```sql
-- create table
CREATE TABLE `user` (
    `user_id`   bigint      NOT NULL AUTO_INCREMENT,
    `login_id`  varchar(20)     NOT NULL,
    `password`  varchar(60)     NOT NULL,
    `name`      varchar(20)     NOT NULL,
    `email`     varchar(50)     NOT NULL,
    `nickname`  varchar(20)     NOT NULL,
    `admission_year`        varchar(8) NOT NULL,
    `authority`     varchar(20)     NOT NULL DEFAULT
'ROLE_BASIC',
    `registration_date`     datetime(6)     NOT NULL,
    `modification_date`     datetime(6)     NULL,
    `school_id`     bigint      NULL,
    `token_id`  bigint      NULL,
    CONSTRAINT `PK_USER` PRIMARY KEY (`user_id`),
    CONSTRAINT `unq_user_login_id` UNIQUE(`login_id`),
    CONSTRAINT `unq_email` UNIQUE(`email`),
    CONSTRAINT `unq_nickname` UNIQUE(`nickname`),
    CONSTRAINT `unq_token_id` UNIQUE(`token_id`)
);

CREATE TABLE `token` (
    `token_id`  bigint      NOT NULL AUTO_INCREMENT,
    `access_token`  varchar(1000)   NULL,
    `registration_date`     datetime(6)     NOT NULL,
    `modification_date`     datetime(6)     NULL,
    CONSTRAINT `PK_TOKEN` PRIMARY KEY (`token_id`)
);

CREATE TABLE `school` (
    `school_id`     bigint      NOT NULL AUTO_INCREMENT,
    `school_name`   varchar(30)     NOT NULL,
    `registrant`    varchar(20)     NOT NULL,
    `modifier`  varchar(20)     NULL,
    `registration_date`     datetime(6)     NOT NULL,
    `modification_date`     datetime(6)     NULL,
    CONSTRAINT `PK_SCHOOL` PRIMARY KEY (`school_id`),
    CONSTRAINT `unq_school_name` UNIQUE (`school_name`)
);
```

```sql
CREATE TABLE `post` (
    `post_id` bigint    NOT NULL AUTO_INCREMENT,
    `board_type`    varchar(20)    NOT NULL,
    `title`    varchar(100)    NOT NULL,
    `contents` varchar(15000)  NOT NULL,
    `is_anonymous`  varchar(1) NOT NULL,
    `is_deleted`    varchar(1) NOT NULL DEFAULT 'N',
    `views`    bigint    NOT NULL DEFAULT 0,
    `registration_date`  datetime(6)    NOT NULL,
    `modification_date`  datetime(6)    NULL,
    `user_id`  bigint    NULL,
    `school_id`    bigint    NOT NULL,
    CONSTRAINT `PK_POST` PRIMARY KEY (`post_id`)
);

CREATE TABLE `comment` (
    `comment_id`    bigint    NOT NULL AUTO_INCREMENT,
    `contents` varchar(15000)  NOT NULL,
    `comment_type`  varchar(10)    NOT NULL,
    `pre_id`  bigint    NULL,
    `is_anonymous`  varchar(1) NOT NULL,
    `is_deleted`    varchar(1) NOT NULL DEFAULT 'N',
    `registration_date`  datetime(6)    NOT NULL,
    `modification_date`  datetime(6)    NULL,
    `user_id`  bigint    NULL,
    `post_id`  bigint    NOT NULL,
    CONSTRAINT `PK_COMMENT` PRIMARY KEY (`comment_id`)
);

CREATE TABLE `file` (
    `file_id` bigint    NOT NULL AUTO_INCREMENT,
    `upload_filename`    varchar(300)    NOT NULL,
    `store_filename`    varchar(300)    NOT NULL,
    `size`    varchar(30)    NOT NULL,
    `sequence` bigint    NOT NULL,
    `registration_date`  datetime(6)    NOT NULL,
    `modification_date`  datetime(6)    NULL,
    `post_id`  bigint    NULL,
    `notice_id`    bigint    NULL,
    CONSTRAINT `PK_FILE` PRIMARY KEY (`file_id`)
);
```

```
CREATE TABLE `likes` (
     `like_id`  bigint    NOT NULL AUTO_INCREMENT,
     `target_type`  varchar(10)    NOT NULL,
     `target_id`    bigint    NOT NULL,
     `registration_date`  datetime(6)    NOT NULL,
     `modification_date`  datetime(6)     NULL,
     `user_id`  bigint    NOT NULL,
   CONSTRAINT `PK_LIKES` PRIMARY KEY (`like_id`)
);

CREATE TABLE `manager` (
     `manager_id`    bigint    NOT NULL AUTO_INCREMENT,
     `login_id` varchar(20)    NOT NULL,
     `password` varchar(60)    NOT NULL,
     `name`     varchar(20)    NOT NULL,
     `authority`    varchar(20)    NOT NULL DEFAULT
'ROLE_MANAGER',
     `registrant`    varchar(20)     NOT NULL,
     `modifier` varchar(20)     NULL,
     `registration_date`  datetime(6)    NOT NULL,
     `modification_date`  datetime(6)     NULL,
     `token_id` bigint     NULL,
   CONSTRAINT `PK_MANAGER` PRIMARY KEY (`manager_id`),
   CONSTRAINT `unq_mng_login_id` UNIQUE (`login_id`),
   CONSTRAINT `unq_mng_token_id` UNIQUE(`token_id`)
);

CREATE TABLE `notice` (
     `notice_id`     bigint    NOT NULL AUTO_INCREMENT,
     `title`    varchar(100)    NOT NULL,
     `contents` varchar(15000)  NOT NULL,
     `views`    bigint    NOT NULL DEFAULT 0,
     `is_deleted`    varchar(1) NOT NULL DEFAULT 'N',
     `registration_date`  datetime(6)    NOT NULL,
     `modification_date`  datetime(6)     NULL,
     `manager_id`    bigint     NULL,
   CONSTRAINT `PK_NOTICE` PRIMARY KEY (`notice_id`)
);


-- FK setting
ALTER TABLE `user` ADD CONSTRAINT `FK_USER_SHCHOOL`
FOREIGN KEY (`school_id`) REFERENCES `school` (`school_id`)
ON UPDATE CASCADE ON DELETE SET NULL;

ALTER TABLE `user` ADD CONSTRAINT `FK_USER_TOKEN`
FOREIGN KEY (`token_id`) REFERENCES `token` (`token_id`)
ON UPDATE CASCADE ON DELETE SET NULL;
```

```sql
ALTER TABLE `likes` ADD CONSTRAINT `FK_LIKES_USER`
FOREIGN KEY (`user_id`) REFERENCES `user` (`user_id`)
ON UPDATE CASCADE ON DELETE CASCADE;


ALTER TABLE `post` ADD CONSTRAINT `FK_POST_USER`
FOREIGN KEY (`user_id`) REFERENCES `user` (`user_id`)
ON UPDATE CASCADE ON DELETE SET NULL;

ALTER TABLE `post` ADD CONSTRAINT `FK_POST_SCHOOL`
FOREIGN KEY (`school_id`) REFERENCES `school` (`school_id`)
ON UPDATE CASCADE ON DELETE CASCADE;


ALTER TABLE `comment` ADD CONSTRAINT `FK_COMMENT_USER`
FOREIGN KEY (`user_id`) REFERENCES `user` (`user_id`)
ON UPDATE CASCADE ON DELETE SET NULL;

ALTER TABLE `comment` ADD CONSTRAINT `FK_COMMENT_POST`
FOREIGN KEY (`post_id`) REFERENCES `post` (`post_id`)
ON UPDATE CASCADE ON DELETE CASCADE;


ALTER TABLE `manager` ADD CONSTRAINT `FK_MANAGER_TOKEN`
FOREIGN KEY (`token_id`) REFERENCES `token` (`token_id`)
ON UPDATE CASCADE ON DELETE SET NULL;


ALTER TABLE `notice` ADD CONSTRAINT `FK_NOTICE_MANAGER`
FOREIGN KEY (`manager_id`) REFERENCES `manager` (`manager_id`)
ON UPDATE CASCADE ON DELETE SET NULL;


ALTER TABLE `file` ADD CONSTRAINT `FK_FILE_POST`
FOREIGN KEY (`post_id`) REFERENCES `post` (`post_id`)
ON UPDATE CASCADE ON DELETE CASCADE;

ALTER TABLE `file` ADD CONSTRAINT `FK_FILE_NOTICE`
FOREIGN KEY (`notice_id`) REFERENCES `notice` (`notice_id`)
ON UPDATE CASCADE ON DELETE CASCADE;
```