

Database Relationships SQL Scripts

Database Relationships SQL Scripts

Database Relationships SQL Scripts

One-to-One Relationships

Employee and Employee Details:

```
CREATE TABLE employee (  
    emp_id INT PRIMARY KEY,  
    emp_name VARCHAR(100)  
);
```

```
CREATE TABLE employee_details (  
    emp_id INT PRIMARY KEY,  
    address VARCHAR(200),  
    phone VARCHAR(20),  
    FOREIGN KEY (emp_id) REFERENCES employee(emp_id)  
);
```

User and User Profile:

```
CREATE TABLE user (  
    user_id INT PRIMARY KEY,  
    username VARCHAR(100),  
    password VARCHAR(100)  
);
```

```
CREATE TABLE user_profile (  
    user_id INT PRIMARY KEY,  
    profile_name VARCHAR(100),  
    profile_data VARCHAR(1000)
```

Database Relationships SQL Scripts

```
user_id INT PRIMARY KEY,  
  
full_name VARCHAR(200),  
  
email VARCHAR(100),  
  
FOREIGN KEY (user_id) REFERENCES user(user_id)  
  
);
```

Database Relationships SQL Scripts

One-to-Many Relationships

Department and Employees:

```
CREATE TABLE department (  
    dept_id INT PRIMARY KEY,  
    dept_name VARCHAR(100)  
);
```

```
CREATE TABLE employee (  
    emp_id INT PRIMARY KEY,  
    emp_name VARCHAR(100),  
    dept_id INT,  
    FOREIGN KEY (dept_id) REFERENCES department(dept_id)  
);
```

Customer and Orders:

```
CREATE TABLE customer (  
    customer_id INT PRIMARY KEY,  
    customer_name VARCHAR(100)  
);
```

```
CREATE TABLE orders (  
    order_id INT PRIMARY KEY,
```

Database Relationships SQL Scripts

```
order_date DATE,  
  
customer_id INT,  
  
total_amount DECIMAL(10, 2),  
  
FOREIGN KEY (customer_id) REFERENCES customer(customer_id)  
  
);
```

Database Relationships SQL Scripts

Many-to-Many Relationships

Students and Courses:

```
CREATE TABLE student (  
    student_id INT PRIMARY KEY,  
    student_name VARCHAR(100)  
);
```

```
CREATE TABLE course (  
    course_id INT PRIMARY KEY,  
    course_name VARCHAR(100)  
);
```

```
CREATE TABLE enrollment (  
    student_id INT,  
    course_id INT,  
    PRIMARY KEY (student_id, course_id),  
    FOREIGN KEY (student_id) REFERENCES student(student_id),  
    FOREIGN KEY (course_id) REFERENCES course(course_id)  
);
```

Authors and Books:

```
CREATE TABLE author (  
    author_id INT PRIMARY KEY,  
    author_name VARCHAR(100)
```

Database Relationships SQL Scripts

```
author_id INT PRIMARY KEY,  
  
author_name VARCHAR(100)  
  
);
```

```
CREATE TABLE book (  
  
    book_id INT PRIMARY KEY,  
  
    book_title VARCHAR(200)  
  
);
```

```
CREATE TABLE author_book (  
  
    author_id INT,  
  
    book_id INT,  
  
    PRIMARY KEY (author_id, book_id),  
  
    FOREIGN KEY (author_id) REFERENCES author(author_id),  
  
    FOREIGN KEY (book_id) REFERENCES book(book_id)  
  
);
```

Database Relationships SQL Scripts

Additional Examples

Products and Reviews:

```
CREATE TABLE product (  
    product_id INT PRIMARY KEY,  
    product_name VARCHAR(100),  
    price DECIMAL(10, 2)  
);  
  
CREATE TABLE review (  
    review_id INT PRIMARY KEY,  
    product_id INT,  
    reviewer_name VARCHAR(100),  
    review_text TEXT,  
    FOREIGN KEY (product_id) REFERENCES product(product_id)  
);
```

Orders and Order Items:

```
CREATE TABLE orders (  
    order_id INT PRIMARY KEY,  
    order_date DATE,  
    total_amount DECIMAL(10, 2)  
);
```


Database Relationships SQL Scripts

```
CREATE TABLE order_items (  
    order_item_id INT PRIMARY KEY,  
    order_id INT,  
    product_id INT,  
    quantity INT,  
    price_per_unit DECIMAL(10, 2),  
    FOREIGN KEY (order_id) REFERENCES orders(order_id),  
    FOREIGN KEY (product_id) REFERENCES product(product_id)  
);
```

Customers and Addresses:

```
CREATE TABLE customer (  
    customer_id INT PRIMARY KEY,  
    customer_name VARCHAR(100)  
);
```

```
CREATE TABLE address (  
    address_id INT PRIMARY KEY,  
    customer_id INT,  
    address_line VARCHAR(200),  
    city VARCHAR(100),  
    state VARCHAR(50),  
    zip_code VARCHAR(20),
```

Database Relationships SQL Scripts

```
FOREIGN KEY (customer_id) REFERENCES customer(customer_id)

);
```

Doctors and Patients:

```
CREATE TABLE doctor (

    doctor_id INT PRIMARY KEY,

    doctor_name VARCHAR(100),

    specialization VARCHAR(100)

);
```

```
CREATE TABLE patient (

    patient_id INT PRIMARY KEY,

    patient_name VARCHAR(100),

    age INT,

    doctor_id INT,

    FOREIGN KEY (doctor_id) REFERENCES doctor(doctor_id)

);
```