KENDRIYA VIDYALAYA SANGATHAN, AGRA REGION

CLASS XII, COMPUTER SCIENCE (083)

Marking Scheme

PRE BOARD EXAMINATION –II (2020-21)

TIME:: 3 hrs   **MAX MARKS::70**

Question No	Part-A	Marks Allocated
	Section-I	
1.	Data items having fixed value are called .... **d) Literals**	1
2.	To print a line of text without ending it with a newline, ... argument is used with print(). **end=' '**	1
3.	Which among the following is not a Python Keyword? c) just	1

4.	Consider the following code and choose the correct output:
	
	 ```In [ ]: n1=60```
	 ```n2=20*3```
	 ```print(id(n1)==id(n2))```
	
	 a) True
		1

5.	print(24//4//2+5) will display output as: c) 8	1
6.	What will be the output of :: print("Hello! "+ " Corona may be fatal" + " for us .", sep= "&&", end='\t') **Hello! Corona may be fatal for us.**	1

7.	What will be the output of the following code?
	
	 ```In [26]: s=str(print())+"Two"```
	 ```b=str(print("hello", end=' '))+"0nne"```
	 ```print(s, " ",b)```
	
	 a) hello NoneTwo None0nne
		1

8.	Write a statement to remove a key 'CARROT' from the dictionary 'VEGETABLES'. del VEGETABLES['CARROT"] or VEGETABLES.pop['CARROT']	1
9.	What is the output of the following statement: print("xyyzxyzxzxyy".count('yy',1)) a) 2	1
10.	What will be the output when the following code is executed: s=[25,45,76,87,10] print(s[:-1]) **s=[25,45,76,87]**	1
11.	Which among the following can be used to import all names from a module m1 into the current calling namespace? **c) from m1 import ***	1
12	A .... is a special control structure that facilitates the row-by- row processing of records in the result set. **b) Cursor**	1
13.	To specify filtering condition for groups, the ....clause is used in MySql **having**	1
14.	A .... is a device that connects dissimilar networks. **gateway**	1
15.	In SQL, write a query to start working in the database "KVSAGRA". **USE KVSAGRA**	1
16.	Write an SQL statement, to display the record of all students from table STUDENTS whose last name column(Column name: LNAME)contains 5 letters ending with 'A'	

	Select * **From STUDENTS** **Where LNAME Like '_ _ _ _ A';**	
17.	**os** module in Python provides functions for working with files and directories	1
18.	Identify the type of cyber crime for the following situations: "Continuously sending bulk requests to a website so that it is not available to any other user." **DenialOf Service(DoS)**	1
19.	It is used by server to identify an user when he logs in next time by storing data in browser. What is it called?................. **cookies**	1
20.	**Plagiarism**	1
21.	**_init_.py**	1
	SECTION - II **Both the Case study based questions are compulsory. Attempt any 4 sub parts from each question. Each question carries 1 mark**	
22.	**A Coaching Institute STAR COACHING ACADEMY is considering to maintain a record of their students using SQL. As a database administrator, Rajesh has decided that::** **Name of the database:: staracademy** **Name of the table :: Engg** **The attributes are as follows:** ID- character Name- character of Variable length. Age- numeric City - Character Fee- numeric	

Phone- Character

ID	NAME	AGE	CITY	FEE	PHONE
P1	SAMEER	34	DELHI	45000	9811076656
P2	ARYAN	35	MUMBAI	54000	9911343989
P4	RAM	34	CHENNAI	45000	9810593578
P6	PREMLATA	36	BHOPAL	60000	9910139987
P7	SHIKHA	36	INDORE	34000	9912139456
P8	RADHA	33	DELHI	23000	8110668888

a) Write the command to create the database staracademy.

CREATE DATABASE STARACADEMY;

1

b) Identify the candidate keys in the above table.

ID, NAME

1

c) Name the datatype for column **Name.**

VARCHAR

1

d)Rajesh wants to add the following data into the attributes ID, Name and Fee in the above table Engg., Write the command for him:

ID- P10, Name - Shyam and Fee- 55000

INSERT INTO ENGG (ID, NAME, FEE) VALUES('P10', 'SHYAM', 55000);

1

e) Further, Rajesh wants to add one more column "Mode" with default value "ONLINE". Write the command for the same.

ALTER TABLE ENGG

ADD(MODE CHAR(7) DEFAULT 'ONLINE')

1

23	a) import pickle	1
	b) fb=open('Ball.dat','ab')	1
	c) r=pickle.load(f)	1
	d) print("Quantity of", **r[0]** ," are:: ", **r[1]**) (½ mark for each)	1
	e) search('Football')	1

		Part – B	
		Section - I	
24.	Which among the following are correct way to get the value of hobby key: student={'name':"Amit", 'class':'IX', 'Hobby':'Swimming'} c) m=student['marks'] d) m=student.get('marks') (1 mark for each correct option)	2
25.	1l3s5 7rEBOA13d	2

26	Positional Arguments are arguments passed to a function in correct positional order. The number and position of arguments must be matched. If we change their order, then the result will be changed.
	
	Eg- def subt(a,b):
	
	       print(a-b)
	
	  subt(200,100) ----------> **Positional Arguments**
	
	Keyword arguments are the named arguments with assigned values being passed in , the function call statement. If there is a function with many parameters and we want to specify only some of them in function call, then value for such parameters can be provided by using their name instead of position.
	
	Eg:
	
	def greet(name, msg):
	
	   print("Hello",name, msg)
	
	greet(msg='"Good Afternoon", name="Vinay") ------------>**Keyword Arguments**
	
	greet(name="Anil", msg="Good Morning")	

27	(i) NX*UA*FM* and (iii) NA*UM*FA* Min and Max values for count are 0 and 2 respectively	2
28.	L=[1,11,21,31] for i in range(len(L)): (1 mark) L[i]=L[i]+2 (1 mark) print(L)	2

	OR
	
	d1=**<u>dict()</u>**
	
	i=1
	
	n=**<u>int</u>**(input("Enter the number of entries:"))
	
	while(i<=n):
	
	   n=input("Enter name::")
	
	   b=input("Enter age:")
	
	   <u>d1[a]=b</u>
	
	   i=i+1
	
	l=**<u>d1.keys()</u>**
	
	for i in l:
	
	   print(i,'\t', **<u>d1[i]</u>**)
	
	        (½ mark for any 4 corrections)	

29	A primarykey is an attribute or a group of attributes that can uniquely identify tuples within the relation. There can be only one primary key in a relation. Primary key constraint is used to declare a primary key in a relation. Whereas a unique constraint is used to declare a candidate key that is a column with distinct values **Primary key constraint can be used only once in Create Table command whereas Unique constraint can be used for any columns**	2
30	mdb=mysql.connector.connect(host="localhost",user="root",passwd='123',database ='EXAM")	2
31	**Group by clause** is used to **group** the results of a SELECT query based on one or more columns. The **GROUP** BY statement **groups** rows that have the same values into summary rows, like "find the number of customers in each country". The **GROUP** BY statement is often used with aggregate functions (COUNT, MAX, MIN, SUM, AVG) to **group** the result-set by one or more columns.	2

	Or **Single row functions** are the one who work on **single row**and return one output per **row**. Eg- len(), round() **Multiple row functions** work upon group of rows and return one result for the complete set of rows. They are also known as Group **Functions**.eg--max(), count()	
32.	E-commerce is the buying and selling of goods and services**,** or the transmitting of funds or data over an electronic network. **(½ mark)** Advantages : 1. Round the clock availability 2. Speed of access 3. Wide availability of goods and services to the consumer 4. Easy accessibility 5. International reach **(½ mark for any three)**	
33.	i) Short Messaging Service ii) Post Office Protocol iii) Infrastructure as a Service iv) Session Initiation Protocol **(½ mark each)** **Or** a) Qualcomm Snapdragon b) Apple A10 Fusion d) Samsung Exynos f)Intel Atom **(½ mark each)**	
	SECTION II	
| 34 | ```
def Disp3Digit(L):
    s=0
    for i in L:
       c=0
       n=i
``` | 3 |

```
        while n>0:
           n=n//10
           c=c+1
        if c==3:
           s=s+i
     return s
```

35

```
   def readh():
      f=open('Poem.txt', 'r')
      f2=open('Hlines.txt','w')      (½  mark for opening files)
      l=f.readlines()                (½  mark for reading)
       for i in l:                   (½  mark for loop)
          i=i.lstrip()
          if i[0]=='H' or i[0]=='h':   (1 mark for checking first character)
             f2.write(i)             (½  mark for writing)
      f.close()
      f2.close()


   OR
   def  dispwords():
      f=open('Story..txt', 'r')     (½ mark )
      l=f.read()                    (½ mark )
      w=l.split()                   (½ mark )
      for i in w:                   (½ mark )
        if len(i)<4:                (½ mark )
           print(i," ")             (½ mark )
      f.close()
```

36	i)	1 mark for each

i)

ItemName	Max(Price)	Count(*)
Personal Computer	37000	3
Laptop	57000	2

ii)

CustomerName	Manufacturer
N Roy	PQR
H Singh	XYZ
R Pandey	COMP
C Sharma	PQR
K Agarwal	ABC

iii)

ItemName	NewPrice
Personal Computer	3500000
Laptop	5500000

37

```
def PUSH(Arr):
    s=[]
```

3

```
        for x in range(0,len(Arr)):
            if x%2==0:
                s.append(Arr[x])
        if len(s)==0:
            print("Empty Stack")
        else:
            print(s)


    OR


    def ADDCustomer(Cust):
        n=int(input("Enter Customer no::"))
        nm=input("Enter employee name::")
        s=int(input("Enter salary::"))
        a=(n,nm,s)
        Cust.append(a)
    def DeleteCustomer(Cust):
        if Cust==[]:
            print("Stack Empty")
        else:
            v=Cust.pop()
            print("Element being deleted",v)
            n= len(Cust)
            for i in range(n-1,-1,-1):
                print(Cust[i], end='-->')
```

38.	(i) Admin Block	
	(1 mark for correct answer)	
	(ii)	

(1 mark for correct answer)

(iii) Modem or Switch or Router

   (1 mark for correct answer)

(iv)Ethernet Cable

   (1 mark for correct answer)

v) TCP/IP or BroadBand

   (1 mark for correct answer)

39	i) Select * from Customer where city='Delhi';	1 mark for each
	ii) Select * from Item where Price between 35000 and 55000.	
	iii) Select C.I_ID, CustomerName, City, Itemname, Price	
	From Customer C, Item I	
	Where C.I_Id=I. I_ID;	
	iv) Select City , Count(*)	
	From Customer	
	Group By City	
	v) Update Item	
	Set Price=Price +1000	
40	def CreateFile(): **(3 marks for creating and inputting record)**	
	n=input("Enter book no::")	
	nm=input("Enter book Name::")	
	a=input("Enter the author name::")	
	p=input("Ente price::")	
	r=[n,nm,a,p]	

```python
    print(r)
    f=open('books.csv','a')
    csv_w=csv.writer(f, delimiter=',')
    csv_w.writerow(r)
    print("File created")
def countrec(author):        (2 marks for counting record)
    c=0
    f=open('books.csv','r')
    csv_r=csv.reader(f)
    for r in csv_r:
        if r[2].lower()==author.lower():
            c=c+1
    return c
```

OR

```python
import csv
def CreateF():                (2 ½  marks)
    fd=['ItemID','ItemName', 'Qty','Price']
    r=[['I02','Cakes','230','92'],['I04','Cookies','110','82'],['I08','Pastries','100','43']]
    f="Items.csv"
    with open(f,'w')as fi:
        c=csv.writer(fi, delimiter=':')
        c.writerow(fd)
        for i in r:
            c.writerow(i)
    print("File created")
# For searching a record in csv file
def search(n):                (2 ½  marks)
```

```python
f=open("Items.csv",'r')
cro=csv.reader(f)
fd=0
h=next(cro)
for r in cro:
    r=r[0].split(':')
    if r[1].lower()==n.lower():
        print("Record found", r)
        fd=1
    if fd==0:
        print(n, "REcord not found")
```