tal
}
```

```
        }
        catch (ClassNotFoundE
        {
            System.out.printl
        }
    }

    public static void main(S
    {
        Connection con = null

        Statement stmt = null

        try
        {
            //Database Creden

            String URL = "jdb

            String username =

            String password =

            //STEP 2 : Creati

            con = DriverManag

            //STEP 3 : Creati

            stmt = con.create

            //Constructing Th

            String sql = "DEL

            //Step 4 : Execut
```

```
x = 10
x = x + 10
x = x - 5
print(x)
x, y = x - 2, 22
print(x, y)
```

Output:

```
15
13 22
```

Step-by-step explanation:

```
x = 10   #value assigned to x is 10
x = x + 10   #variable 'x' has been
assigned the value of x + 10, ⇒ 10 + 10
x = x - 5    #variable 'x' has been
assigned the value of x - 5, ⇒ 20 - 5
print(x)    #prints the value of 'x', ⇒ 15
x, y = x - 2, 22    #variables 'x' and 'y'
```

could separate the values in the print statement.

Correct code (1):

```
name = "aman"
age = "26"
print("Your name and age is", name + age)
```

[Keep in mind, doing so, there will be no space between the name and the age. This is why the second code (given below) is most preferable.)]

Correct code (2):

```
name = "aman"
age = 26
print("Your name and age is", name, age)
```

```
    print("Your name and age is", name +
age)
```

TypeError: can only concatenate str (not "int") to str

Error:

The **data type of the variable 'name' is that of a string value**, whereas the **data type assigned to variable 'age' is an integer type.** You **cannot add a string value and an integer data type.** Doing so will only result in an error.

Solution:

You can convert the data type assigned to 'age' to that of a string instead by adding the value in quotes, or you could separate the values in the print statement.

The Bank Account Management System is an application for maintaining a person's account in a bank. In this project I tried to show the working of a banking account system and cover the basic functionality of a Bank Account Management System. To develop a project for solving financial applications of a customer in banking environment in order to nurture the needs of an end banking user by providing various ways to perform banking tasks. Also to enable the user's work space to have additional functionalities which are not provided under a conventional banking project. The Bank Account Management System

the user's work space to have additional functionalities which are not provided under a conventional banking project. The Bank Account Management System undertaken as a project is based on relevant technologies. The main aim of this project is to develop software for Bank Account Management System. This project has been developed to carry out the processes easily and quickly, which is not possible with the manuals systems, which are overcome by this software. This project is developed using PHP, HTML language and MYSQL use for database connection. Creating and managing requirements is a challenge of IT, systems and product development projects or indeed for any activity where you have to manage a contractual relationship. Organization need to effectively define and manage requirements to ensure they are meeting

challenge of IT, systems and product development projects or indeed for any activity where you have to manage a contractual relationship. Organization need to effectively define and manage requirements to ensure they are meeting needs of the customer, while proving compliance and staying on the schedule and within budget. The impact of a poorly expressed requirement can bring a business out of compliance or even cause injury or death. Requirements definition and management is an activity that can deliver a high, fast return on investment. The project analyzes the system requirements and then comes up with the requirements specifications. It studies other related systems and then come up with system specifications. The system is then designed in accordance with specifications to satisfy the requirements. The system design is

functionalities which are not provided under a conventional banking project. The Bank Account Management System undertaken as a project is based on relevant technologies. The main aim of this project is to develop software for Bank Account Management System. This project has been developed to carry out the processes easily and quickly, which is not possible with the manuals systems, which are overcome by this software. This project is developed using PHP, HTML language and MYSQL use for database connection. Creating and managing requirements is a challenge of IT, systems and product development projects or indeed for any activity where you have to manage a contractual relationship. Organization need to effectively define and manage requirements to ensure they are meeting

that can deliver a high, fast return on investment. The project analyzes the system requirements and then comes up with the requirements specifications. It studies other related systems and then come up with system specifications. The system is then designed in accordance with specifications to satisfy the requirements. The system design is then implemented with MYSQL, PHP and HTML. The system is designed as an interactive and content management system. The content management system deals with data entry, validation confirm and updating whiles the interactive system deals with system interaction with the administration and users. Thus, above features of this project will save transaction time and therefore increase the efficiency of the system.

challenge of IT, systems and product development projects or indeed for any activity where you have to manage a contractual relationship. Organization need to effectively define and manage requirements to ensure they are meeting needs of the customer, while proving compliance and staying on the schedule and within budget. The impact of a poorly expressed requirement can bring a business out of compliance or even cause injury or death. Requirements definition and management is an activity that can deliver a high, fast return on investment. The project analyzes the system requirements and then comes up with the requirements specifications. It studies other related systems and then come up with system specifications. The system is then designed in accordance with specifications to satisfy the requirements. The system design is

and within budget. The impact of a poorly expressed requirement can bring a business out of compliance or even cause injury or death. Requirements definition and management is an activity that can deliver a high, fast return on investment. The project analyzes the system requirements and then comes up with the requirements specifications. It studies other related systems and then come up with system specifications. The system is then designed in accordance with specifications to satisfy the requirements. The system design is then implemented with MYSQL, PHP and HTML. The system is designed as an interactive and content management system. The content management system deals with data entry, validation confirm and updating whiles the interactive system deals with system