ASSIGNMENT

NAME: RITABRATA GANGULY

**Calculator App:**

INDEX:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="styles.css">

    <title>Calculator</title>

</head>

<body>

    <div class="calculator">

        <div class="output">

            <div class="result" id="result">0</div>

        </div>

        <div class="buttons">

            <button class="special" onclick="clearScreen()">C</button>

            <button class="special" onclick="backspace()">&#9003;</button>

            <button onclick="appendToResult('0')">0</button>

            <button onclick="appendToResult('/')">/</button>

            <button onclick="appendToResult('7')">7</button>

            <button onclick="appendToResult('8')">8</button>

            <button onclick="appendToResult('9')">9</button>

            <button onclick="appendToResult('\*')">\*</button>

            <button onclick="appendToResult('4')">4</button>

            <button onclick="appendToResult('5')">5</button>

            <button onclick="appendToResult('6')">6</button>

            <button onclick="appendToResult('-')">-</button>

            <button onclick="appendToResult('1')">1</button>

            <button onclick="appendToResult('2')">2</button>

            <button onclick="appendToResult('3')">3</button>

            <button onclick="appendToResult('+')">+</button>

            <button class="equals" onclick="calculate()">=</button>

            <button class="special" onclick="appendToResult('+')">+</button>

            <button class="special" onclick="calculatePercentage()">%</button>

        </div>

    </div>

    <script src="script.js"></script>

</body>

</html>

CSS:

body {

    font-family: Arial, sans-serif;

    margin: 0;

    padding: 0;

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

    background-color: #f2f2f2;

}

.calculator {

    width: 300px;

    border: 1px solid #ccc;

    border-radius: 5px;

    background-color: #fff;

    box-shadow: 0 0 10px rgba(0, 0, 0, 0.2);

}

.output {

    padding: 20px;

    text-align: right;

    font-size: 24px;

    border-bottom: 1px solid #ccc;

}

.result {

    margin: 0;

    padding: 0;

}

.buttons {

    display: grid;

    grid-template-columns: repeat(4, 1fr);

    gap: 1px;

    background-color: #eee;

}

button {

    padding: 15px;

    font-size: 18px;

    border: none;

    background-color: #fff;

    cursor: pointer;

    transition: background-color 0.3s;

}

button:hover {

    background-color: #f2f2f2;

}

.special {

    background-color: #f2f2f2;

    font-weight: bold;

}

.equals {

    background-color: #ff9800;

    color: #fff;

}

.equals:hover {

    background-color: #f57c00;

}

JS:

let result = document.getElementById('result');

let currentInput = '';

function appendToResult(value) {

    currentInput += value;

    result.innerText = currentInput;

}

function calculate() {

    try {

        currentInput = eval(currentInput).toString();

        result.innerText = currentInput;

    } catch (error) {

        result.innerText = 'Error';

    }

}

function clearScreen() {

    currentInput = '';

    result.innerText = '0';

}

function backspace() {

    currentInput = currentInput.slice(0, -1);

    result.innerText = currentInput;

}

function calculatePercentage() {

    try {

        currentInput = (eval(currentInput) / 100).toString();

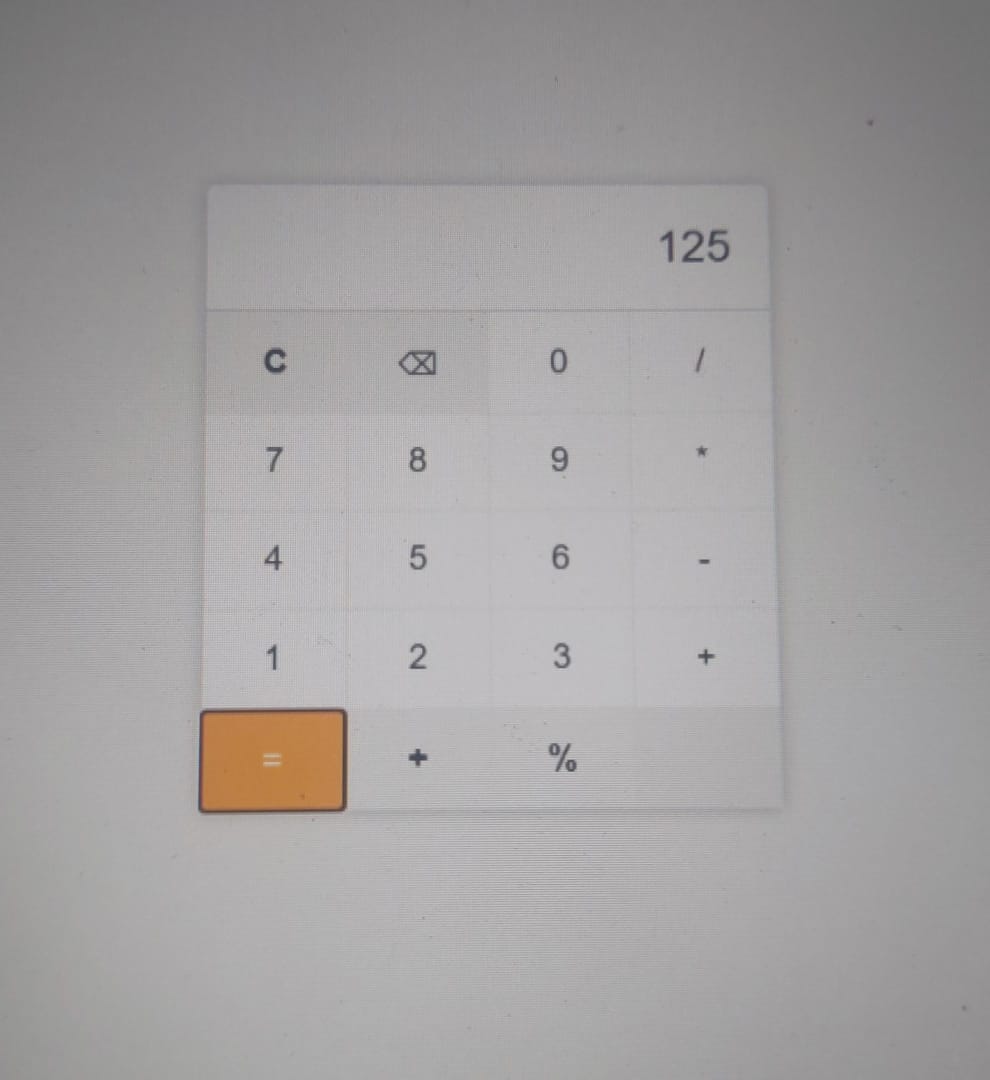
        result.innerText = currentInput;

    } catch (error) {

        result.innerText = 'Error';

    }

}

****

**JAVA Program: E-com website**

import java.util.\*;

class User {

private boolean role;

private String loginID;

private String password;

public User(){

}

public User(boolean role, String lid, String pass){

this.role = role;

loginID = lid;

password = pass;

}

private boolean verifyUser(){

if(loginID != null && password != null && role){

return true;

}

return false;

}

}

class Customer extends User{

public String id, address, phone, email;

public Customer(String i, String a, String p, String e){

this.id = i;

this.address = a;

this.phone = p;

this.email = e;

}

public Customer(String phone){

this.phone = phone;

}

public String login(String mail, String num){

if(mail == email && num == phone){

return "Successful Login...";

}

return "Please try with proper credentials!!!";

}

public String register(String addr, String mail, String num){

if(addr != null && mail != null && num != null){

return "Account created successfully...";

}

return "Please enter a valid credential!!!";

}

public void updateProfile(String addr, String mail, String num){

address = addr;

email = mail;

phone = num;

}

}

class shoppingCart{

private Date created;

private Customer c;

public shoppingCart(Date d){

created = d;

}

public shoppingCart(Date d, Customer c){

this.created = d;

this.c = c;

}

public boolean addCartItem(){

return true;

}

public void checkOut(){

System.out.println("Order summary : ");

}

public void viewCartDetails(){

System.out.println("Showing cart details : ");

}

public void updateQuantity(){

System.out.println("Quantity updated successfully...");

}

}

class Reviews {

private String reviewId;

private String customerId;

private String reviewContent;

private Integer rating;

private String parentId;

private String productId;

private Customer customer;

public Reviews(String rid, String cid, String rc, Integer r, String pid, String prodId ){

reviewId = rid;

customerId = cid;

reviewContent = rc;

rating = r;

parentId = pid;

productId = prodId;

}

public Reviews(String rid, String cid, String rc, Integer r, String pid, String prodId, Customer customer){

reviewId = rid;

customerId = cid;

reviewContent = rc;

rating = r;

parentId = pid;

productId = prodId;

this.customer = customer;

}

public void addReview(String reviewContent){

if(reviewContent.length() > 0){

System.out.println("Review added successfully...");

}

else{

System.out.println("Please write something to add!!!");

}

}

public void deleteReview(String revId){

if(customer.id != null && revId != null){

System.out.println("Review deleted successfully...");

}

else if(customer.id == null){

System.out.println("Please login to delete your review!!!");

}

else if(revId == null){

System.out.println("No review found!!!");

}

}

public void editReview(String content){

if(content.length() > 0 && content != null){

System.out.println("Updated the review...");

}

else{

System.out.println("Please write something to update!!!");

}

}

}

class Orders{

private String id;

private String sellerId;

private String customerId;

private String productId;

private String totalAmount;

private Date orderDate;

private String address;

private Date deliveredDate;

private String deliveryStatus;

private Customer c;

public Orders(String id, String sellerId, String customerId, String productId, String totalAmount, Date orderDate, String address, Date deliveredDate, String deliveryStatus){

this.id = id;

this.sellerId = sellerId;

this.customerId = customerId;

this.productId = productId;

this.totalAmount = totalAmount;

this.orderDate = orderDate;

this.address = address;

this.deliveredDate = deliveredDate;

this.deliveryStatus = deliveryStatus;

}

public Orders(String id, String sellerId, String customerId, String productId, String totalAmount, Date orderDate, String address, Date deliveredDate, String deliveryStatus, Customer c){

this.id = id;

this.sellerId = sellerId;

this.customerId = customerId;

this.productId = productId;

this.totalAmount = totalAmount;

this.orderDate = orderDate;

this.address = address;

this.deliveredDate = deliveredDate;

this.deliveryStatus = deliveryStatus;

this.c = c;

}

public void placeOrder(String totalAmount, String address, Customer c){

if(c.address.equals(address) && totalAmount.length()>0 && !totalAmount.equals("0")){

System.out.println("Order placed successfully...");

}

}

}

class Product{

private String productId;

private String productName;

private Integer productCost;

private String sellerID;

private Date postedDate;

private shoppingCart scart;

public Product(String pid, String pname, Integer pcost, String sid, Date pd, shoppingCart scart){

productId = pid;

productName = pname;

productCost = pcost;

sellerID = sid;

postedDate = pd;

this.scart = scart;

}

public void addToCart(String productName){

if(scart.addCartItem() && productName != null){

System.out.println("Item added to cart Successfully...");

}

}

public void sellProduct(String productId, String productName, int productCost){

if(productId != null && productName.length() > 0 && productCost > 0){

System.out.println("Product sold...");

}

else{

System.out.println("Product not found!!!");

}

}

private void getProductDetails(String pid, String pname, int pcost){

System.out.println("Product id : "+pid+", Product name : "+pname+", Product cost : "+pcost);

}

private void buyProduct(String sellerId, int pcost){

if(sellerId != null && pcost > 0){

System.out.println("Purchase successfull...");

}

else{

System.out.println("Could not buy item!!!");

}

}

}

class Seller extends User{

private String id, address, phone, email;

private int sellerRating;

public Seller(String i, String a, String p, String e, int sr){

this.id = i;

this.address = a;

this.phone = p;

this.email = e;

sellerRating = sr;

}

public String login(String mail, String num){

if(mail == email && num == phone){

return "Successful Login...";

}

return "Please try with proper credentials!!!";

}

public String register(String addr, String mail, String num){

if(addr != null && mail != null && num != null){

return "Account created successfully...";

}

return "Please enter a valid credential!!!";

}

public void updateProfile(String addr, String mail, String num){

address = addr;

email = mail;

phone = num;

}

}

class Payment{

private String id;

private String orderID;

private boolean paid;

private int total;

private String details;

private Orders o;

private Customer c;

private void sendOTP(){

if(c.phone != null){

System.out.println("OTP sent");

}

else{

System.out.println("Couldn't send OTP");

}

}

private String confirmTransaction(){

if(paid){

return "Transaction successfull";

}

return "Could not complete transaction";

}

private void getPaymentDetails(){

if(details != null){

System.out.println(details);

}

else{

System.out.println("No order to display");

}

}

private void makeTransaction(){

if(details != null && paid){

System.out.println("Transaction completed...");

}

else{

System.out.println("Could not proceed with transaction!!!");

}

}

}

public class blackrock {

public static void main(String[] args){

User u = new User(true, "testuser1", "12345");

Customer c = new Customer("testuser1", "westbengal", "123456789", "abc@bigcat.com");

System.out.println(c.login("abc@bigcat.com", "123456789"));

}

}