**5-a.Multilevel User Authentication System**

**Code :**

package Experiment5;

import java.util.\*;

class UserAuthentication

{

private String username,password;

private String email,otp;

boolean isBioMetricEnabled;

public UserAuthentication(String username,String pwd)

{

this.username = username;

this.password = pwd;

System.out.println("User registered with Username and Password..");

}

public UserAuthentication(String email)

{

this.email=email;

this.otp=generateOTP();

System.out.println("User registered with Email "+this.email+" and OTP : "+this.getOTP());

}

public UserAuthentication(boolean isBiometricEnabled)

{

this.isBioMetricEnabled=isBiometricEnabled;

System.out.println("User registered with their Biometric");

}

public String generateOTP()

{

Random rand = new Random();

return String.format("%06d", rand.nextInt(1000000));

}

public String getOTP()

{

return this.otp;

}

public boolean login(String uname, String pwd)

{

if(this.username!=null && this.username.equals(uname) && this.password.equals(pwd))

{

System.out.println("Login Successful: Valid Username & Password");

return true;

}

System.out.println("Login failed: Invalid Username / Password");

return false;

}

public boolean login(String email, int otp)

{

if(this.email!=null && this.otp.equals(String.valueOf(otp)))

{

System.out.println("Login Successful: Valid Email & OTP");

return true;

}

System.out.println("Login failed: Invalid Email / OTP");

return false;

}

public boolean login()

{

if(this.isBioMetricEnabled)

{

System.out.println("Login Successful: valid biometric");

return true;

}

System.out.println("Login failed: Invalid biometric");

return false;

}

}

public class UserAuthenticationSystem {

public static void main(String[] args) {

UserAuthentication ua ;

Scanner sc = new Scanner(System.in);

String uname,pwd,email;

System.out.println("----User Authentication System----");

System.out.println("==================================");

System.out.println("1. Login using Username & Password\n2. Login using Email & OTP\n3. Login using Biometric");

System.out.println("Enter your choice : ");

int choice = sc.nextInt();

sc.nextLine();

switch(choice)

{

case 1:

ua = new UserAuthentication("solai@123","12345");

System.out.println("Enter Username : ");

uname = sc.nextLine();

System.out.println("Enter Password : ");

pwd = sc.nextLine();

ua.login(uname,pwd);

break;

case 2:

System.out.println("Enter Email : ");

email = sc.nextLine();

ua = new UserAuthentication(email);

System.out.println("Enter OTP to verify : ");

int otp = sc.nextInt();

ua.login(email,otp);

break;

case 3:

ua = new UserAuthentication(false);

ua.login();

break;

default:

System.out.println("Please select the valid choice");

break;

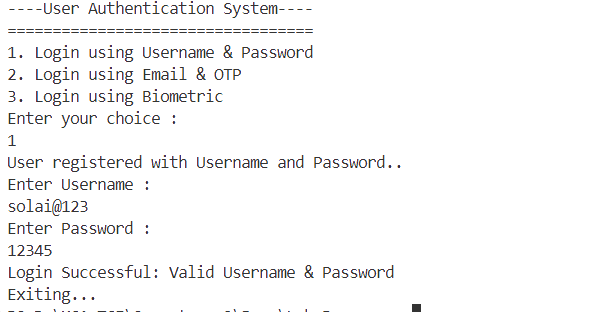
}

System.out.println("Exiting...");

}

}

**Output :**

****

**5-b.Stock Market Prediction System**

**Code :**

import java.util.Scanner;

class StockAnalysis {

private String stockName;

private String stockType;

private double investmentAmount;

private double annualReturnRate;

// Constructor 1: Initialize stockName and stockType (Default investment amount)

public StockAnalysis(String stockName, String stockType) {

this.stockName = stockName;

this.stockType = stockType;

this.investmentAmount = 1000;

this.annualReturnRate = assignReturnRate(stockType);

}

// Constructor 2: Initialize stockName, stockType, and investmentAmount

public StockAnalysis(String stockName, String stockType, double investmentAmount) {

this(stockName, stockType);

this.investmentAmount = investmentAmount;

}

// Assign return rate based on stock type

private double assignReturnRate(String stockType) {

switch (stockType.toLowerCase()) {

case "high":

return 15.0;

case "medium":

return 10.0;

case "low":

return 5.0;

default:

return 3.0;

}

}

// Method Overloading: Predict returns for 1 year (Short-term)

public double predictReturns() {

return investmentAmount \* Math.pow(1 + (annualReturnRate / 100), 1);

}

// Method Overloading: Predict returns for multiple years

public double predictReturns(int years) {

return investmentAmount \* Math.pow(1 + (annualReturnRate / 100), years);

}

// Display stock performance

public void displayStockPerformance(int years) {

double futureValue = (years == 1) ? predictReturns() : predictReturns(years);

System.out.printf("Your future investment value on "+this.stockName+" after %d years: Rs. %.2f\n", years, futureValue);

}

}

public class StockMarket{

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.println("Stock Market Prediction System");

System.out.println("==============================\n\n");

System.out.print("Enter stock name: ");

String stockName = scanner.nextLine();

System.out.print("Enter stock type (High/Medium/Low): ");

String stockType = scanner.nextLine();

System.out.print("Do you want to enter an investment amount? (yes/no): ");

String choice = scanner.nextLine().toLowerCase();

StockAnalysis stock;

if (choice.equals("yes")) {

System.out.print("Enter investment amount: ");

double investmentAmount = scanner.nextDouble();

stock = new StockAnalysis(stockName, stockType, investmentAmount);

} else {

stock = new StockAnalysis(stockName, stockType);

}

System.out.print("Enter investment duration (1, 3, or 5+ years): ");

int years = scanner.nextInt();

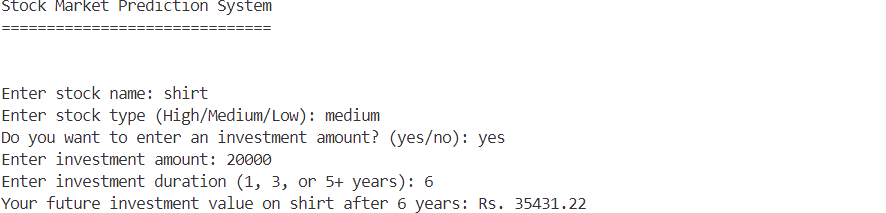
stock.displayStockPerformance(years);

scanner.close();

}

}

**Output :**

****