

5주차 05 - 데이터베이스

CREATE

- 사용자 테이블

```
CREATE TABLE users (  
    id INT NOT NULL AUTO_INCREMENT,  
    name VARCHAR(30) NOT NULL,  
    job VARCHAR(100),  
    birth DATE,  
    PRIMARY KEY (id)  
);
```

- 게시글 테이블

```
CREATE TABLE posts(  
    id INT NOT NULL AUTO_INCREMENT,  
    title VARCHAR(100) NOT NULL,  
    content VARCHAR(2000),  
    created_at TIMESTAMP DEFAULT NOW(),  
    PRIMARY KEY (id));
```

SHOW TABLES

테이블 보여줘

DESC 테이블 이름

테이블 정보

날짜 시간 타입

시스템 시간대를 정보에 맞게 날짜를 저장

1. DATE

- 날짜만
- YYYY-MM-DD

2. DATETIME

- 날짜 + 시간
- YYYY-MM-DD-HH:MM:SS(24시간)

3. TIME

- 시간
- HH:MM:SS

4. TIMESTAMP : 자동 입력

- 날짜 + 시간
- YYYY-MM-DD-HH:MM:SS(24시간)
-

INSERT

- 사용자 데이터 삽입

```
INSERT INTO users(name, job, birth)
VALUES("gongu","actor","800123");
```

- 게시글 데이터 삽입

```
INSERT INTO posts (title, content)
VALUES("title","content");
```

ALTER : 수정

게시글 테이블에 수정일자 추가

```
ALTER TABLE posts
ADD COLUMN updated_at DATETIME
DEFAULT NOW()
ON UPDATE NOW();
```

UPDATE

```
UPDATE posts
SET content = "updates!"
WHERE id =2;
```

외래키 추가

```
ALTER TABLE posts
ADD COLUMN user_id INT;
```

```
ALTER TABLE posts
ADD FOREIGN KEY(user_id)
REFERENCES users(id);
```

JOIN

- 합쳐서 보여줌
- 전체

```
SELECT * FROM posts
LEFT JOIN users
ON posts.user_id =
users.id;
```

- 필요한 것만

```
SELECT posts.id,title,content,created_at,updated_at,name,job,birth FROM posts  
LEFT JOIN users  
ON posts.user_id =  
users.id;
```

auto_increment

- 업데이트 실패해도 숫자 자동 증가