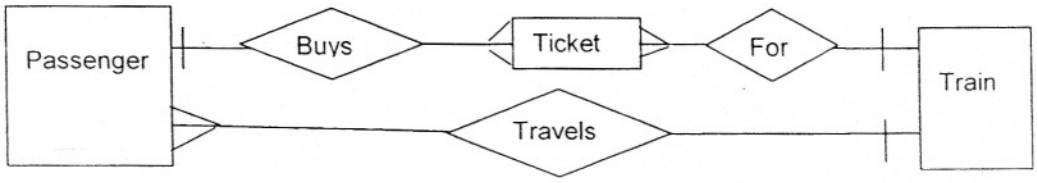
Important Note:

- **All answers provided in the marking scheme are SUGGESTIVE.**
- **Examiners are requested to accept all possible alternative correct answers.**
- **Case sensitivity to be ignored in Visual Basic for commands and identifiers.**
- **Semicolon termination and case sensitivity to be ignored in SQL statements.**
- **In case of answers for a SQL query, commands that produce identical output should be considered as correct.**

Section – A

Q1.	Answer the following questions		
	(a)	Expand the terms OSS and W3C.	2
	Ans:	Open Source Software World Wide Web Consortium	
		(1 Mark for each expansion)	
	(b)	What are the following software used for? (i) PHP (ii) MySQL	2
	Ans:	i) Hypertext Preprocessor is open source software used for server side scripting applications. ii) MySQL is multiuser, multithreaded open source RDBMS	
		(1 Mark for each correct usage) ***(½ Mark only for expanding the acronym PHP)***	
	(c)	Name any four application areas of databases.	2
	Ans:	Application areas: 1. Inventory Control 2. Financial Accounting 3. Pay-Accounting System 4. Invoice Management System 5. Personal Management System/HRD System 6. Fee Management system 7. Result Analysis System 8. Admission Management System 9. Income Tax Management System	
		(½ Mark each for mentioning any 4 correct application areas)	
	(d)	What are the different types of relationships that can be depicted through an ER model? Explain the concept of ER Model using the case study of a Ticket Reservation System that has three entities Ticket, Passenger and Train. Assume that each passenger can buy more than one ticket.	4
	Ans:	i) Many to Many relationship ii) Many to One relationship iii) One to One relationship iv) One to Many	



(1 Mark each for any 2 types of relationships)
(2 Marks for showing any two of three relationships)
NOTE : Relationship between entities can be represented by any name

Q2. Answer the following questions

(a) Differentiate between the FOR ... NEXT and FOR EACH ... NEXT loop of Visual Basic giving a suitable example of each. 2

Ans:

FOR.. NEXT is executed for a given number of times.

```
Example
FOR i=1 to 5
PRINT i
NEXT i
```

FOR EACH.. NEXT LOOP is executed for each element of a collection.

```
DIM i AS Variant
FOR EACH i IN Form1.Controls   OR Any collection can be used for example
MsgBox i.Name
NEXT
```

(1 Mark for any correct difference)
(1 Mark for any correct example)
OR
(2 Marks for any correct example with explanation)

(b) Name and explain the usage of any two types of modules available in Visual Basic. 2

Ans:
Form Module – contains code pertaining to form objects.
Standard Module – contains code pertaining to a project.
Class Module – contains code for user declared objects.

(1/2 Mark each for mentioning any two modules)
(1/2 Mark each for explaining their usage)

(c) What are data-aware controls? Name any two ActiveX data aware controls that can be used on a form 2

Ans:
A data aware control connects a control in Visual Basic to a database table.
ActiveX data-aware controls used on a form are:
i) DataList
ii) DataGrid
iii) DataCombo
iv) MSHflexgrid

(1 Mark for explanation)
(½ Mark each for any 2 ActiveX controls)

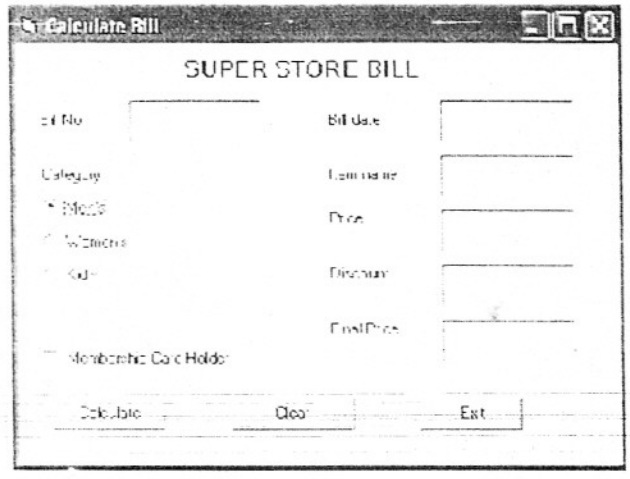
	(d)	Explain the term ADO object model. Differentiate between the Connection Timeout and Command Timeout properties of the Connection object.	4
	Ans:	ADO stands for ActiveX Data Objects. ADO enables us to write an application to access and manipulate data in a database server through an OLEDB/ADODC/ODBC provider. Connection timeout is the time specified after which the connection process is aborted. Command Timeout is the time specified after which the command execution is aborted.	
		(4 Marks for any answer bringing out the meaning of the ADO concept)	
Q3.	Answer the following questions		
	(a)	Differentiate between row-level and statement-level triggers.	2
	Ans:	Row level Triggers are fired / executed once for each row in a table. FOR EACH ROW clause is included in the Trigger Definition. Statement level Triggers are fired / executed once for a DML Statement. FOR EACH ROW clause is not included in the Trigger Definition.	
		(2 Marks for difference) *OR* *(2 Marks for giving one example of each)*	
	(b)	Define the terms Candidate key and Foreign key with respect to a database.	2
	Ans:	Candidate key is any attribute(s) which can serve as a Primary Key. Foreign Key is any non key attribute which references a primary key of some other table.	
		(1 Mark for each correct definition)	
	(c)	Differentiate between DDL and DML commands. Give one example of each type of command	2
	Ans:	DDL: Data Definition language commands create, alter or delete the structure of a table. Example: CREATE, ALTER, DROP DML: Data manipulation language commands manipulate and process data in a table. Example : INSERT, UPDATE, DELETE	
		(1 Mark for difference) *(1/2 Mark for giving one example of each)*	
	(d)	Mention any two advantages of PL/SQL as compared to SQL	2
	Ans:	1. Support for Procedural programming. . 2. Support for Exception handling capabilities. 3. Support for report writing.	
		(1 Mark each for mentioning any 2 advantages)	
	(e)	Name the keyword used to (i) Allow duplicate rows in a query. (ii) Avoid duplicate rows in a query.	2
	Ans:	(i) ALL (By default) / SELECT (ii) DISTINCT	

(1 Mark for each keyword)

SECTION – B

Q4. Read the following case study and answer the questions that follow:

Mr. Presi of Super Store decided to computerize the billing department. The accounts manager at Super Store is creating billing software to generate the bill during the sale period. A new bill is generated for each item sold. The discount is given based on the item category. An additional discount of 5% is given to the Store Membership Card holders. The following. is the data entry screen used at the store:



The list of controls for the above form is as follows:

Object Type	Object Name	Description
Form	FrmBill	The main form
Text Box	TxtBillno	To enter Bill Number
	TxtBillDate	To display System Date
	TxtIName	To input Item Name
	TxtPrice	To input Item Price
	TxtDisc	To display Discount
	TxtFinal	To display Final Price
Option Buttons	OptMens	To select Men's Category
	OptWomens	To Select Women's Category
	OptKids	To select Kid's Category
	ChkMember	To be checked for members
Command Button	CmdCalc	To calculate discount and Final Price
	CmdClear	To clear all textboxes
	CmdExit	To Exit from the application

(a) Write the code for the CmdClear command button to clear all the textboxes except the TxtBillDate textbox. 1

	Ans:	TxtBillno.Text=" " TxtIName.Text=" " .Text is optional TxtPrice.Text=" " TxtDisc .Text=" " instead of " " vbEmpty can be accepted TxtFinal.Text=" "	
		(1 Mark for clearing any 2 textboxes)	
	(b)	Write the code for the form load event of FrmBill so as to : (i) Display the system date in the TxtBillDate textbox. (ii) Disable the TxtBillDate, TxtDisc and TxtFinal textboxes	2
	Ans:	(i) TxtBillDate.text = Date OR TxtBillDate.text =Date() OR TxtBillDate.text = NOW OR TxtBillDate.text=NOW() (ii) TxtBillDate.ENABLED = FALSE TxtDisc.ENABLED = FALSE TxtFinal.ENABLED = FALSE	
		(1 Mark for displaying system date) *(1 Mark for disabling any 2 text boxes)*	
	(c)	Write the code for the change event of the TxtPrice textbox to ensure that the user does not enter a negative or a zero value. If a negative or a zero value is entered then the textbox should be made blank and a warning message should be displayed.	3

	Ans:	V=Val(TxtPrice.text)
		IF v <=0 then
		Msgbox" Re-enter a positive value greater than 0"
		TxtPrice.text=" "
		END IF	

		(1 Mark for conversion of txtPrice to numeric) *(1 Mark for checking the condition)* *(½ Mark for displaying the warning message)* *(½ Mark for clearing txtPrice)* *NOTE : If code is written under lost Focus or Validate event procedure full marks must be awarded.*	

(d) Write the code for the CmdCalc command button to display the discount and final price in the TxtDisc, and TxtFinal textboxes respectively. Note that Final price is calculated as price - discount and the discount is calculated based on the category and price according to the following table. Also remember to give an additional 5% discount for membership card holders i.e. if the ChkMember checkbox is checked. 4

Category	*Price*	*Discount*
Men's	<1000	30%
	>=1000	50%
Women's	<1500	40%
	>=1500	50%
Kid's	<500	20%
	>=500	30%

Ans.

```
Price=Val(TxtPrice.text)
IF OptMens.value=True THEN
      IF price<1000 THEN disc=0.3
            ELSE disc=0.5
       END IF
      ELSEIF OptWomens.Value=True THEN
            IF price<1500 THEN disc=0.4
            ELSE disc=0.5
            END IF
      ELSEIF OptKids.Value =True THEN
            IF price<500 THEN disc=0.2
            ELSE disc=0.3
            END IF
      END IF
      IF ChkMember.Value=1 THEN
            Disc=disc+0.05
      END IF
END IF
TxtDisc.text=disc
TxtFinal.text=price-price*disc
```

(1 Mark for converting TxtPrice into numeric)
(1 ½ Marks for computing discount)
(½ Mark for storing discount in TxtDisc textbox)
(1 Mark for storing amount in TxtFinal textbox)
NOTE : Any other equivalent logic must be accepted

Q5. Answer the following questions

(a) Find the errors from the following code segment and rewrite the corrected code underlining the correction made. 2

```
PRIVATE FUNCTION IsPrime (num As Integer) As Bool
```

```
DIM limit As Integer
limit = num / 2
FOR ctr = 2 TO limit
      IF num Modulo ctr = 0 THEN
            EXIT FOR LOOP
      END IF
NEXT
IsPrime = IIf(ctr > limit, True, False)
End Sub
```

Ans:

```
PRIVATE FUNCTION IsPrime (num As Integer) As Boolean
DIM limit As Integer
limit = num / 2
FOR ctr = 2 TO limit
      IF num Mod ctr = 0 THEN
            EXIT FOR
      END IF
NEXT
IsPrime = IIf(ctr > limit, True, False)
END FUNCTION
```

(½ Mark for identifying and correcting each error)
(1 Mark for ONLY identifying all errors)
NOTE: If ctr is identified as undeclared variable, it can be counted as an error.

(b) Find the output of the following code: 2

```
DIM astr as String
DIM I as Integer
I = 1
astr = "Come"
DO WHILE I <= Len (astr)
        Print Mid(astr, I)
I = I + 1
LOOP
```

Ans

```
Come
ome
me
e
```

(1/2 Mark for each correct line of output)

(c) Rewrite the following code using If Then Else construct without affecting the output: 2

```
DIM a As Integer
a = 1
SELECT Case a
      Case 1
            Print "Sunday"
      Case 2 To 6
            Print "WeekDay"
      Case 7
            Print "Nearing Weekend"
END SELECT
```

Ans:

```
DIM a As Integer
a = 1
IF a = 1 THEN
```

```
            Print "Sunday"
ELSELF a>=2 and a<=6 THEN

            Print "WeekDay"
ELSEIF a = 7 THEN
            Print "Nearing Weekend"
END IF
```

(½ Mark each for checking a=1 and a=7)
(1 Mark for checking a>=2 and a<=6)

(d) Write a Visual Basic procedure that takes a number as argument and displays the sum of all the digits in the number. For example, if the argument passed is 354, the procedure should display 12 (i.e. 3+5+4). 4

Ans:

```
PRIVATE SUB SumDigits(ByVal n As Integer)

OR Public Sub SumDigits( n As Integer)

DIM i, sum  As Integer
sum = 0
DO WHILE n > 0
      i = n Mod 10
      sum = sum + i
      n = n \ 10
LOOP
MsgBox "sum=" & sum
END SUB
```

```
PRIVATE SUB SumDigits(ByVal n As Irteger)

OR Public Sub SumDigits( n As Integer)

DIM i, sum  As Integer
sum = 0
For I=1 to LEN(Str(n))
     sum=sum+Val(Mid(Str(n),I,1))
NEXT

MsgBox "sum=" & sum
END SUB
```

(1 Mark for initializing sum variable)
(1 Mark for loop)
(1 Mark for summation of digits)
(1 Mark for displaying sum)
NOTE : Any other equivalent logic must be accepted

Section – C

Q6. Read the questions given below and answer accordingly :

a) Write the output produced by the following PL/SQL code: 2

```
DECLARE
      A NUMBER;
      B NUMBER;
      TEMP NUMBER;
BEGIN
      FOR X IN 1.. 4 LOOP
            TEMP : = A;
            A      : = B;
            B    : = TEMP;
            A    : A + 1;
            B   : B - 1;
```

```
            DBMS_OUTPUT.PUT_LINE('A = ' || A);
            DBMS_OUTPUT.PUT_LINE('B  = ' || B);
        END LOOP;
END;
```

(2 Marks for attempting or identifying that variables have not been initialized)
OR
(2 Marks to be awarded if either Q6.(b) or (c) or (d) is answered correctly)

b) Find the errors from the following PL/SQL code and rewrite the corrected code underlining the correction made — 2

```
DECLARATION
    V_ MNO MOVIES.MOVIENO%TYPE;
    V_TITLE MOVIES.TITLE%TYPE;
    V_ PRICE MOVIES.PRICE%TYPE;
BEGIN
    V_MNO EQUALS 101
    LOOP
    SELECT TITLE, PRICE, RATING INTO V_TITLE, V_PRICE, V_RATE
    FROM MOVIES
    WHERE MOVIENO = V_MNO;
    DBMS_OUTPUT.PUTTEXT (V_TITLE || ' '|| V_PRICE);
    EXIT WHEN V_RATE < 4;
    V_MNO : = V_MNO + 1;
   LOOP END;
END;
```

Ans:

```
DECLARE
     V_MNO MOVIES. MOVIENO%TYPE;
     V_TITLE MOVIES.TITLE%TYPE;
     V_PRICE MOVIES.PRICE%TYPE;
     V_RATE MOVIES.RATING%TYPE;
BEGIN
     V_MNO := 101;      OR DEFAULT CAN BE USED
     LOOP
            SELECT TITLE, PRICE, RATING INTO  V_TITLE, V_PRICE, V_RATE
            FROM MOVIES
            WHERE MOVIENO = V_MNO;
            DBMS_OUTPUT.PUT_LINE(V_TITLE||' '||V_PRICE);
            EXIT WHEN V_RATE<4;
            V_MNO: = V_MNO + 1;
     END LOOP;

END;
```

(½ Mark for identifying and correcting each error)
(1 Mark for ONLY identifying all errors)

c) Differentiate between the IN and IN OUT modes of a parameter in a PL/SQL block. — 2

Ans:

1. IN mode is used to receive actual argument value into the formal argument.
2. IN OUT mode is used to receive and send formal argument value to the actual argument in the calling environment/block.

(2 Marks for correct difference)
OR
(2 Marks for correct explanation of difference with the help of example)

d) Write a PL/SQL Function POWER that takes two numbers as arguments and returns the value of the first number raised to the power of the second. 4

Ans:

```
CREATE OR REPLACE FUNCTION Power( n IN Number,m IN number)
RETURN number as/is
BEGIN
      Return n ** m;          OR Power function can also be used
END;
```

(2 Marks for Function header)
(1½ Marks for Logic to compute power)
(½ Mark for returning the power value)
Note: Since POWER is a predefined function in PL/SQL , accept as correct a change in name of the function

Q7. Answer the questions based on the table CompLab given below:

Table: CompLab

Column Name	Data Type	Size	Constraint	Description
It_Code	NUMBER	4	PRIMARY KEY	Item Code
It_Cat	CHAR	1	'H' or 'S'	Item category as Hardware or Software .
It_Name	VARCHAR2	25	NOT NULL	Name of the item
It_Cost	NUMBER	8,2		Cost of each unit of an item
It_Qty	NUMBER	3		Item Quantity in the Lab
Dt_Pur	Date			Date of Purchases

(a) Write the SQL command to create the table CompLab including the constraints. 2

Ans:

```
CREATE TABLE CompLab
(
 It_Code NUMBER(2)       PRIMARY KEY,  OR NOT NULL UNIQUE
 It_Cat CHAR CHECK(It_Cat = 'H' OR It_Cat = 'S'),OR CHECK(It_Cat IN('H','S'))
 It_Name     VARCHAR2(30) NOT NULL,
 It_Cost     NUMBER(8, 2),
 It_Qty      NUMBER(3),
 Dt_Pur      Date
);
```

(½ Mark for CREATE TABLE table name)
(½ Mark for fields with data types)
(½ Mark for PRIMARY KEY and NOT NULL constraint)
(½ Mark for CHECK constraint)

(b) Write the SQL command to display the details of the item with the maximum It_Cost. 2

	Ans:	`SELECT * FROM CompLab` `WHERE It_Cost=(SELECT MAX(It_Cost) FROM CompLab);`	
		(1 Mark for outer query) *(1 Mark for sub query)*	
	(c)	Write the PL/SQL code to increase the item cost by 10% for an item code accepted from the user if the date of purchase of the item is later than 12-0ct-2005.	3

	Ans:	`CREATE OR REPLACE PROCEDURE INC_COST(p_It_Code IN NUMBER)`
		`IS/AS`
		`  V_Cost CompLab.It_Cost%TYPE;`
		`BEGIN`
		`  UPDATE CompLab`
		`  SET It_Cost = IT_Cost + 0.1*IT_Cost`
		`  WHERE It_Code = p_It_Code AND Dt_Pur > '12-0ct-2005';`
		`END;`	

		(1 Mark for Header) *(1 Mark for Update)* *(1 Mark for Where Clause)* Note : *Value for Item code to be updated can be accepted within the procedural block* *Any other equivalent code must be accepted*	
	(d)	Write the PL/SQL code to create a stored procedure Disp_Details to display the details of all the items with It_Cat as 'H'. The code should also display the total quantity of all such items.	4

	Ans:	`CREATE OR REPLACE PROCEDURE Disp_Details`
		`AS/IS`
		`CURSOR C1 IS`
		`  SELECT * FROM CompLab`
		`  WHERE It_Cat = 'H';`
		
		`Rec  c1%ROWTYPE;`
		`Tqty Number(5) :=0;`
		`BEGIN`
		`OPEn C1;`
		`LOOP`
		`  FETCH C1 INTO Rec;`
		`  EXIT WHEN C1%NOTFOUND;`
		`  Tqty :=Tqty+ Rec.It_Qty;`
		`  DBMS_OUTPUT.PUT_LINE( Rec. It_Code || ' '|| Rec. It_Name || ' '`
		`  ||Rec. It_Qty);`
		`END LOOP;`
		`CLOSE C1;`
		`DBMS_OUTPUT.PUT_LINE(Tqty);`
		`END;`	

		(1/2 Mark for procedure header) *(1 Mark for Cursor Declaration)* *(2 Marks for Cursor Processing Loop)* *(1/2 Mark for displaying total quantity)* *NOTE : Any other form of Loop for processing cursor must be accepted*	