IT Skill Test GIC Myanmar

Duration: 30 Minutes Total Questions: 8

---

1. You need to create a foreign key in the 'orders' table referencing the 'customers' table. Which SQL statement accomplishes this?

   A.ALTER TABLE orders ADD CONSTRAINT fk_customer FOREIGN KEY (customer_id) REFERENCES customers(id);

   B.ALTER TABLE orders ADD FOREIGN KEY (customer_id) REFERENCES customers(customer_id);

   C.CREATE FOREIGN KEY ON orders (customer_id) REFERENCES customers(id);

   D.ALTER TABLE orders ADD CONSTRAINT fk_customer FOREIGN KEY customer_id REFERENCES customers;

2. You're creating a table to store employee information. Which of the following correctly adds a NOT NULL constraint to the 'email' column?

   A.CREATE TABLE employees (id NUMBER, email VARCHAR2(100) CONSTRAINT nn_email NOT NULL);

   B.CREATE TABLE employees (id NUMBER, email VARCHAR2(100) NOT NULL);

   C.CREATE TABLE employees (id NUMBER, email VARCHAR2(100), CONSTRAINT nn_email NOT NULL (email));

   D.CREATE TABLE employees (id NUMBER, email VARCHAR2(100) NOT ALLOW NULL);

3. You need to add a CHECK constraint to ensure that the 'age' column only accepts values between 18 and 65. Which SQL statement is correct?

   A.ALTER TABLE employees ADD CONSTRAINT chk_age CHECK (age BETWEEN 18 AND 65);

   B.ALTER TABLE employees ADD CHECK (age >= 18 AND age <= 65);

C. ALTER TABLE employees ADD CONSTRAINT chk_age CHECK age BETWEEN 18 AND 65;

D. ALTER TABLE employees MODIFY age CHECK (BETWEEN 18 AND 65);

4. You're troubleshooting a database where users complain about duplicate email addresses. Which constraint should you add to prevent this issue?

A. ALTER TABLE users ADD CONSTRAINT uk_email UNIQUE (email);

B. ALTER TABLE users MODIFY email UNIQUE;

C. ALTER TABLE users ADD CONSTRAINT pk_email PRIMARY KEY (email);

D. ALTER TABLE users ADD DISTINCT (email);

5. In a customer management system, you need to ensure that the 'status' column only accepts values 'Active', 'Inactive', or 'Suspended'. Which constraint would you use?

A. ALTER TABLE customers ADD CONSTRAINT chk_status CHECK (status IN ('Active', 'Inactive', 'Suspended'));

B. ALTER TABLE customers MODIFY status ENUM('Active', 'Inactive', 'Suspended');

C. ALTER TABLE customers ADD CONSTRAINT chk_status VALUES ('Active', 'Inactive', 'Suspended');

D. ALTER TABLE customers ADD CHECK status ('Active', 'Inactive', 'Suspended');

6. You're creating a table to store product information. The product code must be unique, not null, and serve as the primary identifier for each product. Which SQL statement correctly implements these requirements?

A. CREATE TABLE products (product_code VARCHAR2(10) UNIQUE NOT NULL, name VARCHAR2(100), price NUMBER(10,2));

B. CREATE TABLE products (product_code VARCHAR2(10) PRIMARY KEY, name VARCHAR2(100), price NUMBER(10,2));

C. CREATE TABLE products (product_code VARCHAR2(10), name

VARCHAR2(100), price NUMBER(10,2), CONSTRAINT pk_product PRIMARY KEY (product_code), CONSTRAINT uk_product UNIQUE (product_code));

D.CREATE TABLE products (product_code VARCHAR2(10) NOT NULL, name VARCHAR2(100), price NUMBER(10,2), UNIQUE (product_code));

7. You are designing a table to store customer information for an e-commerce platform. Which of the following SQL statements correctly creates the table with appropriate constraints?

A.CREATE TABLE customers (customer_id NUMBER, email VARCHAR2(100), name VARCHAR2(50), PRIMARY KEY (customer_id), UNIQUE (email));

B.CREATE TABLE customers (customer_id NUMBER PRIMARY KEY, email VARCHAR2(100) UNIQUE NOT NULL, name VARCHAR2(50) NOT NULL);

C.CREATE TABLE customers (customer_id NUMBER, email VARCHAR2(100), name VARCHAR2(50), CONSTRAINT pk_customer PRIMARY KEY (customer_id));

D.CREATE TABLE customers (customer_id NUMBER, email VARCHAR2(100), name VARCHAR2(50), PRIMARY KEY (customer_id), CONSTRAINT uq_email UNIQUE (email));

8. You're working on a database for a library system. You need to create a table for books and another for authors, with a many-to-many relationship between them. Which of the following approaches correctly implements this relationship with appropriate constraints?

A.CREATE TABLE books (book_id NUMBER PRIMARY KEY, title VARCHAR2(100) NOT NULL);
CREATE TABLE authors (author_id NUMBER PRIMARY KEY, name VARCHAR2(100) NOT NULL);
CREATE TABLE book_authors (book_id NUMBER, author_id NUMBER, PRIMARY KEY (book_id, author_id), FOREIGN KEY (book_id) REFERENCES books(book_id), FOREIGN KEY (author_id) REFERENCES authors(author_id));

B.CREATE TABLE books (book_id NUMBER PRIMARY KEY, title VARCHAR2(100) NOT NULL, author_id NUMBER, FOREIGN KEY (author_id) REFERENCES

authors(author_id));
CREATE TABLE authors (author_id NUMBER PRIMARY KEY, name VARCHAR2(100) NOT NULL);

C.CREATE TABLE books (book_id NUMBER PRIMARY KEY, title VARCHAR2(100) NOT NULL);
CREATE TABLE authors (author_id NUMBER PRIMARY KEY, name VARCHAR2(100) NOT NULL, book_id NUMBER, FOREIGN KEY (book_id) REFERENCES books(book_id));

D.CREATE TABLE books (book_id NUMBER PRIMARY KEY, title VARCHAR2(100) NOT NULL);
CREATE TABLE authors (author_id NUMBER PRIMARY KEY, name VARCHAR2(100) NOT NULL);
CREATE TABLE book_authors (book_id NUMBER REFERENCES books(book_id), author_id NUMBER REFERENCES authors(author_id));