COMPUTER SCIENCE (083)

CLASS: XII (SESSION: 2023-24)

Sample Paper – 2 (Answer Key)

	SECTION A	
1.	True	1
2.	a. A digit	1
3.	c. Raises an Error	1
4.	b. 11	1
5.	c. 8	1

6.	a. Append	1
7.	a. ALTER TABLE	1
8.	b. Filters rows based on a condition	1
9.	c. Tuples	1
10.	b. PARIS* TOKYO*	1
11.	d. open(file_name, file_mode, [new line character])	1
12.	c. ORDER BY	1
13.	b. Circuit Switching	1
14.	a. -511	1

15.	c. count(*)	1
16.	b. 3	1
17.	a. Both A and R are true and R is the correct explanation for A	1
18.	a. Both A and R are true and R is the correct explanation for A	1
	SECTION B	

19.	`def execmain():`
	`  x = int(input("Enter a number:"))`
	`  if(abs(x) == x):`
	`  print("You entered a positive number:")`
	`  else:`
	`  x *= -1`
	`  print("Number made positive:", x)`
	`execmain()`
		2

20.	**Advantages of Computer Networks:** i. Computer networks let devices talk to each other, making it easy to share things like files and printers. ii. They help with teamwork and let people access information from different places, like working from home. **Disadvantages of Computer Networks:** i. Sometimes, networks can have problems like hackers trying to steal information or the network going down, causing issues. ii. Setting up and keeping networks running can also be costly and a bit complicated for some people	2

	OR Hub: Sends data to all connected devices; Shares available bandwidth among all ports; Does not learn MAC addresses Switch: Sends data only to the specific device; Dedicated bandwidth for each port; Learns MAC addresses to make forwarding decisions	
21.	a. Welcome-to-the-Python-world! b. [1, 2, 3, 4, 5, 6, 7, 8, 9]	1 1
22.	Constraints in RDBMS, such as foreign key constraints, ensure referential integrity by enforcing relationships between tables. For instance, a foreign key constraint prevents inserting data that violates the defined relationships, ensuring that data remains consistent and accurate across interconnected tables.	2
23.	a. i. POP: Point to Point Protocol ii.TCP/IP: Transmission Control Protocol / Internet Protocol	1 1

24.	In MySQL, the WHERE clause filters rows before retrieving them, specifying conditions for selection. Example: SELECT * FROM employees WHERE salary > 50000; retrieves employees with a salary greater than 50000. **OR** DDL (Data Definition Language) includes commands like CREATE and ALTER for database structure modification. Example: CREATE TABLE Students (ID INT, Name VARCHAR(50)); DML (Data Manipulation Language) includes commands like INSERT and UPDATE for data manipulation. Example: INSERT INTO Students VALUES (1, 'John');	2
25.	6 6	2
	SECTION C	

26.	`30 % 41`
	`52 % 60`
	`40 % 25`
		3

27. (i) 3

SID	SNAME	AGE	COURSE	GRADE	ENROLL_DATE
101	John	22	Computer Sci.	A	2022-02-15
102	Maria	23	Biology	B	2022-02-10
103	Raj	21	Physics	C	2021-12-05
105	Aakriti	22	Chemistry	B	2022-01-02

(ii)

SID	SNAME	COURSE	GRADE
101	John	Computer Sci.	A
107	Jyoti	Computer Sci	C

(iii)

SNAME	GENDER	COURSE
Eshaan	Male	Math

28.

```
def count_words_in_file(file_path):
    try:
        with open('WisdomQuotes.txt', 'r') as file:
            content = file.read()
            words = content.split()
            total_words = len(words)
            print("Total number of words in the file:", total_words)
    except FileNotFoundError:
        print("File not found. Please provide a valid file path.")
```

OR

```
def count_my(file_path):
    try:
        with open(file_path, 'r') as file:
            content = file.read()
            count = content.lower().split().count("my")
            print('"my" occurs', count, 'times')
    except FileNotFoundError:
       print('File not found. Please provide a valid file path.')
```

29. 3

i. The most appropriate column to be considered as the Primary key is CustomerID.

ii. The cardinality of the table is the number of rows it contains. In this case, the cardinality is 5.

iii.

a.
```
ALTER TABLE Customers
ADD COLUMN DateOfDelivery DATE;
```

b.
```
ALTER TABLE Customers
DROP COLUMN City;
```

30.

i.
```
def Push_Student(stack, SDetails):
    if SDetails['Percentage'] > 85:
        stack.append(SDetails['Student_name'])
        print(SDetails['Student_name'] + " pushed into the stack.")
    else:
        print(SDetails['Student_name'] + " does not meet the criteria.")
```

ii.
```
def Pop_Student(stack):
    if not stack:
        print("Stack Empty")
    else:
        popped_student = stack.pop()
        print("Popped Student: " + popped_student)
```

3

Section - D

31.

(i)
```
SELECT BOOK.Title, BOOK.Price FROM BOOK, AUTHOR
WHERE BOOK.AuthorID = AUTHOR.AuthorID
AND AUTHOR.Nationality = 'Indian';
```

(ii)
```
SELECT Genre, AVG(Price) AS AvgPrice
FROM BOOK GROUP BY Genre;
```

(iii)
```
SELECT Title, Price
FROM BOOK
WHERE Price > 300;
```

(iv)
```
SELECT Title, Genre FROM BOOK
WHERE Genre IN ('Fiction', 'Fantasy')
ORDER BY Genre DESC;
```

(v)
```
ALTER TABLE Books
MODIFY COLUMN Price DECIMAL(8, 2);
```

4

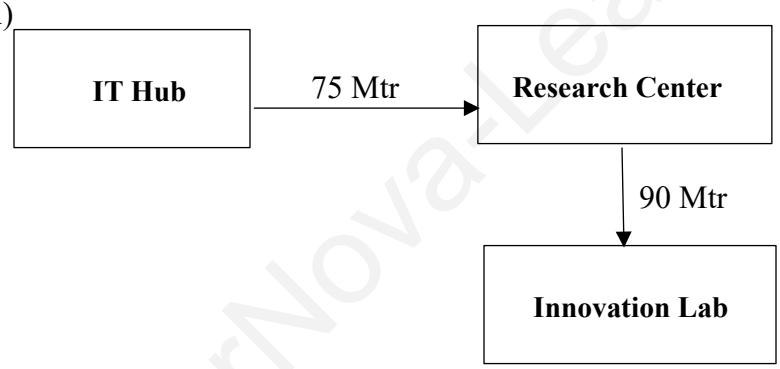
32.

```
import csv
def acceptProject():
    file = open("Projects.csv", mode='a', newline='')
    writer = csv.writer(file)
    proj_id = int(input("Enter Project ID: "))
    proj_title = input("Enter Project Title: ")
    st_name = input("Enter Student Name: ")
    status = input("Enter Project Status(Completed/In Progress/Not
Started): ")
    writer.writerow([proj_id, proj_title, st_name, status])
    print("Record added successfully!")
    file.close()

def completedProjectsCount():
    file = open("Projects.csv", mode='r')
    reader = csv.DictReader(file)

    completed_count = 0
    for row in reader:
        if row['Status'] == 'Completed':
            completed_count += 1

    print("Number of completed projects:", completed_count)
    file.close()
```

4

Section - E

33. (i)



(ii) Ethernet switches can be used to connect computers in each building, forming a local area network (LAN).
(iii) The IT Hub should be designated as the server since it has the highest number of computers (150), making it suitable for handling server functions and services for the campus.
(iv) For connecting the campus with a nearby office 5 kilometers away, a suitable transmission medium would be optical fiber.
(v) (iii) Switch

5

34. (i) The read() method reads the entire content of a file as a single string, while the readline() method reads one line at a time.

(ii)

```
def findExpensiveBooks():
    try:
        with open("BOOKS.TXT", "r") as file:
            expensive_books = []
```

1+4= 5

```
            for line in file.readlines():
                book_info = line.split(',')
                if float(book_info[2]) > 500:
                    expensive_books.append(book_info[0])

            if expensive_books:
                with open("EXPENSIVE_BOOKS.TXT", "w") as output_file:
                    for book_title in expensive_books:
                        output_file.write(book_title + '\n')

            return len(expensive_books)

    except Exception as e:
        print("An error occurred:", e)
        return 0

# Call the function
num_records_copied = findExpensiveBooks()
print("Total number of expensive books copied:", num_records_copied)
```

OR

(i) 'a' mode: It opens the file for appending data. If the file exists, the new data is added at the end otherwise a new file is created. On the other hand, 'w' mode opens the file for writing data. If the file exists, it is truncated, and new data is written from the beginning. If the file does not exist, a new file is created.

(ii)

```
def retrieveHighSalary():
      try:
          with open("EMPLOYEES.DAT", "r") as file:
            print("Employees with a salary greater than Rs.75,000:")

            for line in file.readlines():
                emp_info = line.split(',')
                emp_name = emp_info[1].strip()
                salary = float(emp_info[2])

                if salary > 75000:
                    print(emp_name)

        except Exception as e:
            print("An error occurred:", e)

# Call the function
retrieveHighSalary()
```

35. (i) Referential integrity in RDBMS ensures that relationships between tables are maintained, preventing unmatched foreign key values, maintaining consistency, and avoiding duplicated records. 1+4=5

(ii)

```
import mysql.connector as mysql

connection = mysql.connect(host="localhost", user="admin",
password="abc@123", database="CompanyDB")
# Get user input
emp_id_to_update = int(input("Enter the Employee ID to update: "))
new_emp_name = input("Enter the new Employee Name: ")
new_emp_salary = float(input("Enter the new Employee Salary: "))
new_emp_department = input("Enter the new Employee Department: ")

# Create a cursor
cursor = connection.cursor()

# SQL query to update the record
update_query = "UPDATE Employee SET emp_name = %s, emp_salary = %s,
emp_department = %s WHERE emp_id = %s"

# Data to be updated
data = (new_emp_name, new_emp_salary, new_emp_department,
emp_id_to_update)

# Execute the query
cursor.execute(update_query, data)

# Commit the changes
connection.commit()

print("Record updated successfully!")

# Close the cursor and connection
cursor.close()
connection.close()
```

OR

(i) A candidate key is any set of one or more attributes (columns) in a database table that can uniquely identify each record. An alternate key, on the other hand, is any candidate key that is not selected as the primary key.

(ii)

```
import mysql.connector as mysql

# Establish the database connection
con1 = mysql.connect(host="localhost", user="root", password="sunshine",
database="Inventory")
mycursor = con1.cursor()

# SQL query to select records with a discount greater than 50%
query = "SELECT * FROM Product WHERE discount > {}".format(50)
```

```
mycursor.execute(query)
data = mycursor.fetchall()

# Display the records
for rec in data:
    print(rec)

# Close the database connection
con1.close()
```