

Schema - Create schema db_cms;

1. Role

```
DROP TABLE IF EXISTS `db_cms`.`role`;
```

```
CREATE TABLE `db_cms`.`role` (  
  `id` INT NOT NULL,  
  `role` NVARCHAR(50) NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE INDEX `role_UNIQUE` (`role` ASC) VISIBLE);
```

```
INSERT INTO `db_cms`.`role` (`id`, `role`) VALUES (1, 'Admin');  
INSERT INTO `db_cms`.`role` (`id`, `role`) VALUES (2, 'User');
```

2. User

```
DROP TABLE IF EXISTS `db_cms`.`user`;
```

```
CREATE TABLE `db_cms`.`user` (  
  `id` INT NOT NULL,  
  `name` NVARCHAR(50) NULL,  
  `password` NVARCHAR(50) NULL,  
  `email` NVARCHAR(50) NULL,  
  `role_id` INT NOT NULL,  
  PRIMARY KEY (`id`),  
  INDEX `role_id_idx` (`role_id` ASC) VISIBLE,  
  CONSTRAINT `role_id`  
    FOREIGN KEY (`role_id`)  
    REFERENCES `db_cms`.`role` (`id`)  
    ON DELETE CASCADE  
    ON UPDATE CASCADE);
```

3. Category

```
DROP TABLE IF EXISTS `db_cms`.`category`;
```

```
CREATE TABLE `db_cms`.`category` (  
  `id` INT NOT NULL,  
  `category` NVARCHAR(50) NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE INDEX `category_UNIQUE` (`category` ASC) VISIBLE);
```

```
INSERT INTO `db_cms`.`category` (`id`, `category`) VALUES ('1', 'T&M');
INSERT INTO `db_cms`.`category` (`id`, `category`) VALUES ('2', 'FMB');
INSERT INTO `db_cms`.`category` (`id`, `category`) VALUES ('3', 'FB');
```

4. Visibility

```
DROP TABLE IF EXISTS `db_cms`.`visibility`;
```

```
CREATE TABLE `db_cms`.`visibility` (  
  `id` INT NOT NULL,  
  `visibility` NVARCHAR(50) NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE INDEX `visibility_UNIQUE` (`visibility` ASC) VISIBLE);
```

```
INSERT INTO `db_cms`.`visibility` (`id`, `visibility`) VALUES ('1', 'Public');  
INSERT INTO `db_cms`.`visibility` (`id`, `visibility`) VALUES ('2', 'Private');
```

5. Template

```
DROP TABLE IF EXISTS `db_cms`.`template`;
```

```
CREATE TABLE `db_cms`.`template` (  
  `id` INT NOT NULL,  
  `name` NVARCHAR(50) NULL,  
  `category_id` INT NOT NULL,  
  `visibility_id` INT NOT NULL,  
  `source_code` TEXT(100) NULL,  
  `last_modification_date` DATETIME NULL,  
  `modified_by` INT NOT NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE INDEX `name_UNIQUE` (`name` ASC) VISIBLE,  
  INDEX `category_id_idx` (`category_id` ASC) VISIBLE,  
  INDEX `visibility_id_idx` (`visibility_id` ASC) VISIBLE,  
  CONSTRAINT `category_id`  
    FOREIGN KEY (`category_id`)  
    REFERENCES `db_cms`.`category` (`id`)  
    ON DELETE CASCADE  
    ON UPDATE CASCADE,  
  CONSTRAINT `visibility_id`  
    FOREIGN KEY (`visibility_id`)  
    REFERENCES `db_cms`.`visibility` (`id`)  
    ON DELETE CASCADE
```

```
ON UPDATE CASCADE,  
CONSTRAINT `modified_by`  
FOREIGN KEY (`modified_by`)  
REFERENCES `db_cms`.`user` (`id`)  
ON DELETE CASCADE  
ON UPDATE CASCADE);
```

6. Tags

```
DROP TABLE IF EXISTS `db_cms`.`tags`;
```

```
CREATE TABLE `db_cms`.`tags` (  
  `id` INT NOT NULL,  
  `tag` NVARCHAR(50) NULL,  
  `template_id` INT NULL,  
  PRIMARY KEY (`id`),  
  INDEX `template_id_idx` (`template_id` ASC) VISIBLE,  
  CONSTRAINT `template_id`  
  FOREIGN KEY (`template_id`)  
  REFERENCES `db_cms`.`template` (`id`)  
  ON DELETE CASCADE  
  ON UPDATE CASCADE);
```

7. Document

```
DROP TABLE IF EXISTS `db_cms`.`document`;
```

```
CREATE TABLE `db_cms`.`document` (  
  `id` INT NOT NULL,  
  `name` NVARCHAR(50) NULL,  
  `source_code` TEXT(100) NULL,  
  `template_id` INT NOT NULL,  
  `creation_date` DATETIME NULL,  
  `created_by` INT NOT NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE INDEX `name_UNIQUE` (`name` ASC) VISIBLE,  
  INDEX `template_id_idx` (`template_id` ASC) VISIBLE,  
  CONSTRAINT `dtemplate_id`  
  FOREIGN KEY (`template_id`)  
  REFERENCES `db_cms`.`template` (`id`)  
  ON DELETE CASCADE  
  ON UPDATE CASCADE,  
  CONSTRAINT `created_by`
```

```
FOREIGN KEY (`created_by`)  
REFERENCES `db_cms`.`user` (`id`)  
ON DELETE CASCADE  
ON UPDATE CASCADE);
```

8. Document_Files

```
DROP TABLE IF EXISTS `db_cms`.`document_files`;
```

```
CREATE TABLE `db_cms`.`document_files` (  
  `id` INT NOT NULL,  
  `file_type` NVARCHAR(50) NULL,  
  `content` BLOB NULL,  
  `document_id` INT NOT NULL,  
  PRIMARY KEY (`id`),  
  INDEX `document_id_idx` (`document_id` ASC) VISIBLE,  
  CONSTRAINT `document_id`  
    FOREIGN KEY (`document_id`)  
    REFERENCES `db_cms`.`document` (`id`)  
    ON DELETE CASCADE  
    ON UPDATE CASCADE);
```

9. Element_Type

```
DROP TABLE IF EXISTS `db_cms`.`element_type`;
```

```
CREATE TABLE `db_cms`.`element_type` (  
  `id` INT NOT NULL,  
  `type` NVARCHAR(50) NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE INDEX `type_UNIQUE` (`type` ASC) VISIBLE);
```

```
INSERT INTO `db_cms`.`element_type` (`id`, `type`) VALUES ('1', 'Clause');
```

10. Element

```
DROP TABLE IF EXISTS `db_cms`.`element`;
```

```
CREATE TABLE `db_cms`.`element` (  
  `id` INT NOT NULL,  
  `element_type_id` INT NOT NULL,  
  `name` NVARCHAR(50) NULL,
```

```
`source_code` TEXT(100) NULL,  
PRIMARY KEY (`id`),  
UNIQUE INDEX `name_UNIQUE` (`name` ASC) VISIBLE,  
INDEX `element_type_id_idx` (`element_type_id` ASC) VISIBLE,  
CONSTRAINT `element_type_id`  
  FOREIGN KEY (`element_type_id`)  
    REFERENCES `db_cms`.`element_type` (`id`)  
  ON DELETE CASCADE  
  ON UPDATE CASCADE);
```