

**12<sup>th</sup>**  
**STD**

**HALF YEARLY EXAMINATION - DEC. 2024 - 25**

Reg. No.

--	--	--	--	--	--	--	--

**COMPUTER SCIENCE (with Answers)**

**TIME ALLOWED : 3.00 Hours]**

**[MAXIMUM MARKS : 70**

**PART - I**

- I. Answer all the questions. (15×1=15)**
- The small sections of code that are used to perform a particular task is called:
    - Subroutines
    - Files
    - Pseudo code
    - Modules
  - Which of the following functions that retrieve information from the data type?
    - Constructors
    - Selectors
    - Recursive
    - Nested
  - Which of the following security technique that regulates who can use resources in a Computing environment.
    - Password
    - Authentications
    - Access control
    - Certification
  - Binary search is also called as
    - Linear search
    - Sequential search
    - Random search
    - Half-Interval search
  - Which of the following is not Logical operator?
    - and
    - or
    - not
    - assignment
  - The condition in the if statement should be in the form of
    - Arithmetic or Relational expression
    - Arithmetic or Logical expression
    - Relational or Logical expression
    - Arithmetic
  - Which of the following keywords is used to begin the function block?
    - Define
    - for
    - finally
    - def
  - Defining strings with in triple quotes allows creating :
    - Single line strings
    - Multiline strings
    - Double line strings
    - Multiple strings
  - The keys in Python dictionary is specified by
    - = (equal)
    - ; (semicolon)
    - + (plus)
    - : (colon)
  - The following is the private class variable
    - \_\_num
    - &&num
    - ##num
    - \$\$num
  - Which database model represents parent-child relationship?
    - Relational
    - Network
    - Hierarchical
    - Object

- The clause used to sort data in a database
  - SORT BY
  - ORDER BY
  - GROUP BY
  - SELECT
- A CSV file is also known as a
  - Flat File
  - 3D File
  - String File
  - Random File
- The following is not a Scripting language
  - Java Script
  - PHP
  - Perl
  - HTML
- Any change made in the values of the record should be saved by the command
  - Save
  - Save as
  - Commit
  - Oblige

**PART - II**

- II. Answer any 6 questions and question no. 23 is compulsory 6 × 2 = 12**
- What is space time trade off?
  - What are keywords? Give example.
  - Write a note on 'continue' statement in Python.
  - Write about replace() function in Python.
  - Write a python program using class to accept three sides of a triangle and print its area.
  - List and explain the components of a database.
  - What is the usage of IN keyword in SQL?
  - Explain fetchone() method with an example program.
  - List some commonly used interfaces for wrapping.

**PART - III**

- III. Answer any 6 questions and question no.30 is compulsory. 6 × 3 = 18**
- Mention the characteristics of interface.
  - Write note on Asymptotic notation.
  - Write short notes on Arithmetic operators with examples.
  - Write note on if..else structure.
  - How recursive function works?
  - What are the differences between List and Tuple?
  - Explain Cartesian Product with a suitable example.
  - Write any three DDL commands.
  - What is MinGW? What is Its use?

**PART - IV**

- IV. Answer all the questions. 5 × 5 = 25**
- (a) Explain with example Pure and impure functions.

(OR)

[1]

- (b) Explain the types of scopes for variable or LEGB rule with example.
35. (a) Discuss about Linear search algorithm  
(OR)
- (b) Write a detail note on 'for' loop.
36. (a) Explain about string operators in Python with suitable example. (OR)
- (b) Explain the different set operations supported by python with suitable example.
37. (a) Explain the different types of data model.  
(OR)
- (b) Write the different types of constraints and their functions.
38. (a) Tabulate the different modes in CSV file with its meaning. (OR)
- (b) What is the purpose of sys, os, getopt modules in python? Explain.



**ANSWER**

**PART - I**

1. (a) Subroutines
2. (b) Selectors
3. (c) Access control
4. (d) Half-Interval search
5. (d) assignment
6. (c) Relational or Logical expression
7. (d) def
8. (b) Multiline strings
9. (d) : (colon)
10. (a) \_\_num
11. (c) Hierarchical
12. (b) ORDER BY
13. (a) Flat File
14. (d) HTML
15. (c) Commit

**PART - II**

16. A space-time or time-memory trade off is a way of solving in less time by using more storage space or by solving a given algorithm in very little space by spending more time.
17. Keywords are special words that are used by Python interpreter to recognize the structure of program. **Example:** if, void, class.
18. Continue statement unlike the break statement is used to skip the remaining part of a loop and start with next iteration.
19. replace () is used to eliminate the newline character at the end of each row with open.

**Syntax :**

replace ("char 1", "char 2").

20. Class Tr:
- ```
def __init__(self, a, b, c):
    self.a = float(a)
    self.b = float(b)
    self.c = float(c)
def area(self):
    s = (self.a + self.b + self.c)/2
    return((s*(S.self.a)*(s.self.b)*(s.self.c)**0.5)
a = input("Enter side 1 :")
b = input("Enter side 2 :")
c = input("Enter side 3 :")
ans = Tr(a,b,c)
print(ans.area())
```

**Output :**

```
Enter side 1 : 3
Enter side 2 : 4
Enter side 3 : 5
6.0
```

21. The Database Management System can be divided into five major components as follows:

1. Hardware
2. Software
3. Data
4. Procedures/Methods
5. Database Access Languages

22. **IN Keyword :** The IN keyword is used to specify a list of values which must be matched with the record values. In other words it is used to compare a column with more than one value. It is similar to an OR condition.

23. The fetchone() method returns the next row of a query result set or None in case there is no row left.

**Example :**

```
import sqlite3
connection = sqlite3.connect("Academy.db")
cursor = connection.cursor()
cursor.execute("SELECT * FROM student")
print("\nfetch one:")
res = cursor.fetchone()
print(res)
```

**OUTPUT**

```
fetch one:
(1, 'Akshay', 'B', 'M', 87.8, '2001-12-12')
```

24. ■ Python-C-API (API-Application Programming Interface for interfacing with C programs)
- Ctypes (for interfacing with c programs)
- SWIG (Simplified Wrapper Interface Generator- Both C and C++)
- Cython (Cython is both a Python-like language for writing C-extensions)
- Boost. Python (a framework for interfacing Python and C++)
- MinGW (Minimalist GNU for Windows)

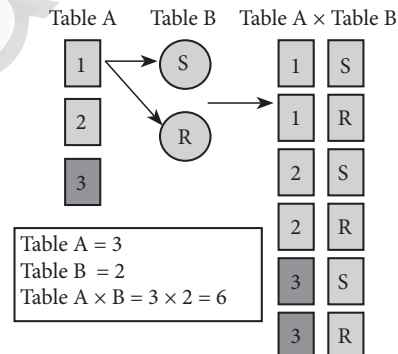
## PART - III

25. (i) The class template specifies the interfaces to enable an object to be created and operated properly.  
(ii) An object's attributes and behaviour is controlled by sending functions to the object.
26. Asymptotic Notations are languages that uses meaningful statements about time and space complexity. The following three asymptotic notations are mostly used to represent time complexity of algorithms:  
(i) **Big O** : Big O is often used to describe the worst-case of an algorithm.  
(ii) **Big  $\Omega$**  : Big Omega is the reverse Big O, if Big O is used to describe the upper bound (worst - case) of a asymptotic function, Big Omega is used to describe the lower bound (best-case).  
(iii) **Big  $\Theta$**  : When an algorithm has a complexity with lower bound = upper bound, say that an algorithm has a complexity  $O(n \log n)$  and  $\Omega(n \log n)$ , it's actually has the complexity  $\Theta(n \log n)$ , which means the running time of that algorithm always falls in  $n \log n$  in the best-case and worst-case.
27. (i) An arithmetic operator is a mathematical operator that takes two operands and performs a calculation on them. They are used for simple arithmetic.  
(ii) Most computer languages contain a set of such operators that can be used within equations to perform different types of sequential calculations.  
(iii) Python supports the following Arithmetic operators.

| Operator - Operation                                      | Examples                         | Result |
|-----------------------------------------------------------|----------------------------------|--------|
| Assume a=100 and b=10. Evaluate the following expressions |                                  |        |
| + (Addition)                                              | >>> a + b                        | 110    |
| - (Subtraction)                                           | >>> a - b                        | 90     |
| * (Multiplication)                                        | >>> a * b                        | 1000   |
| / (Division)                                              | >>> a / b                        | 10.0   |
| % (Modulus)                                               | >>> a % 30                       | 10     |
| ** (Exponent)                                             | >>> a ** 2                       | 10000  |
| // (Floor Division)                                       | >>> a //30<br>(Integer Division) | 3      |

28. (i) The if.. else statement provides control to check the true block as well as the false block.  
(ii) **Syntax:**  
*if <condition>:*  
*statements-block 1*  
*else:*  
*statements-block 2*

- (iii) if..else statement thus provides two possibilities and the condition determines which BLOCK is to be executed.
29. (i) Recursive function is called by some external code.  
(ii) If the base condition is met then the program gives meaningful output and exits.  
(iii) Otherwise, function does some required processing and then calls itself to continue recursion.
30. (i) The elements of a list are changeable (mutable) whereas the elements of a tuple are unchangeable (immutable), this is the key difference between tuples and list.  
(ii) The elements of a list are enclosed within square brackets. But, the elements of a tuple are enclosed by parenthesis.  
(iii) Iterating tuples is faster than list.
31. (i) Cross product is a way of combining two relations. The resulting relation contains, both relations being combined.  
(ii)  $A \times B$  means A times B, where the relation A and B have different attributes.  
(iii) This type of operation is helpful to merge columns from two relations.



Cartesian Product

32. **Data Definition Language :**  
(i) **Create Command :** To create tables in the database.  
CREATE TABLE Student  
(Admno integer,  
Name char(20),  
Gender char(1),  
Age integer,  
Place char(10),  
);  
(ii) **Alter Command :** The ALTER command is used to alter the table structure like adding a column, renaming the existing column, change the data type of any column or size of the column or delete the column from the table.

**ALTER TABLE <table-name> ADD**  
 <column-name><data type><size>;  
 To add a new column "Address" of type 'char' to the Student table, the command is used as

**ALTER TABLE Student ADD Address char;**  
**(iii) Drop Command :** The DROP TABLE command is used to remove a table from the database.

The table can be deleted by DROP TABLE command in the following way:  
 DROP TABLE table-name;

33. (i) MinGW refers to a set of runtime header files, used in compiling and linking the code of C, C++ and FORTRAN to be run on Windows Operating System.
- (ii) MinGw-W64 (version of MinGW) is the best compiler for C++ on Windows. To compile and execute the C++ program, you need 'g++' for Windows. MinGW allows to compile and execute C++ program dynamically through Python program using g++.
- (iii) Python program that contains the C++ coding can be executed through either by using command prompt or by using run terminal.

#### PART - IV

#### 34. (a) Pure functions :

- (i) Pure functions are functions which will give exact result when the same arguments are passed.
- (ii) For example the mathematical function sin (0) always results 0. This means that every time you call the function with the same arguments, you will always get the same result.
- (iii) A function can be a pure function provided it should not have any external variable which will alter the behaviour of that variable.

Let us see an example  
 let square x :=  
 return: x \* x

- (iv) The above function square is a pure function because it will not give different results for same input.
- (v) There are various theoretical advantages of having pure functions. One advantage is that if a function is pure, then if it is called several times with the same arguments, the compiler only needs to actually call the function once.

#### Example :

```
let length s:=
  i: = 0
  let i: = 0;
  if i < strlen (s) then
  -- Do something which doesn't affect s
  ++i
```

(vi) If it is compiled, strlen (s) is called each time and strlen needs to iterate over the whole of 's'. If the compiler is smart enough to work out that strlen is a pure function and that 's' is not updated in the loop, then it can remove the redundant extra calls to strlen and make the loop to execute only one time.

(vii) From these what we can understand, strlen is a pure function because the function takes one variable as a parameter, and accesses it to find its length. This function reads external memory but does not change it, and the value returned derives from the external memory accessed.

#### Impure functions :

- (i) The variables used inside the function may cause side effects though the functions which are not passed with any arguments. In such cases the function is called impure function.
- (ii) When a function depends on variables or functions outside of its definition block, you can never be sure that the function will behave the same every time it's called. For example the mathematical function random() will give different outputs for the same function call.

```
let randomnumber:=
  a := random()
  if a > 10 then
  return: a
else
  return: 10
```

- (iii) Here the function Random is impure as it is not sure what will be the result when we call the function.

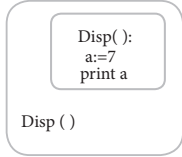
(OR)

- (b) **Types of Variable Scope :** There are 4 types of Variable Scope, let's discuss them one by one:

#### Local Scope :

- (i) Local scope refers to variables defined in current function. Always, a function will first look up for a variable name in its local scope. Only if it does not find it there, the outer scopes are checked.

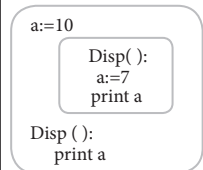
Look at this example

| 1. Disp():<br>2. a:=7<br>3. print a<br>4. Disp() | Entire program<br> | Output of the Program<br>7 |
|--------------------------------------------------|---------------------------------------------------------------------------------------------------------|----------------------------|
|--------------------------------------------------|---------------------------------------------------------------------------------------------------------|----------------------------|

- (ii) On execution of the above code the variable **a** displays the value 7, because it is defined and available in the local scope.

**Global Scope:**

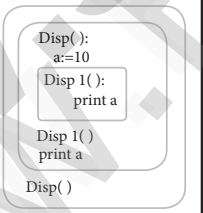
- (i) A variable which is declared outside of all the functions in a program is known as global variable.
- (ii) This means, global variable can be accessed inside or outside of all the functions in a program. Consider the following example.

|                                                                            |                                                                                                     |                                  |
|----------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------|----------------------------------|
| 1. a:=10<br>2. Disp():<br>3. a:=7<br>4. print a<br>5. Disp()<br>6. print a | Entire program<br> | Output of the Program<br>7<br>10 |
|----------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------|----------------------------------|

- (iii) On execution of the above code the variable 'a' which is defined inside the function displays the value 7 for the function call Disp() and then it displays 10, because a is defined in global scope.

**Enclosed Scope :**

- (i) All programming languages permit functions to be nested. A function (method) with in another function is called nested function.
- (ii) A variable which is declared inside a function which contains another function definition with in it, the inner function can also access the variable of the outer function. This scope is called enclosed scope.
- (iii) When a compiler or interpreter search for a variable in a program, it first search Local, and then search Enclosing scopes. Consider the following example.

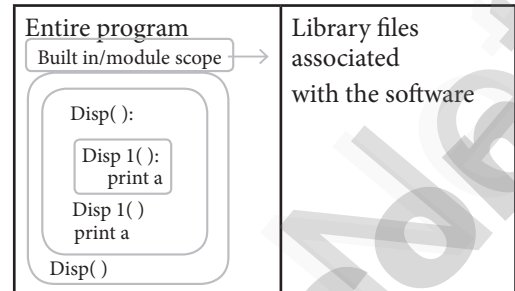
|                                                                                              |                                                                                                       |                                   |
|----------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------|-----------------------------------|
| 1. Disp():<br>2. a:=10<br>3. Disp1():<br>4. print a<br>5. Disp1()<br>6. print a<br>7. Disp() | Entire program<br> | Output of the Program<br>10<br>10 |
|----------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------|-----------------------------------|

- (iv) In the above example Disp1() is defined with in Disp(). The variable 'a' defined in Disp() can be even used by Disp1() because it is also a member of Disp().

**Built-in Scope :**

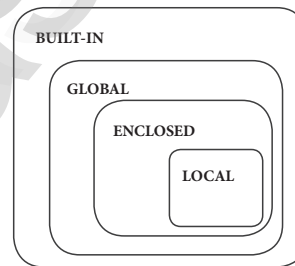
- (i) The built-in scope has all the names that are pre-loaded into the program scope when we start the compiler or interpreter.
- (ii) Any variable or function which is defined in the modules of a programming language

has Built-in or module scope. Consider the following example.



**LEGB rule :** The LEGB rule is used to decide the order in which the scopes are to be searched for scope resolution. The scopes are listed below in terms of hierarchy (highest to lowest).

|             |                                                              |
|-------------|--------------------------------------------------------------|
| Local(L)    | Defined inside function/class                                |
| Enclosed(E) | Defined inside enclosing functions (Nested function concept) |
| Global(G)   | Defined at the uppermost level                               |
| Built-in(B) | Reserved names in built-in functions (modules)               |



35. (a) (i) Linear search also called sequential search is a sequential method for finding a particular value in a list.
- (ii) This method checks the search element with each element in sequence until the desired element is found or the list is exhausted. In this searching algorithm, list need not be ordered.

**Procedure :**

- Traverse the array using for loop
- In every iteration, compare the target search key value with the current value of the list.
  - If the values match, display the current index and value of the array
  - If the values do not match, move on to the next array element.
- If no match is found, display the search element not found.
  - To search the number 25 in the array given below, linear search will go step

by step in a sequential order starting from the first element in the given array if the search element is found that index is returned otherwise the search is continued till the last index of the array. In this example number 25 is found at index number 3.

| Index  | 0  | 1  | 2  | 3  | 4  |
|--------|----|----|----|----|----|
| Values | 10 | 12 | 20 | 25 | 30 |

**Example 1 :**

Input: values[] = {5, 34, 65, 12, 77, 35}

target = 77

**Output:** 4

**Example 2:**

Input: values[] = {101, 392, 1, 54, 32, 22, 90, 93}

target = 200

**Output:** -1 (not found)

(OR)

- (b) (i) **for loop :** The for loop is usually known as a definite loop, because the programmer knows exactly how many times the loop will be executed.

**(ii) Syntax:**

```
for counter_variable in sequence:
    statements-block 1
[else: # optional block
    statements-block 2]
```

- (iii) The for .... in statement is a looping statement used in Python to iterate over a sequence of objects, i.e., it goes through each item in a sequence. Here the sequence is the collection of ordered or unordered values or even a string.

- (iv) **range ():** The range() is a built-in function, to generate series of values between two numeric intervals.

- (v) The **syntax** of range() is as follows:

```
range (start,stop,[step])
```

Where,

start – refers to the initial value

stop – refers to the final value

step – refers to increment value, this is optional part.

**Examples for range() :**

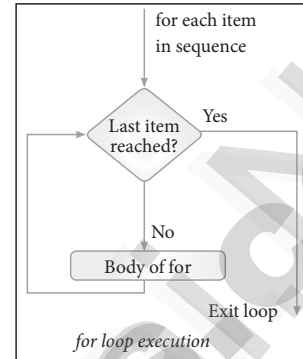
range (1,30,1) - will start the range of values from 1 and end at 29

range (2,30,2) - will start the range of values from 2 and end at 28

range (30,3,-3) - will start the range of values from 30 and end at 6

range (20) - will consider this value 20 as the

end value(or upper limit) and starts the range count from 0 to 19 (remember always range() will work till stop -1 value only)

**Example :**

#Program to illustrate the use of for loop - to print single digit even number

```
for i in range (2,10,2):
```

```
    print (i, end=' ')
```

**Output :**

2 4 6 8

36. (a) **String Operators :** Python provides the following operators for string operations. These operators are useful to manipulate string.

- (i) **Concatenation (+) :** Joining of two or more strings is called as Concatenation. The plus (+) operator is used to concatenate strings in python.

**Example :**

```
>>> "welcome" + "Python"
'welcomePython'
```

- (ii) **Append (+ =) :** Adding more strings at the end of an existing string is known as append. The operator += is used to append a new string with an existing string.

**Example :**

```
>>> str1="Welcome to "
>>> str1+="Learn Python"
>>> print (str1)
```

*Welcome to Learn Python*

- (iii) **Repeating (\*) :** The multiplication operator (\*) is used to display a string in multiple number of times.

**Example :**

```
>>> str1="Welcome "
>>> print (str1*4)
```

Welcome Welcome Welcome Welcome

- (iv) **String slicing :**

- Slice is a substring of a main string.

- A substring can be taken from the original string by using [ ] operator and index or subscript values.
- Thus, [ ] is also known as slicing operator. Using slice operator, you have to slice one or more substrings from a main string.

**General format of slice operation :**

```
str[start:end]
```

- Where **start** is the beginning index and **end** is the last index value of a character in the string.
- Python takes the end value less than one from the actual index specified.

**Example :****slice a single character from a string**

```
>>>str1="THIRUKKURAL"
>>>print (str1[0])
```

**Output :**

```
T
```

**(v) Stride when slicing string :**

- When the slicing operation, you can specify a third argument as the stride, which refers to the number of characters to move forward after the first character is retrieved from the string.
- The default value of stride is 1.
- Python takes the last value as n-1.
- You can also use negative value as stride, to prints data in reverse order.

**Example :**

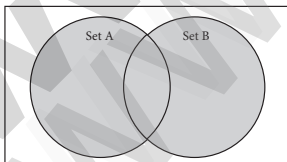
```
>>>str1="Welcome to learn Python"
>>>print (str1[::-2])
```

**Output :**

```
nh y re teoW
(OR)
```

**(b) Set operations:** The set operation such as Union, Intersection, difference and symmetric difference.

**Union:** It includes all elements from two or more sets



- In python, the operator | is used to union of two sets. The function union() is also used to join two sets in python.
- Example :** Program to Join (Union) two sets using union operator
 

```
set_A={2,4,6,8}
set_B={'A', 'B', 'C', 'D'}
U_set=set_A|set_B
```

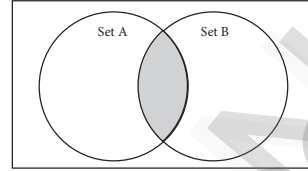
```
print(U_set)
```

**Output :**

```
{2, 4, 6, 8, 'A', 'D', 'C', 'B'}
```

**Intersection :**

- It includes the common elements in two sets



- The operator & is used to intersect two sets in python. The function intersection() is also used to intersect two sets in python.

- Example :** Program to intersect two sets using intersection operator

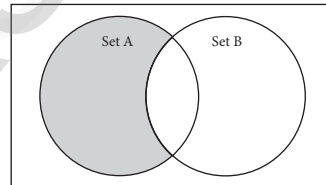
```
set_A={'A', 2, 4, 'D'}
set_B={'A', 'B', 'C', 'D'}
print(set_A & set_B)
```

**Output :**

```
{'A', 'D'}
```

**Difference :**

- It includes all elements that are in first set (say set A) but not in the second set (say set B)



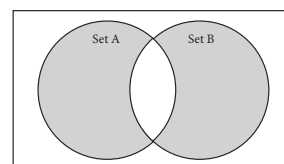
- The minus (-) operator is used to difference set operation in python. The function difference() is also used to difference operation.

- Example :** Program to difference of two sets using minus operator

```
set_A={'A', 2, 4, 'D'}
set_B={'A', 'B', 'C', 'D'}
print(set_A - set_B)
```

**Output : {2, 4}****Symmetric difference :**

- It includes all the elements that are in two sets (say sets A and B) but not the one that are common to two sets.



- The caret (^) operator is used to symmetric difference set operation in python. The function **symmetric\_difference()** is also used to do the same operation.

(iii) **Example :** Program to symmetric difference of two sets using caret operator

```
set_A={'A', 2, 4, 'D'}
```

```
set_B={'A', 'B', 'C', 'D'}
```

```
print(set_A ^ set_B)
```

**Output :**

```
{2, 4, 'B', 'C'}
```

37. (a) **The different types of a Data Model:** Hierarchical Model, Relational Model, Network Database Model, Entity Relationship Model, Object Model.

**Hierarchical Model :**

- (i) Hierarchical model was developed by IBM as Information Management System. In Hierarchical model, data is represented as a simple tree like structure form.
- (ii) This model represents a one-to-many relationship i.e., parent-child relationship. One child can have only one parent but one parent can have many children.
- (iii) This model is mainly used in IBM Main Frame computers.

**Relational Model :**

- (i) The Relational Database model was first proposed by E.F. Codd in 1970 . Nowadays, it is the most widespread data model used for database applications around the world.
- (ii) The basic structure of data in relational model is tables (relations). All the information's related to a particular type is stored in rows of that table.
- (iii) Hence tables are also known as relations in a relational model. A relation key is an attribute which uniquely identifies a particular tuple (row in a relation (table)).

**Network Model :** Network database model is an extended form of hierarchical data model. The difference between hierarchical and Network data model is :

- (i) In hierarchical model, a child record has only one parent node,
- (ii) In a Network model, a child may have many parent nodes. It represents the data in many-to-many relationships.
- (iii) This model is easier and faster to access the data.

**Entity Relationship Model. (ER model) :**

- (i) In this database model, relationship are created by dividing the object into entity and its characteristics into attributes.
- (ii) It was developed by Chen in 1976. This model is useful in developing a conceptual design for the database. It is very simple and easy to design logical view of data. The developer can easily understand the system by looking at ER model constructed.

**Object Model :**

- (i) Object model stores the data in the form of objects, attributes and methods, classes and Inheritance.
- (ii) This model handles more complex applications, such as Geographic information System (GIS), scientific experiments, engineering design and manufacturing.
- (iii) It is used in file Management System. It represents real world objects, attributes and behaviors. It provides a clear modular structure. It is easy to maintain and modify the existing code.

(OR)

(b) **Refer Sura's Guide - Chapter-12 - Part-IV - Q.No. 1 (5 Marks)**

38. (a)

| Mode | Description                                                                                               |
|------|-----------------------------------------------------------------------------------------------------------|
| 'r'  | Open a file for reading. (default)                                                                        |
| 'w'  | Open a file for writing. Creates a new file if it does not exist or truncates the file if it exists.      |
| 'x'  | Open a file for exclusive creation. If the file already exists, the operation fails.                      |
| 'a'  | Open for appending at the end of the file without truncating it. Creates a new file if it does not exist. |
| 't'  | Open in text mode. (default)                                                                              |
| 'b'  | Open in binary mode.                                                                                      |
| '+'  | Open a file for updating (reading and writing)                                                            |

(OR)

(b) **Refer Sura's Guide - Chapter-14 - Part-IV - Q.No. 3 (5 Marks)**

