

Grade 10

INFORMATION TECHNOLOGY

MEMO FOR SAMPLE PRACTICAL PAPER

Time: 2 hours

Marks: 120

QUESTION ONE: DATABASE

1.1 Primary Key vv

| | | |
|-----------------|----------|---|
| EmpNumber | Text | 8 v |
| Name | Text | 30 v |
| Surname | Text | 30 v |
| Gender | Text | 1 v |
| AppointmentDate | Date | shortdate v |
| Salary | Currency | set to accept from R2000.00 to R19999.99v |

Default value of Gender field set to F v

Validation of Gender field to accept only F and M vv

Validation of salary field to accept values ≥ 2000 vv

Input mask for appointment date w [15]

1.1 Create a form all the fields v

The form is in columnar form v

Applies a style to the form v

Form Header v

Input mask for EmpNumberw v

Name in footer of formv

Layout and label of fields v

Save the form with EmpForm filenamev [10]

1.3 Enters data into databasew [2]

1.4 Query : Criteria for Gender set to F v

Criteria for salary set to < 2800 v

Criteria for appointment date set to Like("2005*") or ww

DatePart("yyy",[appointmentDate])=2005 or $>$ "2004/12/31" and \leq "2005/12/31"

Sort vvSaves query as Salv [8]

TOTA [35]

QUESTION 2 - PROGRAM

| | Possible mark | Mark Awarded | Moderated Mark |
|--------------------------------------|---------------|--------------|----------------|
| Read data and correct variables | 4 | | |
| If/Case structure and correct format | 7 | | |
| Rounding | 2 | | |
| Display (name, 2 values) | 2 | | |
| Total | 15 | | |

Possible Delphi solution



```

procedure TfrmComm.Button1Click(Sender: TObject);
var
  rSales, rComm, Comm :real;  ü
  sName : string[20];  ü
begin
  sName := edtName.Text;  ü
  rSales := StrToFloat(edtSales.Text);  ü
  if rSales <= 500 thenü
    Comm := 0.02;  ü
  if (rSales > 500) and (rSales < 5000) thenüü
    Comm := 0.05;
  if (rSales >= 5000) thenü
    Comm := 0.10;
  rComm := rSales * Comm;  ü
  redOutput.Lines.add('Name: ' + sName);  ü          üü
  redOutput.Lines.add('Sales Amount: ' + FloatToStrF(rSales, ffCurrency, 10, 2));  ü

```

```
redOutput.Lines.add('Commission Amount: ' + FloatToStrF(rComm, ffCurrency, 10, 2)); Ü
```

```
end;
```

Possible Java solution

```
// The "Commission" class.  
import java.io.*;  
import java.text.*;
```

```
public class Commission  
{
```

```
    public static void main (String [] args) throws Exception  
    {
```

```
        BufferedReader inKeyboard = new BufferedReader (new InputStreamReader  
                                                        (System.in));
```

```
        System.out.println ("Enter the name of the sales person");
```

```
        Ü
```

```
        String sName = inKeyboard.readLine ();Ü
```

```
        System.out.println ("Enter total sales of the person");
```

```
        double sales = Double.parseDouble (inKeyboard.readLine ());Ü
```

```
        Ü
```

```
        double commission;
```

```
        if (sales <= 500) Ü
```

```
        {
```

```
            commission = sales * 0.02; Ü
```

```
        }
```

```
        else if ((sales > 500) & (sales < 5000)) ÜÜ
```

```
        {
```

```
            commission = sales * 0.05; Ü
```

```
        }
```

```
        else
```

```
        {
```

```
            commission = sales * 0.08; Ü
```

```
        }
```

```
        // Method 1 of producing two decimal points
```

```
        DecimalFormat decfor = new DecimalFormat ("0.##"); ÜÜ
```

```
        System.out.println("Name :      " + sName); Ü
```

```
        System.out.println("Sales amount:    " + decfor.format (sales)); Ü
```

```
        System.out.println("Commission:      " + decfor.format (commission)); Ü
```

```
        System.exit(0);
```

```
    } // main method
```

```
} // Commission class
```

NOTE:

§ Wherever pupils have used a user-defined class to aid in put, allocate marks accordingly. Also check that the class has been imported.

§ Wherever pupils have used a user-defined class to aid out put, allocate marks accordingly. Also check that the class has been imported

QUESTION 3

| Marksheet: Question 3 | Possible Mark | Mark allocated | Moderated Mark |
|--|---------------|----------------|----------------|
| Variables | 9 | | |
| Initialise highpay, total | 4 | | |
| Read waiter's name | 2 | | |
| Loop: Java (until N is pressed) Delphi(Clear components, set Focus) | 4 | | |
| Read week/weekend, hours, overtime and tips | 8 | | |
| Calculate pay: Get rate (6) Calculate Pay (4) Total for waiters (2) Get Highest (5) | 17 | | |
| Display results of calculations Pay (4) Total (2) Most Paid waiter (2) User friendly labels/comments (2) | 10 | | |
| Calculate number of stars | 3 | | |
| Display number of stars For loop (3) In online (2), Stars (1), Message(1) | 7 | | |
| Descriptive variables used (3) Dent statements in – readability of code (2) Use comments (1) | 6 | | |
| Total | 70 | | |

Possible Delphi solution

```

var
    frmPayGrade: TfrmPayGrade;
    rMostPay      :real;
    sMostPayName  :string; // must be global

implementation

{$R *.dfm}

procedure TfrmPayGrade.Button1Click(Sender: TObject);
var
    sName, sWhen, sLine      :string;
    K, iNumber, iHours, iOvertime, iRate :integer;
    rIncome, rTips, rTotal, rPay      :real;
begin
    sName := edtName.Text;
    sWhen := edtWhen.Text;

```

```

iHours := StrToInt(edtHours.Text); ÜÜ
iOvertime := StrToInt(edtOvertime.text); ÜÜ
if (upcase(sWhen[1]) ÜÜ = 'W') Ü then
    iRate := 5; Ü
if (Uppcase(sWhen[1]) = 'E') thenÜ
    iRate := 8; Ü
rTips := StrToFloat(edtTips.Text); ÜÜ
rPay := rTipsÜ + (iHours * iRate) Ü + (iOvertime * (2 * iRate)) ÜÜ;
redOutput.Lines.Add(sName + #9 + FloatToStrF(rPay, ffCurrency, 10, 2));
rTotal := rTotal + rPay; ÜÜ
// Grading
iNumber := Trunc(rPay / 500); ÜÜ
sLine := '';
For K := 1 to iNumber doÜÜÜ
    sLine := sLine + '* '; ÜÜ
redOutput.Lines.Add('Grading: ' + sLine); ÜÜÜ

// Most pay
if rPay > rMostPay thenÜÜ
    beginÜ
        sMostPayName := sName; Ü
        rMostPay := rPay; Ü
    end;

    edtName.Clear;
    edtWhen.Clear;
    edtHours.Clear;
    edtTips.Clear;
    edtOvertime.Clear;
    edtName.SetFocus; Ü
end;

procedure TfrmPayGrade.FormActivate(Sender: TObject);
begin
    rMostPay := 0; ÜÜ
    rTotal := 0; ÜÜ
    redOutput.Clear;
    redGOutput.Paragraph.TabCount := 2;
    redOutput.Paragraph.Tab[0] := 80;
    redOutput.Paragraph.Tab[1] := 150;
end;

procedure TfrmPayGrade.btnMostPayClick(Sender: TObject);
begin
    redOutput.Lines.Add(''); ÜÜ
    redOutput.Lines.Add('Total amount: R ' + #9 + FloatToStrF(rTotal,
        ffCurrency, 10, 2));

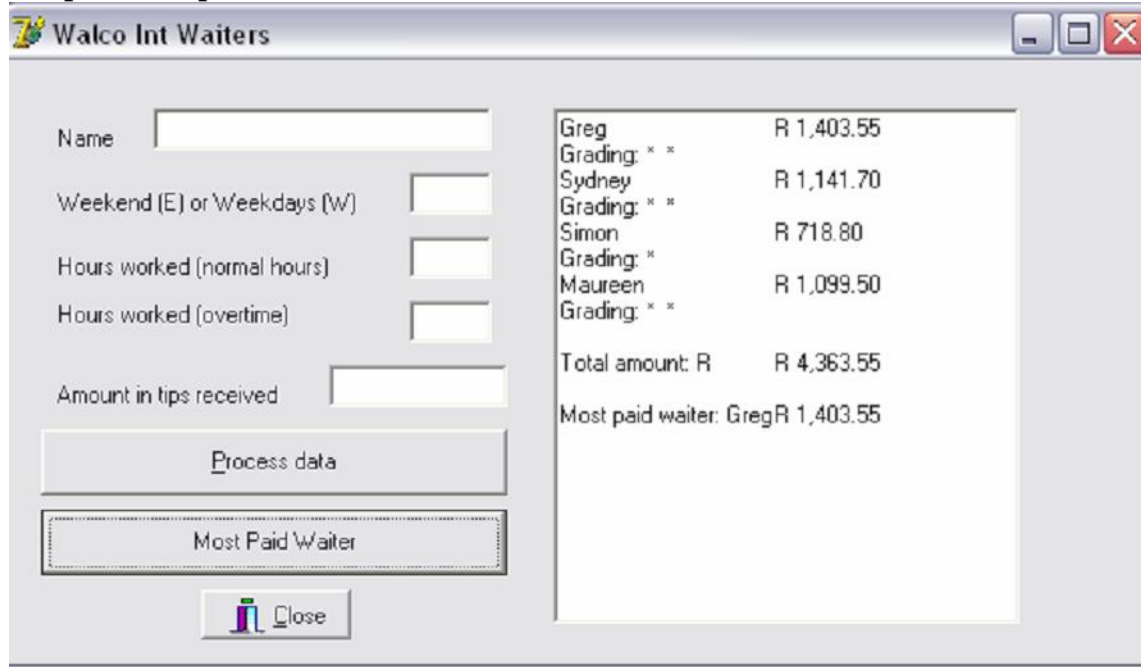
    redOutput.Lines.Add('');
    redOutput.Lines.Add('Most paid waiter: ' + sMostPayName + #9 +
        FloatToStrF(rMostPay, ffCurrency, 10, 2)); ÜÜ
end;

```

end.

Descriptive variable names ÜÜÜ
Dent in, readable structure: ÜÜ
Comments used: Ü

Output in Delphi:



Possible Java solution

```
import java.io.*;  
import java.text.*;  
public class Menu
```

```
{  
    static BufferedReader ink = new BufferedReader (new InputStreamReader  
                                                    (System.in));  
    static String option = "";           Ü Variable  
    public static void main (String [] args) throws Exception  
    {  
  
        String highestIncomeName = "";   Ü  
        double highestIncomePay = 0; ÜÜ Ü Variable  
        double totalIncome = 0; ÜÜ     Ü Variable  
        double pay = 0;                  Ü  
        double overtimePay = 0;         Ü
```

NOTE:

- § Wherever pupils have used a user-defined class to aid in put, allocate marks accordingly. Also check that the class has been imported.
- § Wherever pupils have used a user-defined class to aid out put, allocate marks accordingly. Also check that the class has been imported

```

String choice = "";

do
{
    // Enter the waiters' details
    // *****
    System.out.print ("Waiter's name : ");
    String name = ink.readLine ();
    System.out.print ("(W)eek or Week(E)nd : ");
    String Day = ink.readLine ();
    System.out.print ("Waiter's number of hours worked : ");
    int hours = Integer.parseInt (ink.readLine());
    System.out.print ("Waiter's number of overtime hours : ");
    int overtimeHours = Integer.parseInt (ink.readLine ());
    System.out.print ("Waiter's tips received : ");
    double tips = Double.parseDouble (ink.readLine ());

    // Determine the pay and overtime pay values
    // *****
    if (Day.toUpperCase ().equals ("W"))
    {
        pay = 5.0;
        overtimePay = 10.0;
    }
    else if (Day.toUpperCase ().equals ("E"))
    {
        pay = 8.0;
        overtimePay = 16.0;
    }

    // Calculate total pay
    // *****
    DecimalFormat decfor = new DecimalFormat ("0.00");
    double totalPay = (pay * hours) + (overtimePay * overtimeHours) + tips;
    System.out.println ("Waiter : " + name + " earned R " +
        decfor.format(totalPay));

    totalIncome = totalIncome + totalPay;

    // Determine if this is the waiter with the highest pay thus far
    // *****
    if (totalPay > highestIncomePay)
    {
        highestIncomeName = name;
        highestIncomePay = totalPay;
    }

    // Determine Grading
    // *****
    int Grading = (int) totalPay / 500;
    System.out.print ("Grading: ");

    for (int j = 0 ; j < Grading ; j++)
    {
        System.out.print ("*");
    }
}

```

```

    }
    System.out.println ();

    System.out.println ("Enter another waiter details? (Y/N)");
    choice = ink.readLine ();
}

while (choice.toUpperCase ().equals ("Y"));

// Print out the results of the calculations
// *****
DecimalFormat decfor = new DecimalFormat ("0.00");
System.out.println ("Waiter with the highest income: " + highestIncomeName + "
    with an amount of R " + decfor.format(highestIncomePay));
System.out.println ();
System.out.println ("Total income of all waiters: R " +
    decfor.format(totalIncome));
System.exit(0);
}

}    Descriptive variable names
    Dent in, readable structure:
    Comments used:

```

Part of the output in Java: Waiters Payment

```

Waiter's name : Simon
(W)eek or Week(E)nd : e
Waiter's number of hours worked : 56
Waiter's number of overtime hours : 12
Waiter's tips received : 78.80
Waiter : Simon earned R 718.80
Grading: *
Enter another waiter details? (Y/N)
y
Waiter's name : Maureen
(W)eek or Week(E)nd : e
Waiter's number of hours worked : 88
Waiter's number of overtime hours : 17
Waiter's tips received : 123.50
Waiter : Maureen earned R 1099.50
Grading: **
Enter another waiter details? (Y/N)
n
Waiter with the highest income : Greg with an amount of R 1403.55

Total income of all waiters: R 4363.55

```

=====