

2nd Pre-Board Examination Session 2009-10

Informatics Practices (065)

Answer Key [Set – 3]

Note

1. This question paper is divided into three sections.
2. All questions are compulsory.
3. Section – A consists of 30 marks.
4. Section – B and Section – C are of 20 marks each
5. Answer the questions after carefully reading the text.

Section - A

Q1. Answer the following questions:

- (a) **FLOSS:** Free/Libre/Open Source Software, the term most commonly used when talking about either Free (Libre) Software or Open Source Software. 2

PYTHON is freeware open-source cross-platform interpreted scripting language that can be used for many kinds of software development.

(1 mark each for each definition)

- (b) In ER Modeling ,an Entity is used to model a physical thing in question but in Object Modeling technique it is treated as Object. The collection of similar entities is termed as Entity Set but common collection of objects refers to a repository called Class. 2

(2 marks for the correct difference)

- (c) Relation is the term used to denote a Table. It is the other name given to Table. 2
Tuple is also called as record, it is the row of a table

In the following example the Relation is Dept and there are four tuples in it.

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

(1 mark each for each definition)

- (d) In Banking we can use Business Computing for logging and maintaining daily transactions and reporting requirements. 4

In Financial Accounting of small scale or large scale industries for entering transactions of financial nature and generation of Ledgers, Vouchers, P&L A/C, Balance Sheet etc.

(1 mark each for naming two areas)

(1 mark each for describing application)

- 2 (a) For comments we can use Single Quotation mark character (') . 2
For breaking a long line of code to the next line we use _(underscore) character.

For Example:

' This is a Comment in Visual Basic
MsgBox "Here the Line Breaks into" &_
"New Line in code window"

(1 mark each for both the parts of question)

- (b)
1. Identify the problem
 2. Propose Solutions
 3. Identify a suitable solution
 4. Write the Code
 5. Test the solution
 6. Deliver the solution

For Example

Task: Write a code segment to find out whether a number is Even or Odd

1. Identify the problem

The task is to check a number for Even or Odd

2. Propose Solutions

There is one Solution which requires checking of the number to be divisible by 2 if it is then is a Even number otherwise it is a Odd number

3. Identify a suitable solution

The proposed solution is ideal for this task

4. Write the Code

```
Dim N as Integer
N=13
IF N Mod 2 =0 Then
    MsgBox "Even Number"
Else
    MsgBox "Even Number"
End If
```

5. Test the solution
Test the above code for different values of N and then finally
6. Deliver the solution

(½mark each for all 6 steps)

(1 mark for a suitable example)

- (c) (i) Modular Programming

4

An approach of dividing a big programming task into small manageable program modules

- (ii) Object Oriented Programming

A programming technique used to program with Object Oriented Programming languages and follows object oriented programming paradigms.

- (iii) Event Driven Programming

In this type of programming the Events plays major role to execute a set of program code. Events are the actions such as Mouse Click, Double Click, Key Press etc.

- (iv) RAD

Rapid Application Development is the current programming approach in which a prototype (or sample model) of the software application is made much before its actual implementation to give a feel of application before starting working with the actual code behind the application.

(1 mark each for each brief explanation)

- 3 (a) SQL is called Structured Query Language and PL/SQL is called the 2
Programming Language in SQL.

SQL does not supports Decision Making and Looping constructs while PL/SQL does support all these features and also exploits benefits of SQL Commands in it.

(2 marks for stating any 2 correct differences)

- (b) Cursors are the work areas for SQL SELECT statements to fetch records from 2
a database table and work with each record on record-by-record basis.

Triggers supports event based execution of code statements in database. A trigger gets initiated on the events such as Updating of records, Deletion of records etc. and can be triggered on Statement Level or Row Level.

(2 marks for the correct difference)

- (c) A NULL is not equal to a NULL”, I agree with the statement as NULL is a value 2
that is not present and is a missing value, which cannot be compared to any other value unless it is substituted with some value. So Null is never equal to anything, not Even NULL.

(2 marks for stating the correct reason)

- (d) PL/SQL block is a set of PL/SQL statements, which can be executed to 4
perform some identifiable task

Following are the PL/SQL Block types with general syntax

Anonymous: PL / SQL Block without any name

Syntax for Declaring an Anonymous PL/SQL Block

```
[DECLARE]
BEGIN
--Statements
```

```
[EXCEPTION]
```

```
END;
```

Procedure: Sub programs, which can take arguments.

Syntax for Declaring a Procedure

```
CREATE OR REPLACE PROCEDURE proc_name
IS
BEGIN
    --Statements

[EXCEPTION]

END;
```

Function: A function is similar to a procedure and must return a value.

Syntax for Declaring a Function

```
CREATE OR REPLACE FUNCTION name
RETURN datatype
IS
BEGIN
    --Statements

RETURN value;

[EXCEPTION]

END;
```

(1 mark for correct definition)

(1 mark for naming all 3 types of PL/SQL blocks)

(2 marks for giving example of each type)

Section – B

- 4 (a) Private Sub DrawShape () 2
- shpPlay.Height =2000
shpPlay.Width=3000
shpPlay.Top=1500
shpPlay.Left=4000
shpShape.Visible =True
- End Sub
- (½mark for setting height and width of shape)*
- (½mark for setting Top)*
- (½mark for setting Left)*
- (½mark for displaying the shape)*
-
- (b) Private Sub DisableControl () 2
- cmdDrawShape.Enabled = True
cmdRectangle.Enabled =False
cmdSquare.Enabled =False
cmdOval.Enabled =False
cmdCircle.Enabled =False
cmdRoundedRectangle.Enabled =False
cmdRoundedSquare.Enabled =False
cmdClearShape.Enabled =False
ShpPlay.Visible =False
- End Sub
- (1 mark for fulfilling Requirement 1)*
- (1 mark for fulfilling Requirement 3)*

(c) Private Sub EnableControl ()
 cmdDrawShape.Enabled = False
 cmdRectangle.Enabled = True
 cmdSquare.Enabled = True
 cmdOval.Enabled = True
 cmdCircle.Enabled = True
 cmdRoundedRectangle.Enabled = True
 cmdRoundedSquare.Enabled = True
 cmdClearShape.Enabled = True
 ShpPlay.Visible = False

End Sub

(1 mark for enabling all command buttons except DrawShape)

(1 mark for disabling DrawShape command button)

(d) The following code statement is to be embedded in the click event of each of the 2
respective command button with the respective shape name (Such as
vbShapeRectangle, vbShapeSquare etc.)

For the Click event of Command Button cmdRectangle we will write the code
fragment in it.

shpPlay.Shape = vbShapeRectangle

shpPlay.Visible = True

(1 mark each for both the statements.)

(e) DisableControl

Or

```
cmdDrawShape.Enabled = True
cmdRectangle.Enabled =False
cmdSquare.Enabled =False
cmdOval.Enabled =False
cmdCircle.Enabled =False
cmdRoundedRectangle.Enabled =False
cmdRoundedSquare.Enabled =False
cmdClearShape.Enabled =False
ShpPlay.Visible =False
```

(½ mark for enabling DrawShape command button)

(½mark for disabling all other command buttons exc ept DrawShape)

(1 mark for not displaying any shape on form)

5 (a) **Output:**

2

10

16

18

16

½mark for each line of correct output

(b) PRIVATE SUB Command1_Click()

Dim P,C as integer

P =1

FOR C=1 TO 10

P = P+1

IF P = 5 THEN

P = 1

PRINT "P is equal to 5"

ELSE

PRINT "P is not equal to 5"

ENDIF

NEXT C

(½mark each for correction of any 4 errors(Both th e PRINTs to be treated as one correction)

(c) If Percentage >=60 AND Percentage < = 100 Then

txtcategory.Text ="Excellent"

Elseif Percentage >= 40 Then

txtcategory.Text ="Very Good"

Else

txtcategory.Text ="Need to work hard"

End If

(½ mark for IF statement)

(½ mark for correct ELSEIF statement)

(½ mark for ELSE statement)

(½ mark for END IF statement)

(d) Num=6
Temp=Num

```
FOR NUM = 6 TO 1 STEP -2  
Temp=Temp-1  
IF Temp MOD 2 = 0 THEN  
PRINT Temp," is Even"  
ELSE  
PRINT Temp," is Odd"  
END IF  
NEXT NUM
```

(½mark for correct FOR statement)

(½mark for using correct Step)

(½mark for using correct NEXT statement)

(½mark for the body of loop)

(e) **Output:**

Education@cat
4
Cat
21

(½mark for each line of correct output)

Section C

6 (a) **Output:**

ACCOUNTING
RESEARCH
SALES

1 mark for first line of output

1 mark each for second and third line of output.

(b) **Output:**

W

We

Wel

1 mark for first line of output

1 mark each for second and third line of output.

(c) 60

40

20

(1 mark for first line of output)

(1 mark each for second and third line of output.)

(d) CREATE OR REPLACE FUNCTION MySum (V_NumA IN NUMBER, V_NumB IN NUMBER)

RETURN NUMBER AS

V_Sum NUMBER;

BEGIN

 V_Sum := V_NumA + V_NumB;

 RETURN V_Sum;

END;

(1 mark for header of Function)

(1 mark for using correct arguments and their data type)

(1 mark for calculating sum)

(1 mark for Returning the sum)

7 (a) CREATE TABLE Customer (

 Cust_ID NUMBER(7) PRIMARY KEY,

 Cust_Name VARCHAR2 (30),

 Cust_Add1 VARCHAR2 (20),

 Cust_Add2 VARCHAR2 (30),

 Pin_Code NUMBER (6),

 Cust_Phone VARCHAR2 (10));

(1/2 mark for CREATE TABLE command)

(1 mark for fields with data types)

(1/2 mark for PRIMARY KEY constraint)

- (b) BEGIN 2
 UPDATE Customer Set Cust_Phone = '2' || Cust_Phone
 WHERE Length (Cust_Phone) =7;
END;

(1 mark for Update command)

(½mark for SET)

(½mark for WHERE clause)

- (c) BEGIN 2
UPDATE Customer
 Set Cust_Phone = SUBSTR (Cust_Phone ,2,LEN(Cust_Phone)-1)
 WHERE Cust_Phone Like '2%';
END;

(½ mark for Update command)

(½mark for SET)

(½mark for WHERE clause)

(½mark for using SUBSTR)

- (d) CREATE OR REPLACE TRIGGER TrigBeforeUpdateCustomer
BEFORE UPDATE ON CUSTOMER
BEGIN
 DBMS_OUTPUT.PUT_LINE ('Starting Update');
END;

and

```
CREATE OR REPLACE TRIGGER TrigAfterUpdateCustomer
AFTER UPDATE ON CUSTOMER
BEGIN
    DBMS_OUTPUT.PUT_LINE ('End of Update');
END;
```

4

(2 marks for each trigger)