Project title: Database Reads

ABSTRACT:

This project entitled as "Database reads" ,is nothing but an e-learning website where a brief description and resource material links related to all database management system topics are being provided. The main objective of this website is to provide a detailed information of all database topics on a single platform. The website also provides pictorial representations, animations, flowcharts, etc for a better understanding of the concepts.

We believe that everyone must be equipped with basic knowledge in Database management system, as well as use it in creating websites etc to reach a particular goal and aim. In the 20th century, we have moved from the Industrial Age through the Information Age and now to the Knowledge Age. Knowledge and its efficient management constitute the key to success and survival for organizations in the highly dynamic and competitive world of today. Efficient acquisition, storage, transfer, retrieval, application, and visualization of database management system is often useful for many organizations.

We have also given resource material links for each and every topic, which will help the user to explore more about database management systems. If you select any topic, then you will get a brief description of the topic along with pictorial flowcharts, animated videos and resource links.

REQUIREMENTS:

Overall 6 tables are required for the ER model

representation , they are:

1. Modules
2. Topics
3. Modules_has_Topics
4. Resources
5. User
6. User_Res_Fav

Table 1: Modules:

Attributes:

- m_id (primary key)

- m_name (varchar)

Domain Types:

- m_id : integer

- m_name: varchar

Constraints:

- Primary key: m_id

Table 2: Topics

Attributes:

- t_id (primary key)

- t_name (varchar)

- description (varchar)

Domain Types:

- t_id : integer

- t_name, description : varchar

Constraints:

- Primary key: t_id

Table 3: Modules_has_Topics

Attributes:

- m_id (primary key)

- t_id (number)

Domain Types:

- t_id, m_id: integer

Constraints:

- Primary key: m_id

Table 4: Resources

Attributes:

- r_id (primary key)

- r_name (varchar)

-typeof (varchar)

- url (varchar)

- t_id (foreign key)

Domain Types:

- r_id, t_id: integer

- r_name, typeof, url, : varchar

Constraints:

- Primary key: r_id

-Foreign key:t_id

Table 5: User

Attributes:

- u_id (primary key)

- u_name (varchar)

- email (varchar)

- password (varchar)

Domain Types:

- u_id: integer

- u_name, email, pass: varchar

Constraints:

- Primary key: u_id

Table 6: User_Fav_Res

Attributes:

- u_id (foreign key)

- r_id (number)

Domain Types:

- u_id, r_id: integer

Constraints:

- Foreign key: u_id

MAPPING CARDINALITIES:

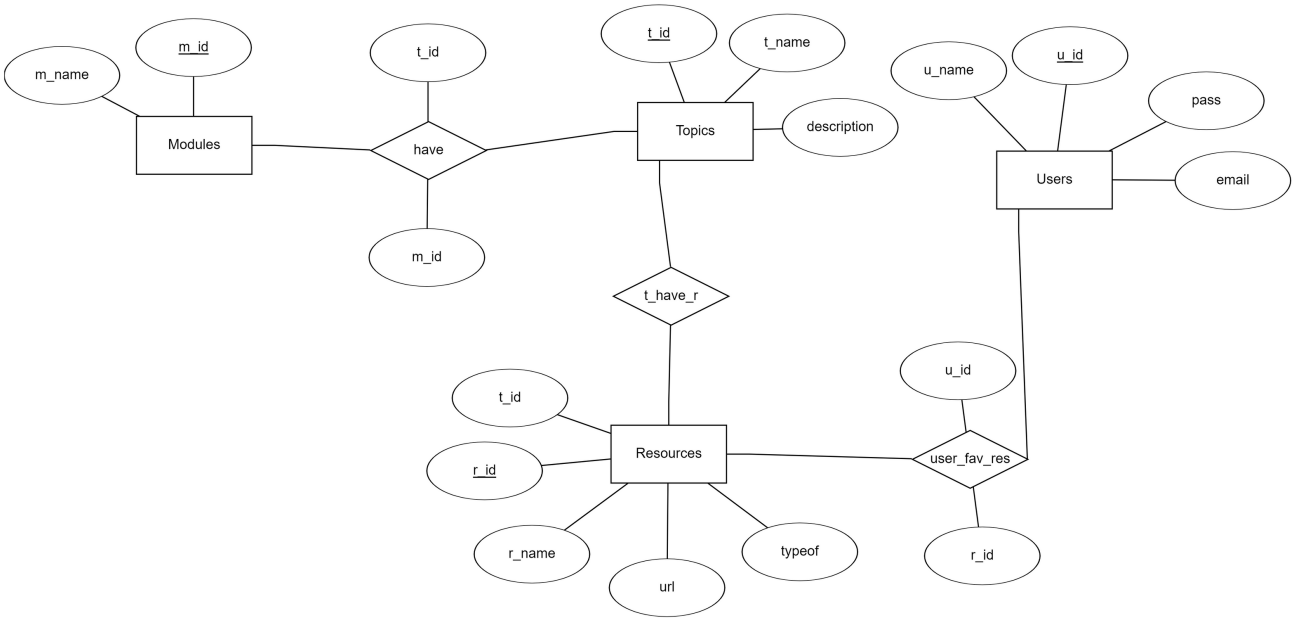
- One User can access single resources.

- One module can have multiple topics.

- One topic can have multiple resources .

- One user can have multiple favourite resources.

DESIGN (ENTITY RELATIONSHIP MODEL):



DDL COMMANDS:

SQL> create table modules(m_id number primary key,m_name varchar2(20));

Table created.

SQL> create table topics(t_id number primary key,t_name varchar2(20),description varchar2(20));

Table created.

SQL> create table have(t_id number,m_id number, primary key(t_id,m_id));

Table created.

SQL> create table resources(r_id number,t_id number,r_name varchar2(20),typeof varchar2(20),url varchar2(20),primary key(r_id),foreign key(t_id) references topics);

Table created.

SQL> create table users(u_id number primary key,email varchar2(20),pass varchar2(20));

Table created.

SQL> create table user_fav_res(u_id number,r_id number,foreign key(u_id) references resources);

Table created.

```
Connected.
SQL> desc modules;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 M_ID                                      NOT NULL NUMBER
 M_NAME                                             VARCHAR2(20)

SQL> desc topics;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 T_ID                                      NOT NULL NUMBER
 T_NAME                                             VARCHAR2(20)
 DESCRIPTION                                        VARCHAR2(20)

SQL> desc have;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 T_ID                                      NOT NULL NUMBER
 M_ID                                      NOT NULL NUMBER

SQL> desc resources;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 R_ID                                      NOT NULL NUMBER
 T_ID                                               NUMBER
 R_NAME                                             VARCHAR2(20)
 TYPEOF                                             VARCHAR2(20)
 URL                                                VARCHAR2(20)

SQL> desc users;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 U_ID                                      NOT NULL NUMBER
 EMAIL                                              VARCHAR2(20)
 PASS                                               VARCHAR2(20)
 U_NAME                                             VARCHAR2(20)

SQL> desc user_fav_res;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 U_ID                                               NUMBER
 R_ID                                               NUMBER
```

DML COMMANDS:

TABLE 1: users

```
SQL> insert into users values (&u_id,'&u_name','&email','&pass');
Enter value for u_id: 001
Enter value for u_name: vanshika
Enter value for email: vanshika@gmail.com
Enter value for pass: 123
old   1: insert into users values (&u_id,'&u_name','&email','&pass')
new   1: insert into users values (001,'vanshika','vanshika@gmail.com','123')

1 row created.

SQL> /
Enter value for u_id: 002
Enter value for u_name: keerthi
Enter value for email: keerthi@gmail.com
Enter value for pass: 456
old   1: insert into users values (&u_id,'&u_name','&email','&pass')
new   1: insert into users values (002,'keerthi','keerthi@gmail.com','456')

1 row created.

SQL> /
Enter value for u_id: 003
Enter value for u_name: sanjana
Enter value for email: sanjana@gmail.com
Enter value for pass: 789
old   1: insert into users values (&u_id,'&u_name','&email','&pass')
new   1: insert into users values (003,'sanjana','sanjana@gmail.com','789')

1 row created.

SQL> /
Enter value for u_id: 004
Enter value for u_name: sathvik
Enter value for email: sathvik@gmail.com
Enter value for pass: 101112
old   1: insert into users values (&u_id,'&u_name','&email','&pass')
new   1: insert into users values (004,'sathvik','sathvik@gmail.com','101112')

1 row created.
```

```
SQL> select * from users;

      U_ID U_NAME               EMAIL               PASS
---------- -------------------- ------------------- --------------------
         1 vanshika             vanshika@gmail.com   123
         2 keerthi              keerthi@gnail.com    345
         3 sanjana              sanjana@gmail.com    567
         4 sathvik              sathvik@gmail.com    789
         5 srinika              srinika@gmail.com    159
```

TABLE 2: modules

```
SQL> insert into modules values (&m_id,'&m_name');
Enter value for m_id: 1
Enter value for m_name: introduction
old   1: insert into modules values (&m_id,'&m_name')
new   1: insert into modules values (1,'introduction')

1 row created.

SQL> /
Enter value for m_id: 2
Enter value for m_name: entity relationship model
old   1: insert into modules values (&m_id,'&m_name')
new   1: insert into modules values (2,'entity relationship model')
insert into modules values (2,'entity relationship model')
                           *
ERROR at line 1:
ORA-12899: value too large for column "VANSH"."MODULES"."M_NAME" (actual: 25,
maximum: 20)


SQL> /
Enter value for m_id: 2
Enter value for m_name: er model
old   1: insert into modules values (&m_id,'&m_name')
new   1: insert into modules values (2,'er model')

1 row created.

SQL> /
Enter value for m_id: 3
Enter value for m_name: relational model
old   1: insert into modules values (&m_id,'&m_name')
new   1: insert into modules values (3,'relational model')

1 row created.

SQL> /
Enter value for m_id: 4
Enter value for m_name: normalization
```

```
SQL> select * from modules;

      M_ID M_NAME
---------- --------------------
         1 introduction
         2 er model
         3 relational model
         4 normalization
         5 sql
```

TABLE 3: topics

```
SQL> insert into topics values(&t_id,'&t_name','&description');
Enter value for t_id: 1
Enter value for t_name: intro of dbms
Enter value for description: .
old   1: insert into topics values(&t_id,'&t_name','&description')
new   1: insert into topics values(1,'intro of dbms','.')

1 row created.

SQL> /
Enter value for t_id: 2
Enter value for t_name: arch
Enter value for description: architecture
old   1: insert into topics values(&t_id,'&t_name','&description')
new   1: insert into topics values(2,'arch','architecture')

1 row created.

SQL> /
Enter value for t_id: 3
Enter value for t_name: adv
Enter value for description: adv over files
old   1: insert into topics values(&t_id,'&t_name','&description')
new   1: insert into topics values(3,'adv','adv over files')

1 row created.

SQL> /
Enter value for t_id: 4
Enter value for t_name: models
Enter value for description: data models
old   1: insert into topics values(&t_id,'&t_name','&description')
new   1: insert into topics values(4,'models','data models')

1 row created.

SQL> /
Enter value for t_id: 5
Enter value for t_name: er diag
Enter value for description: representation
old   1: insert into topics values(&t_id,'&t_name','&description')
```

```
SQL> select * from topics;

      T_ID T_NAME               DESCRIPTION
---------- -------------------- --------------------
         1 intro of dbms        .
         2 arch                 architecture
         3 adv                  adv over files
         4 models               data models
         5 er diag              representation
         6 generalization       aggregation
         7 codds rules          .
         8 logical model        relational model
         9 relational algebra   .
        10 er to relational     .
        11 sql overview         .

      T_ID T_NAME               DESCRIPTION
---------- -------------------- --------------------
        12 normalization        .
        13 joins                .
        14 transaction
```

```
        14 transaction          .
        15 concurrency control  .

15 rows selected.
```

TABLE 4: modules_has_topics

```
SQL> insert into have values(&m_id,&t_id);
Enter value for m_id: 1
Enter value for t_id: 1
old   1: insert into have values(&m_id,&t_id)
new   1: insert into have values(1,1)

1 row created.

SQL> /
Enter value for m_id: 1
Enter value for t_id: 2
old   1: insert into have values(&m_id,&t_id)
new   1: insert into have values(1,2)

1 row created.

SQL> /
Enter value for m_id: 1
Enter value for t_id: 3
old   1: insert into have values(&m_id,&t_id)
new   1: insert into have values(1,3)

1 row created.

SQL> /
Enter value for m_id: 1
Enter value for t_id: l
old   1: insert into have values(&m_id,&t_id)
new   1: insert into have values(1,l)
insert into have values(1,l)
                        *
```

```
SQL> select * from have;

      T_ID       M_ID
---------- ----------
         1          1
         1          2
         1          3
         1          7
         2          4
         2          5
         3          6
         3          8
         3          9
         3         10
         4         12

      T_ID       M_ID
---------- ----------
         4         13
         4         14
         4         15
         5         11

15 rows selected.
```

TABLE 5: resources

```
SQL> insert into resources values(&t_id,&r_id,'&r_name','&typeof','&url');
Enter value for t_id: 1
Enter value for r_id: 1
Enter value for r_name: tutorial
Enter value for typeof: website
Enter value for url: .
old   1: insert into resources values(&t_id,&r_id,'&r_name','&typeof','&url')
new   1: insert into resources values(1,1,'tutorial','website','.')

1 row created.

SQL> /
Enter value for t_id: 2
Enter value for r_id: 1
Enter value for r_name: java t
Enter value for typeof: website
Enter value for url: .
old   1: insert into resources values(&t_id,&r_id,'&r_name','&typeof','&url')
new   1: insert into resources values(2,1,'java t','website','.')

1 row created.

SQL> /
Enter value for t_id: 2
Enter value for r_id: 5
Enter value for r_name: er diag
Enter value for typeof: diagram
Enter value for url: .
old   1: insert into resources values(&t_id,&r_id,'&r_name','&typeof','&url')
new   1: insert into resources values(2,5,'er diag','diagram','.')
insert into resources values(2,5,'er diag','diagram','.')
*
ERROR at line 1:
ORA-00001: unique constraint (VANSH.SYS_C006990) violated
```

```
SQL> select * from resources;

      R_ID       T_ID R_NAME               TYPEOF
---------- ---------- -------------------- --------------------
URL
--------------------
         1          1 tutorial             website
.

         2          1 java t               website
.

         3          2 adv                  web
,


      R_ID       T_ID R_NAME               TYPEOF
---------- ---------- -------------------- --------------------
URL
--------------------
         5          2 er                   diagram
n

         7          3 codd                 description
```

```
n
```

TABLE 6: user_fav_res

```
1 row created.

SQL> /
Enter value for u_id: 002
Enter value for r_id: 1
old   1: insert into user_fav_res values(&u_id,&r_id)
new   1: insert into user_fav_res values(002,1)

1 row created.

SQL> /
Enter value for u_id: 003
Enter value for r_id: 5
old   1: insert into user_fav_res values(&u_id,&r_id)
new   1: insert into user_fav_res values(003,5)

1 row created.

SQL> /
Enter value for u_id: 004
Enter value for r_id: 2
old   1: insert into user_fav_res values(&u_id,&r_id)
new   1: insert into user_fav_res values(004,2)
insert into user_fav_res values(004,2)
*
ERROR at line 1:
ORA-02291: integrity constraint (VANSH.SYS_C006993) violated - parent key not
found


SQL> /
Enter value for u_id: 004
Enter value for r_id: 6
old   1: insert into user_fav_res values(&u_id,&r_id)
new   1: insert into user_fav_res values(004,6)
insert into user_fav_res values(004,6)
*
ERROR at line 1:
ORA-02291: integrity constraint (VANSH.SYS_C006993) violated - parent key not
found
```

```
SQL> select * from user_fav_res;

      U_ID       R_ID
---------- ----------
         1          1
         2          2
         3          3

SQL>
```