

Subject Name (Gujarati)

4331603 -- Summer 2025

Semester 1 Study Material

Detailed Solutions and Explanations

પ્રશ્ન 1(a) [3 ગુણ]

નીચેના શબ્દોની વ્યાખ્યા આપો. 1) મેટાડેટા 2) સ્કીમા 3) ડેટા ડિક્શનરી.

જવાબ

ટેબલ:

શબ્દ	વ્યાખ્યા
મેટાડેટા	ડેટા વિશેનો ડેટા જે ડેટાબેઝની રચના અને વિશેષતાઓ વર્ણવે છે
સ્કીમા	ડેટાબેઝના સંગઠન અને સંબંધોને દર્શાવતી તાર્કિક રચના
ડેટા ડિક્શનરી	ડેટાબેઝના તત્વો વિશેની માહિતી સંગ્રહિત કરતો કેન્દ્રીય ભંડાર

- મેટાડેટા: ડેટાની લાક્ષણિકતાઓ અને ગુણધર્મો વર્ણવતી માહિતી
- સ્કીમા: ડેટાબેઝની રચના અને મર્યાદાઓ વ્યાખ્યાયિત કરતો બ્લુપ્રિન્ટ
- ડેટા ડિક્શનરી: બધા ડેટાબેઝ ઓબ્જેક્ટ્સ અને તેમના ગુણધર્મોની કેટલોગ

મેમરી ટ્રીક

“MSD - My System Dictionary”

પ્રશ્ન 1(b) [4 ગુણ]

ડેટાબેઝ મેનેજમેન્ટ સિસ્ટમના ફાયદા લખો.

જવાબ

ટેબલ:

ફાયદો	વર્ણન
ડેટા સ્વતંત્રતા	એપ્લિકેશન્સ ડેટા સ્ટોરેજથી સ્વતંત્ર
ડેટા અખંડિતતા	ડેટાની ચોકસાઈ અને સુસંગતતા જાળવે છે
સુરક્ષા નિયંત્રણ	વપરાશકર્તા પ્રમાણીકરણ અને અધિકરણ
સમવર્તી પહોંચ	અનેક વપરાશકર્તાઓ એકસાથે પહોંચ કરી શકે છે

- ઘટેલી રીડ-ડ-સી: ડુપ્લિકેટ ડેટા સ્ટોરેજ દૂર કરે છે
- કેન્દ્રીકૃત નિયંત્રણ: ડેટા મેનેજમેન્ટનું એક જ બિંદુ
- ડેટા વર્હચણી: અનેક એપ્લિકેશન્સ સમાન ડેટાનો ઉપયોગ કરી શકે છે
- બેકઅપ પુનઃપ્રાપ્તિ: આપોઆપ ડેટા સુરક્ષા પદ્ધતિઓ

મેમરી ટ્રીક

“DISC-RCDB - Database Is Super Cool”

પ્રશ્ન 1(c) [7 ગુણ]

DBA ની જવાબદારીઓ સમજાવો.

જવાબ

ટેબલ:

જવાબદારી	કાર્યો
ડેટાબેઝ ડિઝાઇન	તાર્કિક અને ભૌતિક રચનાઓ બનાવવી
સુરક્ષા મેનેજમેન્ટ	વપરાશકર્તા પહોંચ અને પરવાનગીઓનું નિયંત્રણ
પર્ફોર્મન્સ ટ્યુનિંગ	ક્વેરીઝ અને ડેટાબેઝ ઓપરેશન્સને ઓપ્ટિમાઇઝ કરવા
બેકઅપ પુનઃપ્રાપ્તિ	ડેટા સુરક્ષા અને પુનઃસ્થાપન સુનિશ્ચિત કરવું
યુઝર મેનેજમેન્ટ	એકાઉન્ટ બનાવવા અને વિશેષાધિકારો અસાઇન કરવા

Mermaid Diagram (Code)

```
{Shaded}
{Highlighting}[]
graph TD
    A[DBA ] --> B[ ]
    A --> C[ ]
    A --> D[ ]
    A --> E[ ]
    A --> F[ ]
    A --> G[ ]
```

```
{Highlighting}
{Shaded}
```

- ડેટાબેઝ ઇન્સ્ટલેશન: DBMS સોફ્ટવેર સેટઅપ અને કોન્ફિગર કરવું
- ડેટા માઇગ્રેશન: સિસ્ટમ્સ વચ્ચે ડેટાને સુરક્ષિત રીતે ટ્રાન્સફર કરવો
- ડોક્યુમેન્ટેશન: ડેટાબેઝ સ્કીમા અને પ્રક્રિયાઓ જાળવવી
- મોનિટરિંગ: સિસ્ટમ પર્ફોર્મન્સ અને રિસોર્સ વપરાશ ટ્રેક કરવું
- ટ્રબલશૂટિંગ: ડેટાબેઝ સમસ્યાઓ અને ભૂલો ઉકેલવી

મેમરી ટ્રીક

“DSPBU-DMT - DBA Solves Problems By Understanding Database Management Tasks”

પ્રશ્ન 1(c OR) [7 ગુણ]

ડેટા એબ્સ્ટ્રેક્શન શું છે? ત્રણ સ્તરની ANSI SPARC આર્કિટેક્ચરને વિગતવાર સમજાવો.

જવાબ

ડેટા એબ્સ્ટ્રેક્શન: વપરાશકર્તાઓથી જટિલ ડેટાબેઝ અમલીકરણ વિગતો છુપાવીને સરળ ઇન્ટરફેસ પ્રદાન કરવું.

Mermaid Diagram (Code)

```
{Shaded}
{Highlighting}[]
graph LR
    A[ ] --> B[ ]
    B --> C[ ]
    A1[ ] --> A
    B1[ ] --> B
    C1[ ] --> C
```

```
{Highlighting}
{Shaded}
```

ટેબલ:

સ્તર	વર્ણન	વપરાશકર્તાઓ
એક્સટર્નલ લેવલ	વ્યક્તિગત વપરાશકર્તા દૃશ્યો અને એપ્લિકેશન્સ	એન્ડ યુઝર્સ
કોન્સેપ્ચ્યુઅલ લેવલ	સંપૂર્ણ તાર્કિક ડેટાબેઝ રચના	ડેટાબેઝ ડિઝાઇનર્સ
ઇન્ટર્નલ લેવલ	ભૌતિક સ્ટોરેજ અને એક્સેસ પદ્ધતિઓ	સિસ્ટમ પ્રોગ્રામર્સ

- એક્સટર્નલ લેવલ: જટિલતા છુપાવતા અનેક વપરાશકર્તા દૃશ્યો
- કોન્સેપ્ચ્યુઅલ લેવલ: સ્ટોરેજ વિગતો વિના સંપૂર્ણ ડેટાબેઝ સ્કીમા
- ઇન્ટર્નલ લેવલ: ભૌતિક ફાઇલ સંગઠન અને ઇન્ડેક્સિંગ
- ડેટા સ્વતંત્રતા: એક સ્તરમાં ફેરફારો અન્યને અસર કરતા નથી

મેમરી ટ્રીક

“ECI - Every Computer Implements”

પ્રશ્ન 2(a) [3 ગુણ]

સ્કીમા અને ઇન્સ્ટન્સનો તફાવત સમજાવો

જવાબ

ટેબલ:

પાસું	સ્કીમા	ઇન્સ્ટન્સ
વ્યાખ્યા	ડેટાબેઝ રચનાનો બ્લુપ્રિન્ટ	ચોક્કસ સમયે વાસ્તવિક ડેટા
પ્રકૃતિ	સ્થિર તાર્કિક ડિઝાઇન	ડાયનામિક ડેટા સામગ્રી
ફેરફારો	ભાગ્યે જ સંશોધિત	વારંવાર અપડેટ

- સ્કીમા: ડેટાબેઝ સંગઠન અને મર્યાદાઓ વર્ણવે છે
- ઇન્સ્ટન્સ: ચોક્કસ ક્ષણે ડેટાબેઝ સામગ્રીનો સ્નેપશોટ
- સંબંધ: સ્કીમા રચના વ્યાખ્યાયિત કરે છે, ઇન્સ્ટન્સ ડેટા સમાવે છે

મેમરી ટ્રીક

“SI - Structure vs Information”

પ્રશ્ન 2(b) [4 ગુણ]

સ્પેશલાઇઝેશન ઉદાહરણ સાથે સમજાવો.

જવાબ

સ્પેશલાઇઝેશન: ચોક્કસ લાક્ષણિકતાઓના આધારે સુપરક્લાસમાંથી સબક્લાસ બનાવવાની પ્રક્રિયા.

```
erDiagram
    EMPLOYEE \{
        int emp\_id
        string name
        float salary
    \}
    MANAGER \{
        string department
        int team\_size
    \}
    DEVELOPER \{
        string programming\_language
        string project
    \}

    EMPLOYEE ||--| MANAGER : specializes
    EMPLOYEE ||--| DEVELOPER : specializes
```

- ટોપ-ડાઉન અપ્રોચ: સામાન્ય એન્ટિટીથી ચોક્કસ એન્ટિટીઓ તરફ
- ઇન્હેરિટન્સ: સબક્લાસેસ સુપરક્લાસના ગુણધર્મો વારસામાં લે છે
- ડિસજોઇન્ટ: મેનેજર અને ડેવલપર અલગ કટેગરી છે

- ઉદાહરણ: એમ્પ્લોયી મેનેજર અને ડેવલપરમાં વિશેષીકૃત

મેમરી ટ્રીક

“STID - Specialization Takes Inheritance Down”

પ્રશ્ન 2(c) [7 ગુણ]

ER ડાયાગ્રામ શું છે? ER ડાયાગ્રામમાં વપરાતા વિવિધ પ્રતીકોને ઉદાહરણ સાથે સમજાવો.

જવાબ

ER ડાયાગ્રામ: ડેટાબેઝ ડિઝાઇનમાં એન્ટિટીઝ, એટ્રિબ્યુટ્સ અને સંબંધો દર્શાવતી ગ્રાફિકલ પ્રતિનિધિત્વ.
ટેબલ:

પ્રતીક	આકાર	હેતુ	ઉદાહરણ
એન્ટિટી	લંબચોરસ	વાસ્તવિક વિશ્વનો ઓબ્જેક્ટ	Student, Course
એટ્રિબ્યુટ	અંડાકાર	એન્ટિટીના ગુણધર્મો	Name, Age, ID
સંબંધ	હીરો	એન્ટિટી કનેક્શન્સ	Enrolls, Takes
પ્રાઇમરી કી	અન્ડરલાઇન અંડાકાર	યુનિક આઇડેન્ટિફાયર	Student_ID

```
erDiagram
    STUDENT \{
        int student\_id PK
        string name
        string email
        date birth\_date
    \}
    COURSE \{
        string course\_id PK
        string course\_name
        int credits
    \}
    ENROLLMENT \{
        date enrollment\_date
        string grade
    \}

    STUDENT ||--o{ ENROLLMENT : enrolls
    COURSE ||--o{ ENROLLMENT : includes
```

- એન્ટિટી સેટ્સ: સમાન ગુણધર્મો ધરાવતી સમાન એન્ટિટીઝનો સંગ્રહ
- વીક એન્ટિટી: ઓળખ માટે સ્ટ્રોંગ એન્ટિટી પર આધારિત
- કાર્ડિનાલિટી: સંબંધ સહભાગિતા વ્યાખ્યાયિત કરે છે (1:1, 1:M, M:N)
- પાર્ટિસિપેશન: ટોટલ (ડબલ લાઇન) અથવા પાર્શિયલ (સિંગલ લાઇન)

મેમરી ટ્રીક

“EARP - Entities And Relationships Program”

પ્રશ્ન 2(a OR) [3 ગુણ]

DA અને DBA નો તફાવત સમજાવો.

જવાબ

ટેબલ:

પાસું	ડેટા એડમિનિસ્ટ્રેટર (DA)	ડેટાબેઝ એડમિનિસ્ટ્રેટર (DBA)
ફોકસ	ડેટા પોલિસીઝ અને સ્ટાન્ડર્ડ્સ	તકનીકી ડેટાબેઝ ઓપરેશન્સ
સ્તર	વ્યૂહાત્મક આયોજન	ઓપરેશનલ અમલીકરણ
સ્કોપ	સંસ્થા-વ્યાપી ડેટા	ચોક્કસ ડેટાબેઝ સિસ્ટમ્સ

- **DA:** સંસ્થાકીય સંસાધન તરીકે ડેટાનું સંચાલન કરે છે
- **DBA:** તકનીકી ડેટાબેઝ જાળવણી અને પર્ફોર્મન્સ સંભાળે છે
- **સહયોગ:** DA નીતિઓ સેટ કરે છે, DBA તેમને અમલમાં મૂકે છે

મેમરી ટ્રીક

“DA-DBA: Design Authority - Database Builder Administrator”

પ્રશ્ન 2(b OR) [4 ગુણ]

જનરલાઈઝેશન ઉદાહરણ સાથે સમજાવો.

જવાબ

જનરલાઈઝેશન: સમાન એન્ટિટીઝને સામાન્ય સુપરક્લાસમાં જોડવાની બોટમ-અપ પ્રક્રિયા.

```
erDiagram
    VEHICLE \{
        string vehicle\_id
        string brand
        int year
        string color
    \}
    CAR \{
        int doors
        string fuel\_type
    \}
    MOTORCYCLE \{
        int engine\_cc
        string bike\_type
    \}

    VEHICLE ||--| CAR : generalizes
    VEHICLE ||--| MOTORCYCLE : generalizes
```

- **બોટમ-અપ અપ્રોચ:** ચોક્કસ એન્ટિટીઝથી સામાન્ય એન્ટિટી તરફ
- **કોમન એટ્રિબ્યુટ્સ:** સહેજ ગુણધર્મો સુપરક્લાસમાં ખસેડાય છે
- **સ્પેશ્યલાઈઝેશન રિવર્સ:** સ્પેશ્યલાઈઝેશન પ્રક્રિયાનું વિપરીત
- **ઉદાહરણ:** કાર અને મોટરસાઈકલ વાહનમાં સામાન્યીકૃત

મેમરી ટ્રીક

“GBCS - Generalization Brings Common Superclass”

પ્રશ્ન 2(c OR) [7 ગુણ]

એટ્રિબ્યુટ શું છે? વિવિધ પ્રકારના એટ્રિબ્યુટ્સ ઉદાહરણ સાથે સમજાવો.

જવાબ

એટ્રિબ્યુટ: એન્ટિટીનું વર્ણન કરતી ગુણવત્તા અથવા લાક્ષણિકતા.
ટેબલ:

એટ્રિબ્યુટ પ્રકાર	વર્ણન	ઉદાહરણ
સિમ્પલ	વધુ વિભાજિત કરી શકાતું નથી	Age, Name
કોમ્પોઝિટ	ઉપવિભાગ કરી શકાય છે	Address (Street, City, ZIP)
સિંગલ-વેલ્યુડ	એન્ટિટી દીઠ એક મૂલ્ય	Student_ID
મલ્ટિ-વેલ્યુડ	અનેક મૂલ્યો શક્ય	Phone_numbers
ડેરાઇવ્ડ	અન્ય એટ્રિબ્યુટ્સમાંથી ગણાય છે	Age from Birth_date

Mermaid Diagram (Code)

```

{Shaded}
{Highlighting}[]
graph TD
    A[ ] --{-}{ } B[ ]
    A --{-}{ } C[ ]
    A --{-}{ } D[ {-} ]
    A --{-}{ } E[ {-} ]
    A --{-}{ } F[ ]
    A --{-}{ } G[ ]

    C --{-}{ } C1[Address: Street, City, ZIP]
    E --{-}{ } E1[Phone: Mobile, Home, Work]
    F --{-}{ } F1[Age calculated from DOB]
  
```

- **કી એટ્રિબ્યુટ:** એન્ટિટી ઇન્સ્ટન્સને યુનિકલી ઓળખે છે
- **નલ વેલ્યુઝ:** એટ્રિબ્યુટ્સ કે જેમાં કોઈ મૂલ્ય ન હોઈ શકે
- **ડિફોલ્ટ વેલ્યુઝ:** નિર્દિષ્ટ ન હોય ત્યારે પૂર્વનિર્ધારિત મૂલ્યો
- **કન્સ્ટ્રેઇન્ટ્સ:** એટ્રિબ્યુટ મૂલ્યોને સંચાલિત કરતા નિયમો

મેમરી ટ્રીક

``SCSMD-K - Simple Composite Single Multi Derived Key``

પ્રશ્ન 3(a) [3 ગુણ]

SQL માં GRANT અને REVOKE સ્ટેટમેન્ટ સમજાવો.

જવાબ

ટેબલ:

સ્ટેટમેન્ટ	હેતુ	સિન્ટેક્સ ઉદાહરણ
GRANT	વપરાશકર્તાઓને વિશેષાધિકારો પ્રદાન કરે છે	GRANT SELECT ON table TO user
REVOKE	વપરાશકર્તા વિશેષાધિકારો દૂર કરે છે	REVOKE INSERT ON table FROM user

```
{--{-} Grant privileges}
GRANT SELECT, INSERT ON employees TO john;
GRANT ALL PRIVILEGES ON database TO admin;
```

```
{--{-} Revoke privileges }
REVOKE DELETE ON employees FROM john;
REVOKE ALL ON database FROM user;
```

- **વિશેષાધિકારો:** SELECT, INSERT, UPDATE, DELETE, ALL
- **ઓબ્જેક્ટ્સ:** ટેબલ્સ, વ્યૂઝ, ડેટાબેઝિસ, પ્રોસીજર્સ
- **સુરક્ષા:** ડેટા એક્સેસ અને મોડિફિકેશન રાઇટ્સનું નિયંત્રણ

મેમરી ટ્રીક

“GR - Grant Rights, Remove Rights”

પ્રશ્ન 3(b) [4 ગુણ]

નીચેના Character functions સમજાવો. 1) INITCAP 2) SUBSTR

જવાબ

ટેબલ:

ફંક્શન	હેતુ	સિન્ટેક્સ	ઉદાહરણ
INITCAP	દરેક શબ્દનો પહેલો અક્ષર મોટો કરે છે	INITCAP(string)	INITCAP('hello world') = 'Hello World'
SUBSTR	સ્ટ્રિંગમાંથી સબસ્ટ્રિંગ કાઢે છે	SUBSTR(string, start, length)	SUBSTR('Database', 1, 4) = 'Data'

{--{-} INITCAP examples}

```
SELECT INITCAP({database management}) FROM dual; {--{-} Database Management}
```

```
SELECT INITCAP({gtu university}) FROM dual; {--{-} Gtu University}
```

{--{-} SUBSTR examples }

```
SELECT SUBSTR({Programming}, 1, 7) FROM dual; {--{-} Program}
```

```
SELECT SUBSTR({Database}, 5) FROM dual;
```

{--{-} base}

- **INITCAP:** સ્ટ્રિંગને યોગ્ય કેસ ફોર્મેટમાં કન્વર્ટ કરે છે
- **SUBSTR:** પેરામીટર્સ છે સ્ટ્રિંગ, શરૂઆતની સ્થિતિ, વૈકલ્પિક લંબાઈ
- **વપરાશ:** ટેક્સ્ટ ફોર્મેટિંગ અને સ્ટ્રિંગ મેનિપ્યુલેશન ઓપરેશન્સ

મેમરી ટ્રીક

“IS - Initialize String, Split String”

પ્રશ્ન 3(c) [7 ગુણ]

નીચે દર્શાવેલ ટેબલને ધ્યાનમાં લઈ આપેલ ક્વેરીઝના જવાબ લખો. stud_master (enroll_no, name, city, dept)

જવાબ

```
{--{-} 1. IT dept }
SELECT * FROM stud\_master
WHERE dept = {IT};
```

```

{-{-} 2. p
SELECT * FROM stud\_master
WHERE name LIKE {p\%};

{-{-} 3.
INSERT INTO stud\_master (enroll\_no, name, city, dept)
VALUES ({202501}, {John Smith}, {Mumbai}, {CS});

{-{-} 4. stud\_master gender
ALTER TABLE stud\_master
ADD gender VARCHAR(10);

{-{-} 5. stud\_master
SELECT COUNT(*) FROM stud\_master;

{-{-} 6. enroll\_no
SELECT * FROM stud\_master
ORDER BY enroll\_no DESC;

{-{-} 7. stud\_master
DROP TABLE stud\_master;

```

ટેબલ:

કવેરી પ્રકાર	SQL કમાન્ડ	હેતુ
SELECT	ડેટા મેળવે છે	રેકૉર્ડ્સ દર્શાવે છે
INSERT	નવો ડેટા ઉમેરે છે	રેકૉર્ડ્સ બનાવે છે
ALTER	રચના સંશોધિત કરે છે	કૉલમ્સ ઉમેરે છે
COUNT	એગ્રિગેટ ફંક્શન	પંક્તિઓ ગણે છે

મેમરી ટ્રીક

“SIAC-DOC - SQL Is A Complete Database Operations Collection”

પ્રશ્ન 3(a OR) [3 ગુણ]

SQL માં equi join ઉદાહરણ સાથે સમજાવો.

જવાબ

Equi Join: સમાન કૉલમ્સના આધારે ટેબલ્સને જોડવા માટે સમાનતા શરતનો ઉપયોગ કરતું જોઇન ઓપરેશન.

```

{-{-} Equi Join
SELECT s.name, c.course\_name
FROM students s, courses c
WHERE s.course\_id = c.course\_id;

{-{-} JOIN
SELECT s.name, c.course\_name
FROM students s
JOIN courses c ON s.course\_id = c.course\_id;

```

- **સમાનતા ઓપરેટર:** કૉલમ મૂલ્યો મેચ કરવા માટે = નો ઉપયોગ
- **કોમન કૉલમ્સ:** ટેબલ્સમાં સંબંધિત એટ્રિબ્યુટ્સ હોવા જોઈએ
- **પરિણામ:** મેચના આધારે અનેક ટેબલ્સમાંથી સંયુક્ત ડેટા

મેમરી ટ્રીક

“EJ - Equal Join”

પ્રશ્ન 3(b OR) [4 ગુણ]

નીચેના Aggregate functions સમજાવો. 1) MAX 2) SUM

જવાબ

ટેબલ:

ફંક્શન	હેતુ	સિન્ટેક્સ	ઉદાહરણ
MAX	મહત્તમ મૂલ્ય પરત કરે છે	MAX(column)	MAX(salary) = 50000
SUM	મૂલ્યોનો કુલ સરવાળો પરત કરે છે	SUM(column)	SUM(marks) = 450

```
--{-} MAX      }
SELECT MAX(salary) FROM employees; {-{-}      }
SELECT MAX(age) FROM students; {-{-}      }
```

```
--{-} SUM      }
SELECT SUM(credits) FROM courses; {-{-}      }
SELECT SUM(price * quantity) FROM orders; {-{-}      }
```

- એગ્રિગેટ ફંક્શન્સ: અનેક પંક્તિઓ પર કામ કરે છે, એક મૂલ્ય પરત કરે છે
- NULL હેન્ડલિંગ: ગણતરીમાં NULL મૂલ્યોને અવગણે છે
- GROUP BY: કેટેગરી-વાઇઝ પરિણામો માટે ગ્રુપિંગ સાથે ઉપયોગ કરી શકાય છે

મેમરી ટ્રીક

``MS - Maximum Sum``

પ્રશ્ન 3(c OR) [7 ગુણ]

નીચે દર્શાવેલ ટેબલ માટે SQL ક્વેરીઝ લખો: PRODUCT_Master: (prod_no, prod_name, profit, quantity, sell_price, cost_price)

જવાબ

```
--{-} 1. PRODUCT\_Master      }
CREATE TABLE PRODUCT\_Master (
    prod\_no VARCHAR(10) PRIMARY KEY,
    prod\_name VARCHAR(50),
    profit NUMBER(10,2),
    quantity NUMBER,
    sell\_price NUMBER(10,2),
    cost\_price NUMBER(10,2)
);

--{-} 2.      }
INSERT INTO PRODUCT\_Master VALUES
({P001}, {Laptop}, 15000, 10, 45000, 30000);

--{-} 3. 20000      }
SELECT * FROM PRODUCT\_Master
WHERE profit {} 20000;

--{-} 4. 5      quantity      }
DELETE FROM PRODUCT\_Master
WHERE quantity {} 5;

--{-} 5. 5000      sell\_price      2\%      }
UPDATE PRODUCT\_Master
SET profit = profit * 1.02
WHERE sell\_price {} 5000;
```