

Series : S7PQR

SET~4

प्रश्न-पत्र कोड
Q.P. Code

91/7

रोल नं.
Roll No.

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

परीक्षार्थी प्रश्न-पत्र कोड को उत्तर-पुस्तिका के मुख-पृष्ठ पर अवश्य लिखें।
Candidates must write the Q.P. Code on the title page of the answer-book.

नोट	NOTE
(I) कृपया जाँच कर लें कि इस प्रश्न-पत्र में मुद्रित पृष्ठ 47 हैं।	(I) Please check that this question paper contains 47 printed pages.
(II) प्रश्न-पत्र में दाहिने हाथ की ओर दिए गए प्रश्न-पत्र कोड को परीक्षार्थी उत्तर-पुस्तिका के मुख-पृष्ठ पर लिखें।	(II) Q.P. Code given on the right hand side of the question paper should be written on the title page of the answer-book by the candidate.
(III) कृपया जाँच कर लें कि इस प्रश्न-पत्र में 37 प्रश्न हैं।	(III) Please check that this question paper contains 37 questions.
(IV) कृपया प्रश्न का उत्तर लिखना शुरू करने से पहले, उत्तर-पुस्तिका में यथा स्थान पर प्रश्न का क्रमांक अवश्य लिखें। []	(IV) Please write down the Serial Number of the question in the answer-book at the given place before attempting it.
(V) इस प्रश्न-पत्र को पढ़ने के लिए 15 मिनट का समय दिया गया है। प्रश्न-पत्र का वितरण पूर्वाह्न में 10.15 बजे किया जाएगा। 10.15 बजे से 10.30 बजे तक परीक्षार्थी केवल प्रश्न-पत्र को पढ़ेंगे और इस अवधि के दौरान वे उत्तर-पुस्तिका पर कोई उत्तर नहीं लिखेंगे।	(V) 15 minute time has been allotted to read this question paper. The question paper will be distributed at 10.15 a.m. From 10.15 a.m. to 10.30 a.m., the candidates will read the question paper only and will not write any answer on the answer-book during this period.

कम्प्यूटर साइंस

COMPUTER SCIENCE

निर्धारित समय : 3 घण्टे
Time allowed : 3 hours

अधिकतम अंक : 70
Maximum Marks : 70

सामान्य निर्देश :

- (i) इस प्रश्न-पत्र में 37 प्रश्न हैं।
- (ii) सभी प्रश्न अनिवार्य हैं। परंतु कुछ प्रश्नों में आंतरिक विकल्प दिए गए हैं। ऐसे प्रश्नों में केवल एक विकल्प को चुनिए।
- (iii) इस प्रश्न-पत्र को 5 खण्डों क, ख, ग, घ और ङ में बाँटा गया है।
- (iv) खण्ड – क में 21 प्रश्न (1 से 21) हैं। प्रत्येक प्रश्न के लिए 1 अंक निर्धारित है।
- (v) खण्ड – ख में 7 प्रश्न (22 से 28) हैं। प्रत्येक प्रश्न के लिए 2 अंक निर्धारित हैं।
- (vi) खण्ड – ग में 3 प्रश्न (29 से 31) हैं। प्रत्येक प्रश्न के लिए 3 अंक निर्धारित हैं।
- (vii) खण्ड – घ में 4 प्रश्न (32 से 35) हैं। प्रत्येक प्रश्न के लिए 4 अंक निर्धारित हैं।
- (viii) खण्ड – ङ में 2 प्रश्न (36 और 37) हैं। प्रत्येक प्रश्न के लिए 5 अंक निर्धारित हैं।
- (ix) सभी प्रोग्रामिंग प्रश्नों का उत्तर केवल *Python Language* में दिया जाना है।
- (x) बहुविकल्पी प्रश्नों (*Multiple Choice Questions*) के मामले में सही उत्तर का *text* भी लिखा जाना चाहिए।

खण्ड क

21×1=21

1. निम्नलिखित में से कौन-सा पाइथन (Python) में कीवर्ड (keyword) है ? 1
- (A) Break (B) None
(C) For (D) Class
2. निम्नलिखित स्टेटमेंट (statement) का आउटपुट (output) क्या होगा ? 1
- `print(18/3 == 18//3)`
- (A) 0 (B) False
(C) 1 (D) True
3. नीचे दिए गए स्टेटमेंट्स (statements) पर विचार कीजिए और फिर दिए गए विकल्पों में से सही आउटपुट (output) चुनिए : 1
- `D={"A": "MUMBAI", "B": "CHENNAI"}`
`print("C" in D)`
- (A) False (B) 0
(C) True (D) 1

General Instructions :

- (i) *This question paper contains **37** questions.*
- (ii) ***All** questions are **compulsory**. However, internal choices have been provided in some questions. Attempt only one of the choices in such questions.*
- (iii) *The paper is divided into **5** Sections – **A, B, C, D** and **E**.*
- (iv) ***Section A**, consists of **21** questions (**1 to 21**). Each question carries **1** mark.*
- (v) ***Section B**, consists of **7** questions (**22 to 28**). Each question carries **2** marks.*
- (vi) ***Section C**, consists of **3** questions (**29 to 31**). Each question carries **3** marks.*
- (vii) ***Section D**, consists of **4** questions (**32 to 35**). Each question carries **4** marks.*
- (viii) ***Section E**, consists of **2** questions (**36 and 37**). Each question carries **5** marks.*
- (ix) *All programming questions are to be answered using Python Language only.*
- (x) *In case of MCQs, text of the correct answer should also be written.*

SECTION A

21×1= 21

1. Which of the following is a keyword in Python ? 1
(A) **Break** (B) **None**
(C) **For** (D) **Class**
2. What will be the output of the following statement ? 1

```
print(18/3 == 18//3)
```


(A) **0** (B) **False**
(C) **1** (D) **True**
3. Consider the statements given below and then choose the correct output from the given options : 1

```
D={"A": "MUMBAI", "B": "CHENNAI"}
```

```
print("C" in D)
```


(A) **False** (B) **0**
(C) **True** (D) **1**

4. निम्नलिखित पाइथन (Python) कोड (code) का आउटपुट (output) क्या होगा ? 1

```
try:
    num = 5
    text = "15"
    result = str(num) + text
    print(result)
except:
    print("Error")
```

- (A) Error (B) 515
(C) 20 (D) 75

5. निम्नलिखित स्टेटमेंट (statement) का आउटपुट (output) क्या होगा ? 1

```
print("X" * 2, "Y", end = 2 * "X", sep = "#")
```

- (A) Error (B) YY#XX
(C) XXYY# (D) XX#YXX

6. यदि L एक खाली लिस्ट (empty list) है, तो निम्नलिखित में से कौन-सा एक्सप्रेशन (expression) अमान्य है ? 1

- (A) L + 'Hello' (B) L + []
(C) L + [2] (D) L * 2

7. निम्नलिखित में से कौन-सा स्टेटमेंट (statement) स्ट्रिंग (string) "STRING" से सबस्ट्रिंग (substring) "STR" (अर्थात् पहले 3 कैरेक्टर्स (characters)) रिटर्न (return) करेगा ? 1

- (A) "STRING"[:-3]
(B) "STRING"[-3::-1]
(C) "STRING"[3:]
(D) "STRING".startswith(3)

4. What will be the output of the following Python code ? 1
- ```
try:
 num = 5
 text = "15"
 result = str(num) + text
 print(result)
except:
 print("Error")
```
- (A) Error (B) 515  
(C) 20 (D) 75
5. What will be the output of the following statement ? 1
- ```
print("X"* 2,"Y",end = 2 * "X", sep = "#")
```
- (A) Error (B) YY#XX
(C) XXYY# (D) XX#YXX
6. If **L** is an empty list, then which of the following expressions are invalid ? 1
- (A) **L + 'Hello'** (B) **L + []**
(C) **L + [2]** (D) **L * 2**
7. Which of the following statements will return the substring "**STR**" (i.e. the first 3 characters) from the string "**STRING**" ? 1
- (A) **"STRING"[:-3]**
(B) **"STRING"[-3::-1]**
(C) **"STRING"[3:]**
(D) **"STRING".startswith(3)**

8. नीचे दिए गए स्टेटमेंट्स (statements) पर विचार कीजिए और फिर दिए गए विकल्पों में से सही आउटपुट (output) चुनिए :

1

```
K=('A','B')
V='TEST'
D = dict.fromkeys(K, V)
print(D)
```

- (A) {'A': 'TEST', 'B': None}
- (B) {'A': None, 'B': 'TEST'}
- (C) {'A': 'TEST', 'B': 'TEST'}
- (D) Error

9. पाइथन (Python) में `clear()` के बारे में निम्नलिखित में से कौन-सा सत्य है ?

1

- (A) यह डिक्शनरी (dictionary) को उसके सभी एलिमेंट्स (elements) के साथ डिलीट (delete) कर देता है।
- (B) यह डिक्शनरी (dictionary) के सभी आइटम्स (items) को डिलीट (delete) कर देता है, जिसके परिणामस्वरूप एक एम्प्टी (empty) डिक्शनरी (dictionary) मिलती है।
- (C) यदि `D` एक डिक्शनरी (dictionary) है, तो `clear(D)` सही स्टेटमेंट (statement) है।
- (D) पाइथन (Python) में ऐसा कोई फ़ंक्शन (function) / मेथड (method) मौजूद नहीं है।

10. नीचे दिए गए स्टेटमेंट्स (statements) पर विचार कीजिए और फिर दिए गए विकल्पों में से सही आउटपुट (output) चुनिए :

1

```
L = ["A", "B"]
S=['C','D']
L.extend(S)
print(L)
```

- (A) ['A', 'B', 'C', 'D']
- (B) ['A', 'B', 'CD']
- (C) ['A', 'B', ['CD']]
- (D) ['A', 'B', ['C', 'D']]

8. Consider the statements given below and then choose the correct output from the given options : 1

```
K=('A','B')  
V='TEST'  
D = dict.fromkeys(K, V)  
print(D)
```

- (A) {'A': 'TEST', 'B': None}
- (B) {'A': None, 'B': 'TEST'}
- (C) {'A': 'TEST', 'B': 'TEST'}
- (D) Error

9. Which of the following is true about `clear()` in Python ? 1

- (A) It deletes the dictionary itself with all its elements.
- (B) It deletes all items of the dictionary resulting in an empty dictionary.
- (C) If `D` is a dictionary, then `clear (D)` is the correct statement.
- (D) No such function/method exists in Python.

10. Consider the statements given below and then choose the correct output from the given options : 1

```
L = ["A", "B"]  
S=['C','D']  
L.extend(S)  
print(L)
```

- (A) ['A', 'B', 'C', 'D']
- (B) ['A', 'B', 'CD']
- (C) ['A', 'B', ['CD']]
- (D) ['A', 'B', ['C', 'D']]

11. पहले से मौजूद बाइनरी पिकल्ड फ़ाइल (binary pickled file) में अधिक डेटा (data) जोड़ने के लिए उसे खोलने का सही मोड (mode) निम्नलिखित में से कौन-सा विकल्प है ? 1
- (A) 'rb' (B) 'ab'
(C) 'w' (D) 'r'
12. सत्य (True) या असत्य (False) बताइए : 1
- SQL में, किसी एट्रिब्यूट (attribute) पर **UNIQUE** कंस्ट्रेंट (constraint) यह सुनिश्चित करता है कि उसमें मान (values) खाली न छोड़े जाएँ।
13. **A** और **B** डिग्री (degrees) वाले दो SQL टेबल्स (tables) के कार्टेशियन प्रोडक्ट (Cartesian Product) के बाद प्राप्त होने वाले रिलेशन (relation) की डिग्री (degree) क्या होगी ? 1
- (A) $A \times B$ (B) $A + B$
(C) $A - B$ (D) A / B
14. SQL के बारे में निम्नलिखित में से कौन-सी बात सही **नहीं** है ? 1
- (A) SQL केस सेंसिटिव (case sensitive) है।
(B) SQL स्टेटमेंट हमेशा कोलन (:) पर खत्म होता है।
(C) SQL का पूर्ण रूप Structured Query Language है।
(D) SQL का इस्तेमाल क्वेरी के ज़रिए टेबल से डाटा निकालने के लिए किया जाता है।
15. MySQL में बनाई गई टेबल (table) के किसी कॉलम (column) में प्राइमरी की (primary key) जोड़ने के लिए निम्नलिखित में से किसका उपयोग किया जाता है ? 1
- (A) **INSERT** (B) **MODIFY**
(C) **UPDATE** (D) **ALTER**

11. Which of the following options is the correct mode to open a binary pickled file to add more data in a previously existing binary pickled file ? 1
- (A) 'rb' (B) 'ab'
(C) 'w' (D) 'r'
12. State True or False : 1
- In SQL, the **UNIQUE** constraint on an attribute ensures that values in it are not left empty.
13. The degree of the resulting relation after the Cartesian Product of two SQL tables with degrees **A** and **B** will be : 1
- (A) $A \times B$ (B) $A + B$
(C) $A - B$ (D) A / B
14. Which of the following is **not** true about SQL ? 1
- (A) SQL is case sensitive.
(B) An SQL statement will always end with a colon (:).
(C) SQL stands for Structured Query Language.
(D) SQL is used to retrieve data from tables through queries.
15. Which of the following is used to add a primary key to a column in a table created in MySQL ? 1
- (A) **INSERT** (B) **MODIFY**
(C) **UPDATE** (D) **ALTER**

16. MySQL में, निम्नलिखित में से कौन-सा ऑपरेटर **WHERE** clause के साथ उपयोग किया जाता है और यदि मान (value) निर्दिष्ट मानों के सेट से संबंधित हो, तो **True** लौटाता है ? 1
- (A) **DISTINCT** (B) **IN**
(C) **LIKE** (D) **NOT**
17. निम्नलिखित में से कौन-सा नेटवर्किंग माध्यम एक (uni)-दिशात्मक है और लंबी दूरी के वायरलेस संचार के लिए उपयोग हेतु सर्वाधिक उपयुक्त है ? 1
- (A) रेडियो वेव (Radio waves)
(B) इन्फ्रारेड (Infrared)
(C) ब्लूटूथ (Bluetooth)
(D) माइक्रोवेव (Microwaves)
18. वेब पेज (Web page) को एक्सेस (access) करने के लिए निम्नलिखित में से कौन-सा सही प्रोटोकॉल (protocol) है ? 1
- (A) **HTTP** (B) **FTP**
(C) **VoIP** (D) **SMTP**
19. XML के बारे में निम्नलिखित में से कौन-सा विकल्प सही है ? 1
- (A) इसका पूरा नाम Expert Manipulating Language है ।
(B) यह एक प्रोग्रामिंग भाषा है ।
(C) यह एक डायनामिक (dynamic) भाषा है, जिसका उपयोग वेब पेजों के माध्यम से डेटा को स्टोर और ट्रांसपोर्ट करने के लिए किया जाता है ।
(D) यह HTML के समान है ।

16. In MySQL, which of the following operator is used with **WHERE** clause and returns a **True** if the value belongs to the set of specified values ? 1
- (A) **DISTINCT** (B) **IN**
(C) **LIKE** (D) **NOT**
17. Which of the following networking medium is uni-directional and most appropriate to be used for long distance wireless communication ? 1
- (A) Radio waves
(B) Infrared
(C) Bluetooth
(D) Microwaves
18. Which of the following is the correct protocol used for accessing a Web page ? 1
- (A) HTTP (B) FTP
(C) VoIP (D) SMTP
19. Which of the following options is true about XML ? 1
- (A) It stands for Expert Manipulating Language.
(B) It is a programming language.
(C) It is a dynamic language used to store and transport data through web pages.
(D) It is same as HTML.

प्रश्न 20 और 21 अभिकथन (A) और कारण (R) पर आधारित प्रश्न हैं। सही विकल्प चुनिए :

- (A) अभिकथन (A) और कारण (R) दोनों सही हैं और कारण (R), अभिकथन (A) का सही स्पष्टीकरण है।
- (B) अभिकथन (A) और कारण (R) दोनों सही हैं, परंतु कारण (R), अभिकथन (A) का सही स्पष्टीकरण नहीं है।
- (C) अभिकथन (A) सही है, परंतु कारण (R) ग़लत है।
- (D) अभिकथन (A) ग़लत है, परंतु कारण (R) सही है।

20. अभिकथन (A) : पाइथन (Python) में **upper()** फ़ंक्शन/मैथड (function/method) दी गई स्ट्रिंग (string) के सभी लोअरकेस अक्षरों (lowercase letters) को अपरकेस (uppercase) में बदलकर स्ट्रिंग (string) को रिटर्न (return) करता है।

कारण (R) : पाइथन (Python) में स्ट्रिंग्स (strings) म्यूटेबल (mutable) ऑब्जेक्ट्स (objects) होते हैं।

1

21. अभिकथन (A) : यदि हम किसी टेबल (table) के एट्रिब्यूट (attribute) को फॉरेन की (Foreign key) के रूप में परिभाषित करते हैं, तो **UNIQUE** और **NOT NULL** कंस्ट्रेंट्स (constraints) उस पर स्वतः रूप से लागू हो जाते हैं।

कारण (R) : किसी टेबल (table) का फॉरेन की (Foreign key) कॉलम (column) अपने मान (values) दूसरी टेबल (table) की प्राइमरी की (Primary key) से लेता है।

1

Questions **20** and **21** are Assertion (A) and Reason (R) based questions. Mark the correct choice as :

- (A) Both Assertion (A) and Reason (R) are true and Reason (R) is the correct explanation of the Assertion (A).
- (B) Both Assertion (A) and Reason (R) are true, but Reason (R) is **not** the correct explanation of the Assertion (A).
- (C) Assertion (A) is true, but Reason (R) is false.
- (D) Assertion (A) is false, but Reason (R) is true.

20. Assertion (A) : The **upper ()** function/method in Python returns the string with all lowercase letters of the given string converted to uppercase.

Reason (R) : Strings are mutable objects in Python.

1

21. Assertion (A) : If we define an attribute of a table as a Foreign key, then **UNIQUE** and **NOT NULL** constraints are automatically applied to it.

Reason (R) : The Foreign key column of a table takes its values from the Primary key of another table.

1

खण्ड ख

22. पाइथन स्ट्रिंग्स (Python Strings) के संबंध में `find()` और `index()` के बीच अंतर स्पष्ट कीजिए। दोनों के लिए सहायक उदाहरण दीजिए। 2
23. पाइथन फ़ंक्शन्स (Python functions) में डिफ़ॉल्ट पैरामीटर (default parameters) क्या हैं ? एक उदाहरण की सहायता से समझाइए। 2
24. (I) **PRICE** नाम की निम्नलिखित टपल (tuple) पर विचार कीजिए : 2

PRICE = (100, 55, 72, 65, 85, 90)

- (क) पाइथन (Python) में एक सिंगल लाइन स्टेटमेंट (single line statement) लिखिए जो टपल (tuple) **PRICE** के कंटेंट (content) को उनके मानों (values) के बढ़ते क्रम (ascending order) में प्रदर्शित कीजिए।

अथवा

- (ख) दिए गए टपल (tuple) में सभी मानों (values) का औसत **PRICE** ढूँढ़ने और प्रदर्शित करने के लिए पाइथन (Python) में एक सिंगल लाइन स्टेटमेंट (single line statement) लिखिए।
- (II) (क) एक स्ट्रिंग (string) **s** के कंटेंट (content) को इस प्रकार प्रदर्शित करने के लिए अंतर्निहित फ़ंक्शन्स (built-in functions) / मेथड्स (methods) का उपयोग करके एक सिंगल लाइन पाइथन स्टेटमेंट (single line Python statement) लिखिए जिससे स्ट्रिंग **s** में अक्षर 'i' के प्रत्येक स्थान पर एक अंडरस्कोर (underscore), यानी '_' आ जाए।

उदाहरण के लिए, यदि स्ट्रिंग **s** है :

S = "time is precious"

तो स्टेटमेंट को इस प्रकार आउटपुट (output) प्रदर्शित करना चाहिए :

t_me _s prec_ous

अथवा

- (ख) **myList** नाम की लिस्ट (list) से एलिमेंट (element) 25 की पहली अकरोन्स (occurrence) को डिलीट (delete) करने के लिए अंतर्निहित फ़ंक्शन (built-in function) / मेथड (method) का उपयोग करके एक पाइथन स्टेटमेंट (statement) लिखिए।

SECTION B

22. Differentiate between `find()` and `index()` with respect to strings in Python. Give supporting examples for both. 2
23. What are default parameters in Python functions ? Explain with the help of an example. 2
24. (I) Consider the following tuple named **PRICE** : 2

PRICE = (100, 55, 72, 65, 85, 90)

- (a) Write a single line statement in Python that displays the contents of the tuple **PRICE** in ascending order of the values.

OR

- (b) Write a single line statement in Python to find and display the average **PRICE** of all the values in the given tuple.
- (II) (a) Write a single line Python statement using built-in functions/methods to display the content of a string **S** such that every occurrence of the alphabet 'i' of the string **S** is replaced with an underscore, i.e ' _ '.

For example, if the string **S** is :

S = "time is precious"

then the statement should display the output as :

t_me _s prec_ous

OR

- (b) Write a Python statement using built-in function/method to delete the first occurrence of element **25** from the list named **myList**.

25. निम्नलिखित कोड (code) को निष्पादित (execute) करने पर दिए गए विकल्पों में से कौन-सा संभावित आउटपुट (output) प्रदर्शित होने की उम्मीद है ? अपने उत्तर का औचित्य सिद्ध कीजिए। 2

```
import random

ITEMS={1:"PEN", 2: "ERASER", 3: "PENCIL"}

for i in range(len(ITEMS)):

    print(ITEMS[random.randint(0,i)+1],end="*")
```

- (A) PEN*PENCIL*PENCIL*
- (B) PENCIL*PEN*ERASER*
- (C) PEN*PENCIL*ERASER*
- (D) PEN*ERASER*PEN*

26. नीचे दिया गया कोड (code) दी गई लिस्ट (list) से उन सभी संख्याओं को प्रदर्शित करने के उद्देश्य से लिखा गया है जो 7 से विभाज्य नहीं (not divisible) हैं। हालांकि, कोड में सिंटैक्स (syntax) और लॉजिकल (logical) त्रुटियाँ (errors) हैं। सभी त्रुटियों को दूर करने के बाद इसे फिर से लिखिए। इसके अलावा, किए गए सभी सुधारों को रेखांकित (underline) कीजिए। 2

```
define display(L):

    for i in range(0,len(L))

        if i % 7 == 0:

            break

        else:

            print(L[i])

numbers = [11,21,33,42,55]

display(numbers)
```


25. What possible output from the given options is expected to be displayed when the following code is executed ? Justify your answer. 2

```
import random

ITEMS={1:"PEN", 2: "ERASER", 3: "PENCIL"}

for i in range(len(ITEMS)):

print(ITEMS[random.randint(0,i)+1],end="*")
```

- (A) PEN*PENCIL*PENCIL*
- (B) PENCIL*PEN*ERASER*
- (C) PEN*PENCIL*ERASER*
- (D) PEN*ERASER*PEN*
26. The code given below is intended to display all numbers not divisible by 7 from the given list. However, there are syntax and logical errors in the code. Rewrite it after removing all errors. Also, underline all the corrections made. 2

```
define display(L):

    for i in range(0,len(L))

        if i % 7 == 0:

            break

        else:

            print(L[i])

numbers = [11,21,33,42,55]

display(numbers)
```

27. EcoHarvest Solutions Co. एक कृषि प्रौद्योगिकी फर्म है जो फार्म प्रबंधन (farm management) और फसल अनुकूलन (crop optimization) में विशेषज्ञता रखती है। कंपनी फसल रोपण (crop planting) के विवरण को **PLANTINGS** नामक डेटाबेस टेबल (database table) में स्टोर (store) करती है, जिसमें निम्नलिखित एट्रिब्यूट्स (attributes) हैं :

PlantingID, CropType, PlantingDate, FieldLocation.

प्रत्येक हार्वेस्ट (harvest) का विवरण, जिसमें उपज की मात्रा (yield quantity) और गुणवत्ता (quality) शामिल हैं, **HARVESTS** नामक एक अन्य टेबल (table) में संग्रहित किया जाता है, जिसमें निम्नलिखित एट्रिब्यूट्स (attributes) हैं :

HarvestID, PlantingID, YieldQuantity, YieldQuality, HarvestDate.

2

- (I) (क) उस एट्रिब्यूट (attribute) की पहचान कीजिए जिसे दोनों टेबल्स (tables) को जोड़ने के लिए फॉरेन की (FOREIGN KEY) माना जाना चाहिए।

अथवा

- (ख) **HarvestID** एट्रिब्यूट (attribute) पर कौन-सा कंस्ट्रेंट (constraint) लागू किया जाना चाहिए ताकि इसका उपयोग प्रत्येक हार्वेस्ट रिकॉर्ड (harvest record) को अद्वितीय (unique) रूप से पहचानने के लिए किया जा सके ?
- (II) (क) उस रोपण (Planting) के **FieldLocation** की वैल्यू (value) बदलकर 'North Field 3' करने के लिए SQL command लिखिए जिसका **PlantingID** 'PL005' है।

अथवा

- (ख) उन सभी रिकॉर्ड्स के लिए **HARVESTS** टेबल (table) से रिकॉर्ड्स (records) प्रदर्शित करने के लिए SQL command लिखिए जिनकी उपज की गुणवत्ता (**YieldQuality**) उत्कृष्ट (excellent) है।

27. EcoHarvest Solutions Co. is an agricultural technology firm specializing in farm management and crop optimization. The company stores crop planting details in a database table called **PLANTINGS** with attributes such as :

PlantingID, CropType, PlantingDate, FieldLocation.

Details of each harvest, including yield quantity and quality, are stored in another table named **HARVESTS** with attributes like.

HarvestID, PlantingID, YieldQuantity, YieldQuality, HarvestDate.

2

- (I) (a) Identify the attribute which should be considered as the FOREIGN KEY for joining the two tables.

OR

- (b) Which constraint should be applied to the attribute **HarvestID** such that it can be used to uniquely identify every harvest record for all crops ?
- (II) (a) Write the SQL command to change the value of **FieldLocation** of the Planting whose **PlantingID** is 'PL005' to 'North Field 3'.

OR

- (b) Write the SQL command to display the records from the table **HARVESTS** for all records which have excellent **YieldQuality**.

28. (I) (क) वेब सर्वर (web server) शब्द को परिभाषित कीजिए।

(ख) कंप्यूटर नेटवर्क (Computer Networks) में उपयोग की जाने वाली सर्किट स्विचिंग (Circuit Switching) और पैकेट स्विचिंग (Packet Switching) तकनीकों के बीच एक अंतर लिखिए।

2

अथवा

(II) एफटीपी (FTP) और एसएमटीपी (SMTP) के बीच अंतर स्पष्ट कीजिए। कोई दो बिन्दु (point) लिखिए।

खण्ड ग

29. (क) पाइथन (Python) में `Find_Pattern()` नाम से एक यूजर-डिफाईंड (user-defined) फ़ंक्शन (function) लिखिए जो `LOGS.TXT` नामक एक टेक्स्ट (text) फ़ाइल (file) के कंटेंट (content) को पढ़ता है और ऐसी सभी लाइन्स (lines) को प्रदर्शित करता है जो `"ERROR"` शब्द (केस-सेंसिटिव (case-sensitive)) से शुरू होती हैं।

3

उदाहरण के लिए, यदि `LOGS.TXT` फ़ाइल (file) का कंटेंट (content) इस प्रकार है :

`INFO: System startup complete.`

`WARNING: Disk space low on drive C:.`

`ERROR: Failed to connect to database.`

`INFO: User 'admin' logged in.`

`DEBUG: Processing request 'GET /data'.`

`ERROR: Invalid input received from client.`

`SUCCESS: Data saved.`

तो फ़ंक्शन (function) को निम्नलिखित लाइन्स (lines) को प्रदर्शित करना चाहिए :

`ERROR: Failed to connect to database.`

`ERROR: Invalid input received from client.`

अथवा

28. (I) (a) Define the term web server.
- (b) Write one difference between Circuit Switching and Packet Switching techniques used in Computer Networks. 2

OR

- (II) Differentiate between FTP and SMTP. Give any two points.

SECTION C

29. (a) Write a user-defined function in Python named `Find_Pattern()` which reads the contents of a text file named `LOGS.TXT` and displays all such lines that begin with the word "ERROR" (case-sensitive). 3

For example, if the content of file `LOGS.TXT` is as follows :

`INFO: System startup complete.`

`WARNING: Disk space low on drive C:.`

`ERROR: Failed to connect to database.`

`INFO: User 'admin' logged in.`

`DEBUG: Processing request 'GET /data'.`

`ERROR: Invalid input received from client.`

`SUCCESS: Data saved.`

Then the function should display the lines :

`ERROR: Failed to connect to database.`

`ERROR: Invalid input received from client.`

OR

- (ख) पाइथन (Python) में `ContainsChar()` नाम से एक यूजर-डिफाइंड (user-defined) फ़ंक्शन (function) लिखिए जो `DOCUMENT.TXT` नामक एक टेक्स्ट (text) फ़ाइल (file) के कंटेंट (content) को पढ़ता है और ऐसे सभी शब्दों (words) को प्रदर्शित करता है जिनमें 'o' अक्षर (केस-इनसेंसिटिव (case-insensitive)) शामिल है।

उदाहरण के लिए, यदि `DOCUMENT.TXT` फ़ाइल (file) का कंटेंट (content) इस प्रकार है :

```
"A brown fox quickly jumped over the dog. It loves  
to frolic."
```

तो फ़ंक्शन (function) को निम्नलिखित शब्दों (words) को प्रदर्शित करना चाहिए :

```
brown  
fox  
over  
dog.  
loves  
to  
frolic.
```

30. (क) एक ग्लोबली (globally) डिक्लेयर्ड (declared) डिक्शनरी (dictionary) `EMPLOYEE_PERFORMANCE` पर विचार कीजिए जिसमें एम्प्लॉई आईडी (employee IDs) और उनके संबंधित परफॉर्मेंस स्कोर (performance scores) (100 में से) की:वैल्यू (key:value) पेयर्स (pairs) के रूप में मौजूद हैं। उदाहरण के लिए, डिक्शनरी (dictionary) `EMPLOYEE_PERFORMANCE` में निम्नलिखित आइटम्स (items) हो सकते हैं :

$3 \times 1 = 3$

```
EMPLOYEE_PERFORMANCE = {  
    "E101": 85,  
    "E102": 92,  
    "E103": 78,  
    "E104": 95,  
    "E105": 80,  
    "E106": 70  
}
```

- (b) Write a user-defined function in Python named **ContainsChar()** which reads the contents of a text file named **DOCUMENT.TXT** and displays all such words that contain the letter 'o' (case-insensitive).

For example, if the content of file **DOCUMENT.TXT** is as follows :

"A brown fox quickly jumped over the dog. It loves to frolic."

Then the function should display the following words:

brown

fox

over

dog.

loves

to

frolic.

30. (a) Consider a globally declared dictionary named **EMPLOYEE_PERFORMANCE** which contains employee IDs and their respective performance scores (out of 100) as key:value pairs. For example, the dictionary **EMPLOYEE_PERFORMANCE** may contain items as follows :

$3 \times 1 = 3$

```
EMPLOYEE_PERFORMANCE = {  
    "E101": 85,  
    "E102": 92,  
    "E103": 78,  
    "E104": 95,  
    "E105": 80,  
    "E106": 70  
}
```

TOP_PERFORMERS_LIST नामक एक स्टैक (stack) को इम्प्लीमेंट (implement) करने वाली ग्लोबली (globally) डिक्लेयर्ड (declared) लिस्ट (list) पर दिए गए निम्नलिखित ऑपरेशन्स (operations) को करने के लिए निम्नलिखित यूजर-डिफाईंड (user-defined) फ़ंक्शंस (functions) लिखिए :

- (I) **PushHighPerformers()** : इस फ़ंक्शन (function) को डिक्शनरी (dictionary) **EMPLOYEE_PERFORMANCE** से उन सभी एम्प्लॉई आईडी (employee IDs) को स्टैक (stack) **TOP_PERFORMERS_LIST** में पुश (push) करना चाहिए जिनका परफॉर्मेंस स्कोर (performance score) 90 या उससे अधिक है।

उदाहरण के लिए, **PushHighPerformers()** फ़ंक्शन (function) के निष्पादन (execution) पर, स्टैक (stack) **TOP_PERFORMERS_LIST** में, डिक्शनरी (dictionary) **EMPLOYEE_PERFORMANCE** के ऊपर दिए गए उदाहरण से, **["E102", "E104"]** होने चाहिए (यह मानते हुए कि इंसर्शन (insertion) का क्रम पहले **E102** और फिर **E104** है)।

- (II) **IsPerformanceListEmpty()** : इस फ़ंक्शन (function) को यह जाँचना चाहिए कि स्टैक (stack) **TOP_PERFORMERS_LIST** खाली (empty) है या नहीं। यदि यह खाली (empty) है, तो फ़ंक्शन (function) को **True** रिटर्न (return) करना चाहिए; अन्यथा, इसे **False** रिटर्न (return) करना चाहिए।

- (III) **PopPerformer()** : इस फ़ंक्शन (function) को स्टैक (stack) **TOP_PERFORMERS_LIST** से सबसे ऊपरी एम्प्लॉई आईडी (employee ID) को पॉप (pop) और रिटर्न (return) करना चाहिए। यदि स्टैक (stack) पहले से ही खाली (empty) है (इसे **IsPerformanceListEmpty()** फ़ंक्शन (function) का उपयोग करके जाँचिए), तो फ़ंक्शन (function) को संदेश (message) **"Stack Underflow: No top performer to pop."** प्रदर्शित करना चाहिए।

अथवा

Write the following user-defined functions to perform the following given operations on a globally declared list implementing a stack named **TOP_PERFORMERS_LIST** :

- (I) **PushHighPerformers()** : This function should push all employee IDs from the dictionary **EMPLOYEE_PERFORMANCE** whose performance score is greater than or equal to 90 into the stack **TOP_PERFORMERS_LIST**. For example, on execution of the **PushHighPerformers()** function, the stack **TOP_PERFORMERS_LIST** should contain ["E102", "E104"] (assuming the order of insertion is E102 then E104) from the above considered example of the dictionary **EMPLOYEE_PERFORMANCE**.
- (II) **IsPerformanceListEmpty()** : This function should check whether the stack **TOP_PERFORMERS_LIST** is empty or not. If it is empty, the function should return **True**; otherwise, it should return **False**.
- (III) **PopPerformer()** : This function should pop and return the topmost employee ID from the stack **TOP_PERFORMERS_LIST**. If the stack is already empty (check using the **IsPerformanceListEmpty()** function), then the function should display the message "**Stack Underflow: No top performer to pop.**"

OR

- (ख) **PRODUCT_PRICES** नामक प्रोडक्ट (product) कीमतों (prices) की एक ग्लोबली (globally) डिक्लेयर्ड (declared) लिस्ट (list) पर विचार कीजिए जो इस प्रकार है :

```
PRODUCT_PRICES = [ 150.75, 85.00, 299.99, 50.50, 120.00, 99.95, 450.00 ]
```

DISCOUNT_PRICES नामक एक स्टैक (stack) को इम्प्लीमेंट (implement) करने वाली ग्लोबली (globally) डिक्लेयर्ड (declared) लिस्ट (list) पर दिए गए निम्नलिखित ऑपरेशन्स (operations) को करने के लिए निम्नलिखित यूजर-डिफाइंड (user-defined) फ़ंक्शंस (functions) लिखिए :

- (I) **PushDiscPrices()** : इस फ़ंक्शन (function) को लिस्ट (list) **PRODUCT_PRICES** से उन सभी कीमतों (prices) को स्टैक (stack) **DISCOUNT_PRICES** में पुश (push) करना चाहिए जो **100.00** या उससे कम हैं। उदाहरण के लिए, **PushDiscPrices()** फ़ंक्शन (function) के निष्पादन (execution) पर, स्टैक (stack) **DISCOUNT_PRICES** में लिस्ट (list) **PRODUCT_PRICES** के ऊपर दिए गए उदाहरण से **[85.00, 50.50, 99.95]** होने चाहिए (यह मानते हुए कि पुशिंग (pushing) का यह क्रम **PRODUCT_PRICES** लिस्ट (list) में उनकी उपस्थिति के आधार पर है)।
- (II) **PopDiscPrices()** : इस फ़ंक्शन (function) को स्टैक (stack) **DISCOUNT_PRICES** से अंतिम (सबसे ऊपरी) कीमत (price) को पॉप (pop) और प्रदर्शित (display) करना चाहिए। यदि स्टैक (stack) पहले से ही खाली (empty) है, तो फ़ंक्शन (function) को संदेश (message) **"No More Discount Prices to Pop"** प्रदर्शित करना चाहिए और **None** रिटर्न (return) करना चाहिए।
- (III) **ShowDiscPrices()** : इस फ़ंक्शन (function) को स्टैक (stack) **DISCOUNT_PRICES** से किसी भी कीमत (price) को हटाए बिना सभी कीमतों (prices) को प्रदर्शित करना चाहिए। **DISCOUNT_PRICES** का कंटेंट (content) स्टैक (stack) के लीफो (LIFO) नियम के अनुसार प्रदर्शित किया जाना चाहिए (यानी, अंतिम कीमत (price) से पहली कीमत (price) तक)। यदि स्टैक (stack) में कोई कीमत (price) नहीं है, तो संदेश (message) **"Stack is Empty: No discount prices to show"** प्रदर्शित कीजिए।

- (b) Consider a globally declared list of product prices named **PRODUCT_PRICES** as follows :

```
PRODUCT_PRICES = [ 150.75, 85.00, 299.99, 50.50,  
120.00, 99.95, 450.00 ]
```

Write the following user-defined functions to perform the following given operations on a globally declared list implementing a stack named **DISCOUNT_PRICES** :

- (I) **PushDiscPrices()** : This function should push all those prices from the list **PRODUCT_PRICES** which are less than or equal to **100.00** into the stack **DISCOUNT_PRICES**. For example, on execution of the **PushDiscPrices()** function, the stack **DISCOUNT_PRICES** should contain **[85.00, 50.50, 99.95]** (assuming this order of pushing is based on appearance in the **PRODUCT_PRICES** list) from the above considered example of the list **PRODUCT_PRICES**.
- (II) **PopDiscPrices()** : This function should pop and display the last (topmost) price from the stack **DISCOUNT_PRICES**. If the stack is already empty, then the function should display the message **"No More Discount Prices to Pop"** and return **None**.
- (III) **ShowDiscPrices()** : This function should display all the prices from the stack **DISCOUNT_PRICES** without deleting any price from the stack. The content of **DISCOUNT_PRICES** should be displayed as per stack LIFO rule (i.e., last price to first price). If there are no prices in the stack, then display the message **"Stack is Empty: No discount prices to show"**.

31. (क) निम्नलिखित पाइथन (Python) कोड (code) के निष्पादन (execution) पर आउटपुट (output) का अनुमान लगाइए :

```
SCORES = [5, 12, 7, 20, 10, 18]
for X in range(len(SCORES)):
    if SCORES[X] % 4 == 0:
        SCORES[X] = SCORES[X] + SCORES[X-1]
    elif SCORES[X] % 3 == 0:
        SCORES[X] = SCORES[len(SCORES) - 1 - X] + 5
    else:
        SCORES[X] = SCORES[X] // 2
for Y in range(0, len(SCORES), 2):
    if Y + 1 < len(SCORES):
        print(SCORES[Y], SCORES[Y+1], sep="$")
    else:
        print(SCORES[Y], sep="$")
```

अथवा

- (ख) निम्नलिखित पाइथन (Python) कोड (code) के निष्पादन (execution) पर आउटपुट (output) का अनुमान लगाइए :

```
def Transform(D_A, D_B):
    D_C = {}
    D_C.update(D_A)
    D_A.update(D_B)
    D_B.update(D_C)
    return D_C

Init_Stock = {"Chocolate": 100, "Cookie": 50}
New_Arriv = {"Cookie": 60, "Chocolate": 120}
T_Data = Transform(Init_Stock, New_Arriv)
print(Init_Stock)
print(New_Arriv)
print(T_Data)
```

31. (a) Predict the output on execution of the following Python code : 3

```
SCORES = [5, 12, 7, 20, 10, 18]
for X in range(len(SCORES)):
    if SCORES[X] % 4 == 0:
        SCORES[X] = SCORES[X] + SCORES[X-1]
    elif SCORES[X] % 3 == 0:
        SCORES[X] = SCORES[len(SCORES) - 1 - X] + 5
    else:
        SCORES[X] = SCORES[X] // 2
for Y in range(0, len(SCORES), 2):
    if Y + 1 < len(SCORES):
        print(SCORES[Y], SCORES[Y+1], sep="$")
    else:
        print(SCORES[Y], sep="$")
```

OR

- (b) Predict the output on execution of the following Python code :

```
def Transform(D_A, D_B):
    D_C = {}
    D_C.update(D_A)
    D_A.update(D_B)
    D_B.update(D_C)
    return D_C

Init_Stock = {"Chocolate": 100, "Cookie": 50}
New_Arriv = {"Cookie": 60, "Chocolate": 120}
T_Data = Transform(Init_Stock, New_Arriv)
print(Init_Stock)
print(New_Arriv)
print(T_Data)
```

खण्ड घ

32. **TechMart Inc.**, **PRODUCTS** नामक एक SQL टेबल (table) का उपयोग करके अपने प्रोडक्ट इन्वेंट्री (product inventory) को प्रबंधित करता है। टेबल (table) में एट्रिब्यूट्स (attributes) **PID**, **PName**, **Category**, **Price**, **Quantity**, और **LastUpdated** शामिल हैं, और इसमें निम्नलिखित रिकॉर्ड्स (records) शामिल हैं :

Table : PRODUCTS

PID	PName	Category	Price	Quantity	LastUpdated
101	Laptop Pro	Electronics	1200	50	2024-01-10
102	Gaming Mouse	Peripherals	45.99	200	2023-11-25
103	Ergonomic Keyboard	Peripherals	75	150	2024-02-01
104	Smart Speaker	Electronics	89.99	100	2024-01-15
105	USB-C Hub	Accessories	29.5	300	2023-12-05
106	External SSD 1TB	Storage	99	75	2024-03-01
107	Wireless Earbuds	Audio	129.99	120	2024-02-10
108	Desk Lamp	Home	35	180	2023-10-20
109	Web Camera HD	Peripherals	60	NULL	2024-01-25
110	Portable Charger	Accessories	49.99	250	2023-12-15

SECTION D

32. TechMart Inc. manages its product inventory using an SQL table named **PRODUCTS**. The table consists of the attributes **PID**, **PName**, **Category**, **Price**, **Quantity**, and **LastUpdated**, and contains the following records:

Table : PRODUCTS

PID	PName	Category	Price	Quantity	LastUpdated
101	Laptop Pro	Electronics	1200	50	2024-01-10
102	Gaming Mouse	Peripherals	45.99	200	2023-11-25
103	Ergonomic Keyboard	Peripherals	75	150	2024-02-01
104	Smart Speaker	Electronics	89.99	100	2024-01-15
105	USB-C Hub	Accessories	29.5	300	2023-12-05
106	External SSD 1TB	Storage	99	75	2024-03-01
107	Wireless Earbuds	Audio	129.99	120	2024-02-10
108	Desk Lamp	Home	35	180	2023-10-20
109	Web Camera HD	Peripherals	60	NULL	2024-01-25
110	Portable Charger	Accessories	49.99	250	2023-12-15

(क) ऊपर दिए गए डेटा (data) के आधार पर निम्नलिखित प्रश्नों के उत्तर दीजिए :

4×1=4

- (I) उन **PRODUCTS** के **PName** और **Price** को प्रदर्शित करने के लिए SQL स्टेटमेंट (statement) लिखिए जो '**Electronics**' कैटेगरी (category) से संबंधित हैं और जिनका **Price 100.00** से अधिक है।
- (II) **Category** और प्रत्येक कैटेगरी (category) में प्रोडक्ट्स (products) की कुल संख्या को प्रदर्शित करने के लिए SQL स्टेटमेंट (statement) लिखिए।
- (III) अपडेट होने की अंतिम तिथि (**Last date of Updation**) के आरोही क्रम (ascending order) में (सबसे पहले के अपडेट्स को पहले दिखाते हुए) पूरे **PRODUCTS** टेबल (table) के कंटेंट (content) को प्रदर्शित करने के लिए SQL स्टेटमेंट (statement) लिखिए।
- (IV) उन सभी **PRODUCTS** के **Price** को 0 करने के लिए SQL स्टेटमेंट (statement) लिखिए जिनकी मात्रा (quantity) ज्ञात नहीं है।

अथवा

(ख) ऊपर दी गई **PRODUCTS** टेबल (table) पर विचार करते हुए, निम्नलिखित SQL कमांड्स (commands) के execution पर आउटपुट (output) लिखिए :

- (I) **SELECT Avg(Quantity) FROM PRODUCTS**
WHERE Category = 'Peripherals';
- (II) **SELECT DISTINCT Category FROM PRODUCTS;**
- (III) **SELECT COUNT(PID) FROM PRODUCTS**
WHERE Quantity < 100;
- (IV) **SELECT PName, Category FROM PRODUCTS**
WHERE PName LIKE '%Speaker%';

(a) Based on the data given above answer the following questions : $4 \times 1 = 4$

- (I) Write the SQL statement to display the **PName** and **Price** of those **PRODUCTS** that belong to the '**Electronics**' category and have a Price greater than 100.00.
- (II) Write the SQL statement to display the **Category** and the total number of products in each category.
- (III) Write the SQL statement to display the entire **PRODUCTS** table content, in ascending order of **Last date of Updation** (showing the earliest updates first).
- (IV) Write the SQL statement to make the **Price** of all **PRODUCTS** whose quantity is not known, to 0.

OR

(b) Considering the above given table **PRODUCTS**, write the output on execution of the following SQL commands:

- (I) **SELECT Avg(Quantity) FROM PRODUCTS**
WHERE Category = 'Peripherals';
- (II) **SELECT DISTINCT Category FROM PRODUCTS;**
- (III) **SELECT COUNT(PID) FROM PRODUCTS**
WHERE Quantity < 100;
- (IV) **SELECT PName, Category FROM PRODUCTS**
WHERE PName LIKE '%Speaker%';

33. "ORDERS.CSV" नामक एक सीएसवी (CSV) फ़ाइल (file) में एक ई-कॉमर्स (e-commerce) प्लेटफॉर्म पर दिए गए कस्टमर ऑर्डर्स (customer orders) के रिकॉर्ड्स (records) शामिल हैं। फ़ाइल (file) के प्रत्येक रिकॉर्ड (record) में निम्नलिखित के लिए डेटा (data) शामिल है : $2+2=4$

[OrderID, CustomerID, OrderDate, TotalAmount, OrderStatus]

संदर्भ के लिए सीएसवी (CSV) फ़ाइल (file) की एक नमूना पंक्ति (row) नीचे दी गई है :

['ORD001', 'C101', '2024-05-20', 150.75, 'Shipped']

सीएसवी (CSV) फ़ाइल (file) के साथ निर्दिष्ट कार्यों को करने के लिए निम्नलिखित पाइथन (Python) फ़ंक्शन (function) डेफिनिशन्स (definitions) लिखिए :

- (i) **RecentOrders()** : इस फ़ंक्शन (function) को सीएसवी (CSV) फ़ाइल (file) से पंक्तियों (rows) को पढ़ना चाहिए और वर्ष '2024' में दिए गए सभी ऑर्डर्स (orders) के लिए OrderID और TotalAmount प्रदर्शित करना चाहिए।
- (ii) **AddOrder(newOrder)** : इस फ़ंक्शन को CSV फ़ाइल के अंत में डेटा लिखना चाहिए। रिकॉर्ड को फ़ंक्शन में एक आर्गुमेंट के रूप में पास किया जाता है।

34. **LIBRARY_CATALOG** नामक एक डेटाबेस (database) पर विचार कीजिए जिसमें नीचे दी गई दो टेबल्स (tables) – **PUBLISHERS** और **BOOKS** शामिल हैं :

Table : PUBLISHERS

PublID	Name	Country	Genre
101	Grand Books	USA	Fiction
102	World Lit Press	UK	Non-Fiction
103	Tech Tome	Germany	Sci-Fi
104	Story Weaver	USA	Fiction
105	Global Insights	Canada	Non-Fiction

33. A CSV file named "ORDERS.CSV" contains records of customer orders placed on an e-commerce platform. The structure of each record is as follows : 2+2=4

[OrderID, CustomerID, OrderDate, TotalAmount, OrderStatus]

A sample row for the CSV file is given below for reference :

['ORD001', 'C101', '2024-05-20', 150.75, 'Shipped']

Write the following Python function definitions to perform the specified tasks with the CSV file :

- (i) **RecentOrders()** : This function should read the rows from the CSV file and display the **OrderID** and **TotalAmount** for all orders placed in the year '2024'.
- (ii) **AddOrder(newOrder)** : This function should write data at the end of the CSV file. The record is passed as argument to the function.

34. Consider a database **LIBRARY_CATALOG** containing the following two tables – **PUBLISHERS** and **BOOKS** given below :

Table : PUBLISHERS

PublID	Name	Country	Genre
101	Grand Books	USA	Fiction
102	World Lit Press	UK	Non-Fiction
103	Tech Tome	Germany	Sci-Fi
104	Story Weaver	USA	Fiction
105	Global Insights	Canada	Non-Fiction

Table : BOOKS

BookID	Title	PublID	PublYear	Pages
3001	The Last Horizon	101	2018	450
3002	Echoes of Tomorrow	103	2020	320
3003	A Brief History	102	2015	580
3004	Whispers in the Wind	101	2019	280
3005	Starship Odyssey	103	2021	400
3006	The Silent Shore	104	2022	350
3007	Understanding AI	103	2019	290

नोट (Note) : **BOOKS** टेबल (table) में एट्रिब्यूट (attribute) **Pages** एक पुस्तक में कुल पृष्ठों (pages) की संख्या को दर्शाता है, **PublID** पब्लिशर के आइडेंटिफिकेशन नंबर (Identification Number) को दर्शाता है, और **PublYear** पब्लिकेशन के वर्ष (Year of Publication) को दर्शाता है।

निम्नलिखित के लिए SQL क्वेरीज (queries) लिखिए :

$4 \times 1 = 4$

- (I) (क) इक्वी जॉइन (Equi Join) के माध्यम से उन सभी **BOOKS** के **Title** और उनके संबंधित **PublYear** को प्रदर्शित करने के लिए जो '**Fiction**' प्रकार के जॉनर (**Genre**) (**PUBLISHERS** टेबल (table) से) से संबंधित हैं।

अथवा

- (ख) नेचुरल जॉइन (Natural Join) के माध्यम से उन सभी **BOOKS** के **Title** और उनके संबंधित **PublYear** को प्रदर्शित करने के लिए जो '**Fiction**' प्रकार के जॉनर (**Genre**) (**PUBLISHERS** टेबल (table) से) से संबंधित हैं।

Table : BOOKS

BookID	Title	PublID	PublYear	Pages
3001	The Last Horizon	101	2018	450
3002	Echoes of Tomorrow	103	2020	320
3003	A Brief History	102	2015	580
3004	Whispers in the Wind	101	2019	280
3005	Starship Odyssey	103	2021	400
3006	The Silent Shore	104	2022	350
3007	Understanding AI	103	2019	290

Note : The attribute **Pages** in the table **BOOKS** represents the total number of pages in a book, **PublID** represents the Publisher's Identification Number, **PublYear** represents the Year of Publication.

Write the SQL queries for the following :

4×1=4

- (I) (a) To display **Title** of all **BOOKS** and their corresponding **PublYear** which belong to the '**Fiction**' type of **Genre** (from **PUBLISHERS** table) through Equi Join.

OR

- (b) To display **Title** of all **BOOKS** and their corresponding **PublYear** which belong to the '**Fiction**' type of **Genre** (from **PUBLISHERS** table) through Natural Join.

- (II) सभी **PUBLISHERS** के **Name** और उनके संबंधित **BOOKS** के **Title** को **PublYear** के आरोही क्रम (ascending order) में प्रदर्शित करने के लिए।
- (III) **BOOKS** टेबल (table) से उन सभी रिकॉर्ड्स (records) को डिलीट (delete) करने के लिए जिनका **PublYear** 2020 से पुराना है।
- (IV) डेटाबेस (database) **LIBRARY_CATALOG** को खोलने (open करने) के लिए और फिर डेटाबेस (database) में मौजूद सभी टेबल्स (tables) के नामों को प्रदर्शित करने के लिए।

35. INVESTMENT डेटाबेस (database) में **EQUITY** नामक एक टेबल (table) का स्ट्रक्चर (structure) निम्नलिखित है :

4

Field	Type	Remarks
Code	Char(6)	Code of stock
Name	Varchar(15)	Name of stock
Exchange	Char(3)	Name of exchange
LTP	Float	Last traded Price
Volume	int	Amount of stock

EQUITY टेबल (table) से उन सभी रिकॉर्ड्स (records) को प्रदर्शित करने के लिए पाइथन (Python) में एक फ़ंक्शन (function) लिखिए जहाँ **Exchange** का मान **BSE** है। साथ ही ऐसे रिकॉर्ड्स (records) की कुल संख्या को भी प्रदर्शित कीजिए।

पाइथन (Python) डेटाबेस (database) कनेक्टिविटी (connectivity) के लिए निम्नलिखित मानों को मान लीजिए :

- Host – localhost
- User – Finman
- Password – Stk2026

- (II) To display **Name** of all **PUBLISHERS** and their corresponding **Title** of **BOOKS** in ascending order of **PublYear**.
- (III) To delete all records from the table **BOOKS** whose **PublYear** is older than **2020**.
- (IV) To open the database **LIBRARY_CATALOG** and then display the names of all tables present in the database.

35. A table named **EQUITY** in **INVESTMENT** database has the following structure :

4

Field	Type	Remarks
Code	Char(6)	Code of stock
Name	Varchar(15)	Name of stock
Exchange	Char(3)	Name of exchange
LTP	Float	Last traded Price
Volume	int	Amount of stock

Write a function in Python to display all the records where Exchange is **BSE** from the **EQUITY** table. Also display the total number of such records.

Assume the following for Python database connectivity :

- Host – localhost
- User – Finman
- Password – Stk2026

खण्ड ड

36. एक फ़ाइल (file) **CLINICAL_TRIALS.DAT** निम्नलिखित स्ट्रक्चर (structure) के रिकॉर्ड्स (records) को स्टोर (store) करती है :

[**TRIAL_ID**, **PATIENT_NAME**, **DRUG_TESTED**, **DOSAGE_MG**,
OUTCOME_SCORE]

जहाँ :

- **TRIAL_ID** - क्लीनिकल ट्रायल (clinical trial) के प्रतिभागी के लिए यूनिक (unique) आइडेंटिफायर (identifier) (स्ट्रिंग (string) टाइप (type))
- **PATIENT_NAME** मरीज का नाम (स्ट्रिंग (string) टाइप (type))
- **DRUG_TESTED** - दी गई दवा (drug) का नाम (स्ट्रिंग (string) टाइप (type)), जैसे, "DrugX", "DrugY", "Placebo")
- **DOSAGE_MG** - मिलीग्राम (milligrams) में दवा की खुराक (डोजेज) (इंटीजर (integer) टाइप (type))
- **OUTCOME_SCORE** - उपचार के परिणाम को दर्शाने वाला संख्यात्मक स्कोर (फ्लोट (float) टाइप (type)), जहाँ अधिक स्कोर का मतलब बेहतर परिणाम है)

निम्नलिखित कार्यों को करने के लिए पाइथन (Python) में यूज़र-डिफाइंड (user-defined) फ़ंक्शंस (functions) लिखिए :

2+3=5

- (I) **RecordTrialData()** : इस फ़ंक्शन (function) को क्लीनिकल ट्रायल (clinical trial) के प्रतिभागियों का डेटा (data) इनपुट (input) लेना चाहिए और उसे बाइनरी (binary) फ़ाइल (file) **CLINICAL_TRIALS.DAT** में लिखना चाहिए।
- (II) **AdjustLowDosage()** : इस फ़ंक्शन (function) को उन सभी प्रतिभागियों के **DOSAGE_MG** को 20% बढ़ाना चाहिए जिनका **OUTCOME_SCORE** 0.5 से कम है, और अपडेटेड (updated) रिकॉर्ड्स (records) को वापस फ़ाइल (file) **CLINICAL_TRIALS.DAT** में फिर से लिखना (rewrite) चाहिए।

SECTION E

36. A file `CLINICAL_TRIALS.DAT` stores records of the following structure :

[`TRIAL_ID`, `PATIENT_NAME`, `DRUG_TESTED`, `DOSAGE_MG`,
`OUTCOME_SCORE`]

Where :

- `TRIAL_ID` - Unique identifier for the clinical trial participant (string type)
- `PATIENT_NAME` - Name of the patient (string type)
- `DRUG_TESTED` - Name of the drug administered (string type, e.g., "DrugX", "DrugY", "Placebo")
- `DOSAGE_MG` - Dosage of the drug in milligrams (integer type)
- `OUTCOME_SCORE` - Numerical score indicating the treatment outcome (float type, higher score means better outcome)

Write user-defined functions in Python for the following tasks : 2+3=5

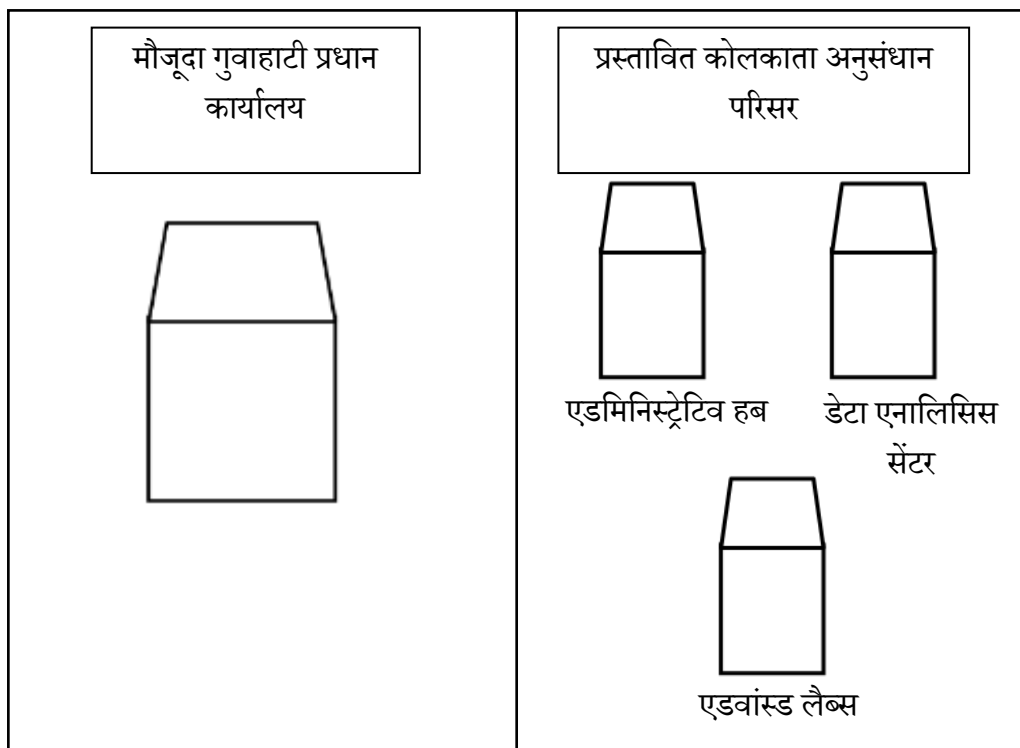
- (I) `RecordTrialData()`: This function should input data for clinical trial participants and write it to the binary file `CLINICAL_TRIALS.DAT`.
- (II) `AdjustLowDosage()`: This function should increase the `DOSAGE_MG` of all participants whose `OUTCOME_SCORE` is below 0.5 by 20% and rewrite the updated records into the file `CLINICAL_TRIALS.DAT`.

37. आप गुवाहाटी, असम में स्थित मुख्यालय (head office) वाले एक तेजी से विस्तार कर रहे कृषि अनुसंधान और विकास संगठन “द ग्रीन स्प्राउट इंस्टीट्यूट” (The Green Sprout Institute) के लीड (lead) नेटवर्क (network) आर्किटेक्ट (architect) हैं। द ग्रीन स्प्राउट इंस्टीट्यूट का एक महत्वपूर्ण विस्तार हो रहा है, जिसमें निम्नलिखित शामिल हैं : 5×1=5

- कोलकाता में एक नया, बड़ा केंद्रीय अनुसंधान परिसर (central research campus) प्राप्त (acquire) करना, जिसमें उन्नत प्रयोगशाला उपकरण और एक बड़ा प्रशासनिक एवं अनुसंधान स्टाफ (staff) होगा।
- असम के विभिन्न कृषि क्षेत्रों में फैले चार छोटे रिमोट (remote) फील्ड (field) टेस्टिंग (testing) स्टेशनों (stations) को एकीकृत (integrate) करना : ये स्टेशन मुख्य रूप से ऑटोमेटेड सेंसर डेटा कलेक्शन (automated sensor data collection) पर ध्यान केंद्रित करेंगे और डेटा (data) को कोलकाता के केंद्रीय अनुसंधान परिसर (central research campus) में सुरक्षित रूप से ट्रांसफर (transfer) करेंगे।
- नए कोलकाता रिसर्च कैंपस (Kolkata Research Campus) में तीन अलग-अलग सेक्शन्स/ब्लॉक्स (sections/blocks) शामिल होंगे। एडवांस्ड लैब्स (Advanced Labs), डेटा एनालिसिस सेंटर (Data Analysis Centre), और एडमिनिस्ट्रेटिव हब (Administrative Hub), जैसा कि नीचे दिए गए कॉन्सेप्टुअल (conceptual) लेआउट (layout) में दिखाया गया है :

37. You are the lead network architect for “The Green Sprout Institute,” a rapidly expanding agricultural research and development organization with its head office located in Guwahati, Assam. The Green Sprout Institute is undergoing significant expansion, which includes : $5 \times 1 = 5$

- Acquiring a new, larger central research campus in Kolkata, to house advanced laboratory equipment and a large administrative and research staff.
- Integrating four smaller remote field-testing stations across various agricultural regions in Assam : These stations will primarily focus on automated sensor data collection and transfer the data securely to the central research campus in Kolkata.
- The new Kolkata Research Campus will comprise of three distinct sections/blocks : Advanced Labs, Data Analysis Centre, and Administrative Hub, as shown in the conceptual layout below :

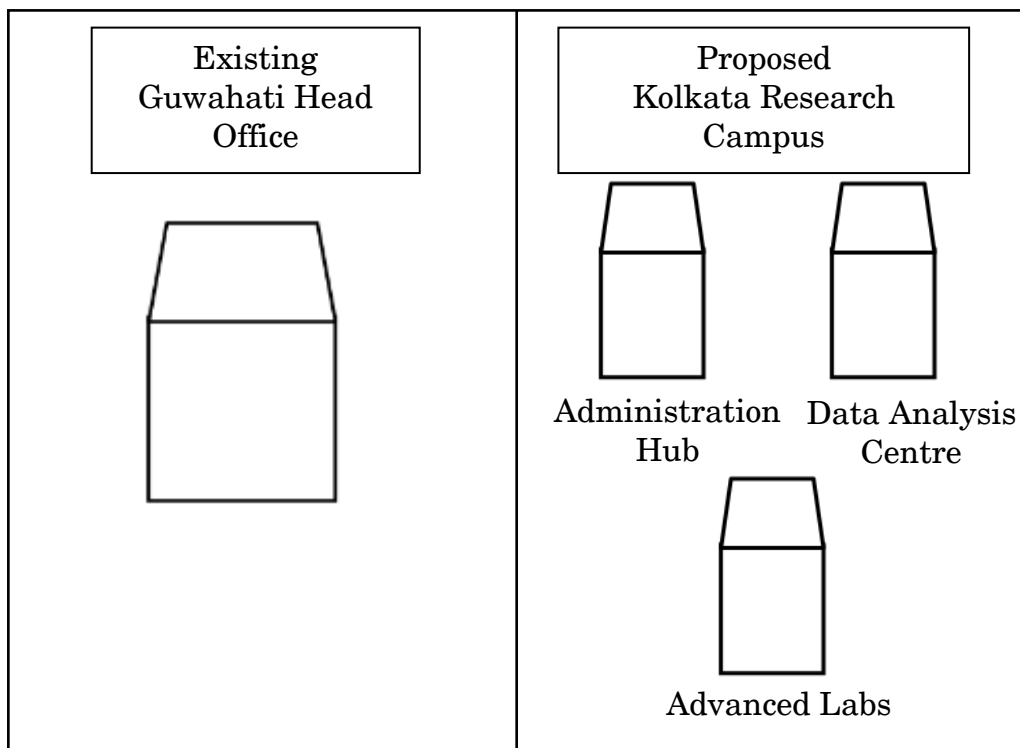


प्रस्तावित कोलकाता परिसर सेक्शन्स (sections) के बीच की दूरी :

सेक्शन से सेक्शन	दूरी
एडवांस्ड लैब्स से डेटा एनालिसिस सेंटर	280 मीटर
डेटा एनालिसिस सेंटर से एडमिनिस्ट्रेटिव हब	320 मीटर
एडमिनिस्ट्रेटिव हब से एडवांस्ड लैब्स	250 मीटर

- कोलकाता रिसर्च कैंपस (Kolkata Research Campus) के प्रत्येक सेक्शन (section) में कम्प्यूटर्स/डिवाइसिज़ (computers/devices) की संख्या निम्नलिखित होगी :

सेक्शन	कम्प्यूटर्स/डिवाइसिज़ (computers/devices) की संख्या
एडवांस्ड लैब्स	100
डेटा एनालिसिस सेंटर	60
एडमिनिस्ट्रेटिव हब	150



Distance between the sections of Proposed Kolkata Research Campus :

Section to Section	Distance
Advanced Labs to Data Analysis Centre	280 metres
Data Analysis Centre to Administrative Hub	320 metres
Administrative Hub to Advanced Labs	250 metres

- Each section in the Kolkata Research Campus will have the following number of computers/devices :

Section	Number of computers/devices
Advanced Labs	100
Data Analysis Centre	60
Administrative Hub	150

लीड (lead) नेटवर्क (network) आर्किटेक्ट (architect) के रूप में, आपको एक व्यापक नेटवर्क (network) डिज़ाइन (design) और कार्यान्वयन योजना (implementation plan) का प्रस्ताव देना होगा जो द ग्रीन स्प्राउट इंस्टीट्यूट (The Green Sprout Institute) के विस्तार लक्ष्यों के सभी प्रमुख मुद्दों का समाधान करती हो। इस संबंध में नीचे दिए गए प्रश्नों के उत्तर दीजिए :

- (I) कोलकाता रिसर्च कैम्पस (Kolkata Research Campus) के प्रत्येक सेक्शन (section) में उस सेक्शन के व्यक्तिगत कम्प्यूटर्स/डिवाइसिज़ (computers/devices) को जोड़ने वाले केबल्स (cables) को कनेक्ट (connect) करने के लिए स्थापित किए जाने वाले किसी डिवाइस (device) का सुझाव दीजिए।
- (II) महत्वपूर्ण अनुसंधान डेटा (data) के लिए सर्वोत्तम और सबसे प्रभावी कनेक्टिविटी (connectivity) और सुरक्षा प्राप्त करने के लिए कोलकाता कैम्पस (Kolkata campus) में सर्वर (SERVER) स्थापित करने के लिए सबसे उपयुक्त स्थान (सेक्शन (section)) का सुझाव दीजिए (3 सेक्शन्स (sections) में से)। अपने उत्तर का औचित्य सिद्ध (justify) कीजिए।
- (III) कोलकाता कैम्पस (Kolkata Campus) के भीतर विभिन्न सेक्शन्स (sections) को क़िफ़ायती और विश्वसनीय तरीके से जोड़ने के लिए केबल (cable) लेआउट (layout) का चित्र बनाइए।
- (IV) जब गुवाहाटी के केंद्रीय मुख्यालय (Guwahati Head Office) को नए कोलकाता रिसर्च कैम्पस (Kolkata Research Campus) से जोड़ा जाएगा, तो PAN, LAN, MAN, WAN में से किस प्रकार का नेटवर्क (network) स्थापित होगा ?
- (V) (क) बड़े डेटासेट (datasets) और अनुसंधान फ़ाइल्स (files) को ट्रांसफर (transfer) करने के लिए गुवाहाटी मुख्यालय (Guwahati Head Office) को कोलकाता रिसर्च कैम्पस (Kolkata Research Campus) से जोड़ने के लिए किस सुरक्षित और उच्च-बैंडविड्थ (high-bandwidth) वाले वायर्ड ट्रांसमिशन मीडियम (wired transmission medium) का उपयोग किया जाना चाहिए ?

अथवा

- (ख) यदि गुवाहाटी मुख्यालय (Guwahati HQ) के शोधकर्ताओं को कोलकाता अनुसंधान केंद्र (Kolkata Research centre) के साथ वीडियो कॉन्फ़रेंसिंग (video conferencing) का उपयोग करके सहयोगात्मक अनुसंधान सत्रों (Collaborative Research Sessions) की आवश्यकता है, तो किस विशिष्ट प्रोटोकॉल (protocol) का उपयोग किया जाएगा ?

As the lead network architect, you need to propose a comprehensive network design and implementation plan that addresses all the key issues of The Green Sprout Institute's expansion goals. Answer the questions given below in this respect :

- (I) Suggest a device to be installed in each of the sections of the Kolkata Research Campus to connect the cables of individual computers/devices within that section.
- (II) Suggest the most appropriate location (Section) to install the SERVER in the Kolkata campus (out of the 3 sections) to get the best and most effective connectivity and security for critical research data. Justify your answer.
- (III) Draw the cable layout to reliably connect various sections within the Kolkata Campus.
- (IV) What type of network out of PAN, LAN, MAN, WAN will be set up when the central Guwahati Head Office is connected with the new Kolkata Research Campus ?
- (V) (a) Which secure and high-bandwidth wired transmission medium should preferably be used to connect the Guwahati Head Office with the Kolkata Research Campus for transferring large datasets and research files ?

OR

- (b) Which specific protocol will be used if researchers at the Guwahati HQ need to have Collaborative Research Sessions using video conferencing with Kolkata Research Centre ?