1. Regular attendance is essential to the Company’s efficient operation and is a necessary condition of employment.   
The objective behind attendance policy is to encourage attendance and to monitor employees regarding their punctuality .  
Office Timing is 09:00 AM - 6:00 PM.  
  
Write a Java program to monitor the employee check-in and check-out time.  
  
Create a class Attendance with following public static methods:  
  
a) void checkIn() - which displays "Welcome to office" if he/she is on or before 09:00 AM, else "You are late"  
b) void checkOut() - which displays "Thank you" if he/she logout on or after 06:00 PM, else "You are too early to logout"  
  
Create a class Office with main method, and invoke the Attendance class methods according to the user option.  
  
Refer sample input/output for better understanding:  
  
**Sample 1:**1. Check-In  
2. Check-Out  
1  
You Are Late  
  
**Sample 2:**1. Check-In  
2. Check-Out  
2  
You are too early to logout  
  
**Sample 3:**  
1. Check-In  
2. Check-Out  
1  
Welcome to Office  
  
-----------------------------------------  
Solution Code:  
-----------------------------------------  
public class Main {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        System.out.println("1. Check-In");  
        System.out.println("2. Check-Out");  
        int opt = sc.nextInt();  
  
        if (opt == 1) {  
            Attendance.checkIn();  
        } else if (opt == 2) {  
            Attendance.checkOut();  
        } else {  
            System.out.println("Invalid Option");  
        }  
    }  
}  
  
class Attendance {  
    public static void checkIn() {  
        Date currentTime = new Date();  
        Date inTime = new Date();  
        inTime.setHours(9);  
        inTime.setMinutes(0);  
        inTime.setSeconds(0);  
  
        if (currentTime.after(inTime)) {  
            System.out.println("You Are Late");  
        } else {  
            System.out.println("Welcome to Office");  
        }  
    }  
  
    public static void checkOut() {  
        Date currentTime = new Date();  
        Date outTime = new Date();  
        outTime.setHours(18);  
        outTime.setMinutes(0);  
        outTime.setSeconds(0);  
  
        if (currentTime.before(outTime)) {  
            System.out.println("You are too early to logout");  
        } else {  
            System.out.println("Thank You");  
        }  
    }  
}

2.Write a program to read a non-negative integer n, compute the sum of its digits. If sum is greater than 9 repeat the process and calculate the sum once again until the final sum comes to single digit.Return the single digit.  
  
Include a class UserMainCode with a static method getDigitSum which accepts the integer value. The return type is integer.  
  
Create a Class Main which would be used to accept the string and call the static method present in UserMainCode.  
  
**Input and Output Format:**

Input consists of a integer.  
Output consists of integer.

Refer sample output for formatting specifications.  
  
**Sample Input 1:**9999  
**Sample Output 1:**9  
  
**Sample Input 2:**698  
**Sample Output 2:**5  
  
-------------------------------------------------------------------------------  
Solution Code:  
-------------------------------------------------------------------------------  
import java.io.\*;  
import java.util.\*;  
  
public class Main {  
    public static void main(String[] args) {  
        int rem, sum = 0, dsum = 0, rem1;  
        Scanner sc = new Scanner(System.in);  
        int digit = sc.nextInt();  
        while (digit != 0) {  
            rem = digit % 10;  
            sum = sum + rem;  
            digit /= 10;  
        }  
        if (sum < 9) {  
            System.out.println(sum);  
        } else {  
  
            while (sum != 0) {  
                rem1 = sum % 10;  
                dsum += rem1;  
                sum /= 10;  
            }  
            System.out.println(dsum);  
        }  
    }  
}

3. Create a class TaxCalculator with the public static method calculate which accepts age as an integer value, income as a double value, and print the taxAmount (Note: refer sample output for result format).  
  
Calculate Tax Amount (tax\_percent\*income) as follows in the above said method  
  
taxpercent/Age below 60 60 to 80 80 and above  
0.0 <= 250000 <= 300000 <= 500000  
0.1 <= 500000 <= 500000   
0.2 <= 1000000 <= 1000000 <= 1000000  
0.3 >1000000 >1000000 >1000000  
  
Create a class TaxMain with main method, based on given sample input/output implement the main method and call the TaxCalculator's calculate method with the user value.  
  
**Sample 1:**Enter the Income:   
300000  
Enter the age  
51  
Tax Amount: 30000.00  
  
**Sample 2:**Enter the Income:   
300000  
Enter the age  
71  
Tax Amount: 0.00  
  
**Sample 3:**Enter the Income:   
800000  
Enter the age  
82  
Tax Amount: 160000.00  
  
-------------------------------------------------------------------------------  
Solution Code:  
-------------------------------------------------------------------------------  
public class TaxMain {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        System.out.println("Enter the Income: ");  
        double income = sc.nextDouble();  
        System.out.println("Enter the age");  
        int age = sc.nextInt();  
        TaxCalculator.calculate(age, income);  
    }  
}  
  
class TaxCalculator {  
    public static void calculate(int age, double income) {  
        double taxPercent = 0.0;  
  
        if (age < 60) {  
            if (income <= 250000) {  
                taxPercent = 0.0;  
            } else if (income <= 500000) {  
                taxPercent = 0.1;  
            } else if (income <= 1000000) {  
                taxPercent = 0.2;  
            } else {  
                taxPercent = 0.3;  
            }  
        } else if (age < 80) {  
            if (income <= 300000) {  
                taxPercent = 0.0;  
            } else if (income <= 500000) {  
                taxPercent = 0.1;  
            } else if (income <= 1000000) {  
                taxPercent = 0.2;  
            } else {  
                taxPercent = 0.3;  
            }  
        } else {  
            if (income <= 500000) {  
                taxPercent = 0.0;  
            } else if (income <= 1000000) {  
                taxPercent = 0.2;  
            } else {  
                taxPercent = 0.3;  
            }  
        }  
        System.out.printf("Tax Amount: %.2f", (income \* taxPercent));  
    }  
}  
  
  
4. A shop will give discount of 10% for the second product, if the cost of purchased quantity is more than 1000.  
Ask user to enter the first product ,say the price is morethan 1000  
enter the value for the second product, if the value of the first product is more than 1000 then give 10% discount for the second product  
  
Judge and print total cost for user.Note the following input and output  
  
Enter the first product price:1000  
Enter the second prodcut Price:500(50rs discount)  
Total cost of the product:1450  
  
Solution code:  
-----------  
import java.util.Scanner;  
class Ans{  
  public static void main(String[] args){  
    Scanner s = new Scanner(System.in);  
    System.out.println("Enter quantity");  
    int x = s.nextInt();  
    if((x\*100)>1000){  
      System.out.println("You get a discount of "+(.1\*x\*100)+" and your total cost is "+(x\*100-(.1\*x\*100)));  
    }  
    else{  
      System.out.println("No discount");  
    }  
  }  
}  
  
5. Create a Java program to manage Bank Account transactions like opening account, depositing amount, and withdraw.  
  
Use the following notes to create classes, its properties and methods.  
  
a) Create a class Account with following instance variables:  
-accountNumber:int  
-balance:double  
  
Include appropriate getters/setters, default constructor and a constructor with all properties.  
  
Include a method deposit which accepts "amount" as double, update the balance, and display updated balance.  
Include a method withdraw which accepts "amount" as double, update the balance, and display updated balance.  
  
Note:  
1) Account number should be exactly 5 digits, and shouldn't be all zeros.  
2) Opening balance should be Rs.1000 or above.  
3) depositing amount not less than Rs. 100  
4) while withdrawing, the balance should not go below Rs. 1000  
5) All properties are private, and methods & constructors are public.  
  
Implement validation 1 & 2 in constructors, validation 3 in method "deposit", and validation 4 in method "withdraw".  
  
b) Create a class Bank with main method, and implement the code using given sample input/output.  
  
**Sample 1:**Enter the account number  
00000  
Invalid Account Number  
  
**Sample 2:**Enter the account number  
25A21  
Invalid Account Number  
  
**Sample 3:**Enter the account number:   
47503  
Enter the opeing balance:   
500  
Insufficient Balance  
  
**Sample 4:**Enter the account number:   
47503  
Enter the opeing balance:   
1000  
Enter the amount to deposit:   
500  
Your Account Balance is: 1500.00  
Enter the amount to withdraw:   
200  
Your Account Balance is: 1300.00  
  
  
-------------------------------------------------------------------------------  
Solution Code - Bank Account Transaction  
-------------------------------------------------------------------------------  
public class Bank {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        System.out.println("Enter the account number: ");  
        String strAccNo = sc.nextLine();  
        int accountNumber=0;  
          
        if(strAccNo.matches("[0-9]{5}")){  
            accountNumber=Integer.parseInt(strAccNo);  
        }else{  
            System.out.println("Invalid Account Number");  
            System.exit(0);  
        }  
          
        if(accountNumber==0){  
            System.out.println("Invalid Account Number");  
            System.exit(0);  
        }  
          
        System.out.println("Enter the opeing balance: ");  
        double balance = Double.parseDouble(sc.nextLine());  
          
        if(balance<1000) {  
            System.out.println("Insufficient Balance");  
            System.exit(0);  
        }  
          
        Account acc1 = new Account(accountNumber, balance);  
          
        System.out.println("Enter the amount to deposit: ");  
        double amount = Double.parseDouble(sc.nextLine());  
        acc1.deposit(amount);  
          
        System.out.println("Enter the amount to withdraw: ");  
        amount = Double.parseDouble(sc.nextLine());  
        acc1.withdraw(amount);  
          
    }  
}  
  
class Account{  
    private int accountNumber;  
    private double balance;  
    public Account() {  
    }  
    public Account(int accountNumber, double balance) {  
        super();  
        this.accountNumber = accountNumber;  
        this.balance = balance;  
    }  
    public int getAccountNumber() {  
        return accountNumber;  
    }  
    public void setAccountNumber(int accountNumber) {  
        this.accountNumber = accountNumber;  
    }  
    public double getBalance() {  
        return balance;  
    }  
    public void setBalance(double balance) {  
        this.balance = balance;  
    }  
    public void deposit(double amount) {  
        if(amount<100) {  
            System.out.println("Insufficient amount to deposit");  
            System.exit(0);  
        }else {  
            balance+=amount;  
            System.out.printf("Your Account Balance is: %.2f\n",balance);  
        }  
          
    }  
    public void withdraw(double amount) {  
        if((balance-amount)<1000) {  
            System.out.println("Insufficient balance");  
            System.exit(0);  
        }else {  
            balance-=amount;  
            System.out.printf("Your Account Balance is: %.2f\n",balance);  
        }  
    }  
}