

GUDI EDU 개발표준정의서

개발 원칙

- 주석은 모든 코드에 기능을 기술하는 것을 원칙으로 함
- 소스 코드는 불가피한 내용을 제외하고 원칙적으로 중복을 금지함
- 소스 코드에서 사용하는 모든 단어는 용어사전의 내용을 기준으로 함

자바

- Java file의 명명규칙은[userType][[용어사전](#)][postfix].java로 한다.
- 용어사전의 단어를 사용하는 경우 첫 글자만 대문자로 한다.(UpperCamelCase)
- postfix는 framework에서 정의하는 class 요소에 대한 구분으로 해당 java class의 성격을 나타낸다.
 - ex) postfix
 - Controller, Wrapper, Service, ServiceImpl, Cmd, DAO, VO, Test 등

interface Class

- Interface는 Java 클래스명과 동일한 방법으로 부여한다.

자바 메소드

- java method의 명명규칙은 [userType][용어사전][기능_동사]로 한다.
 - 클래스 명명은 필수로 하되 메소드 명명규칙은 용어사전을 최대한 준용하도록 한다.
 - 첫 글자는 소문자를 사용하며 이후 용어의 첫 글자만 대문자를 사용하며, 숫자 및 특수문자는 사용하지 않는다.(lowerCamelCase)

변수 및 상수

- Java 변수의 명명규칙은 [용어사전]을 조합하여 30자 이내로 명명한다.
- 첫 글자는 소문자를 사용하며 이후 용어의 첫 글자만 대문자를 사용하며, 숫자 및 특수문자는 사용하지 않는다.
- loop index에서 사용하는 변수는 i, j, k, l, x, y, z 등을 (관용적으로) 사용할 수 있다.
- 일반 변수는 “_” 또는 “\$” 사용을 하지 않는다.
 - 단, 데이터베이스의 속성명을 그대로 사용하는 경우에는 ‘_’ 사용을 허용.
 - 상수는 Static Final 변수는 용어사전을 사용하여 대문자로만 작성하며 단어 사이는 ‘_’를 사용하여 구분한다.

SQL

- SQL 예약어는 대문자를 사용한다.
- SELECT, UPDATE시 컬럼 나열은 1줄당 1개씩 소문자로 작성을 기본으로 하며, 컬럼이 많은 경우에 한해 1줄에 3개 컬럼까지 쓰는 것을 예외로 허용한다.
- SELECT의 들여쓰기는 SELECT를 기준으로 FROM, WHERE, INTO등의 시작점을 맞춘다.
 - 단, GROUP BY나 ORDER BY는 SELECT(6자)보다 더 긴 관계로 BY 이후 스페이스 1칸을 띄우고 관련내용을 기술한다.
- FROM 절의 테이블명은 1줄당 1개씩 소문자로 작성하는 것을 기본으로 하며, 테이블이 많은 경우에 한하여서, 1줄당 2개의 테이블까지 적는 것을 예외로 허용한다.
- WHERE 절의 조건은 1줄당 1개의 조건씩을 작성한다.
- INSERT 문장의 경우 전체 컬럼에 대해서 insert를 하더라도 컬럼명을 반드시 적어준다.
- DML의 들여쓰기는 UPDATE, DELETE, INSERT를 기준으로 SET, FROM, WHERE등의 시작점을 맞추고, 컬럼 구분은 가장 길이가 긴 컬럼의 쉼표(,)나 등호(=등)을 기준으로 다른 컬럼을 맞춘다.
 - 괄호를 사용할 경우, 즉 인라인 뷰, 스칼라 서브쿼리 등을 사용할 경우 괄호가 존재하는 라인은 라인을 변경하지 않는다.
 - 줄 바꿈을 일정하게 한다.
 - 콤마(,) 후에는 반드시 공백을 쓴다.
 - 계속되는 콤마는 줄 앞에 위치시킨다.
 - 연산자를 기준으로 양쪽에 공백을 준다.
 - SQL문 안에는 comment를 사용하지 않는다.

Vue.js

- 표기법은 Vue.js의 [공식 문서](#)에 기재된 작성법을 따른다.
- 컴포넌트
 - 여러 단어를 사용하여 컴포넌트 이름을 작성할 경우 케밥표기법(Kebab Case)를 사용하여 작성하며, 최대 3단어를 사용하여 만든다.
 - 컴포넌트의 이름은 약어를 사용하기 보다 전체 단어를 기재하는 것을 원칙으로 한다.
- 파일이름
 - 파일이름은 항상 PascalCase를 사용하여 작성한다.
- 메소드
 - 메소드 이름은 항상 lowerCamelCase를 사용하여 작성한다.

HTML

- ID

- 영어, 숫자, 특수문자 중 일부만 인식이 가능하다. **id**는 전체에서 1개만 사용 가능하다.
- 첫 시작에는 가능한 문자로 시작하고, 숫자, 특수문자, 대문자의 사용은 지양한다.
- 띄어쓰기가 필요한 부분에는 다음 문자의 앞글자를 대문자로 사용하거나 언더바를 사용한다.
- 위치, 역할, 형태를 조합하며 상태(e.g. on/ off)에 대한 부분도 필요시 추가하며, 내용이 많아질 경우, 위치와 상태를 나누어 클래스를 2개로 작성한다.
- 작성 시 케밥표기법(Kebab Case)로 다음단어와 '-'(대시)로 구분한다.
 - user-name, user-age, phone-number

URL 규칙

- url은 database에 있는 'tm_mnu_mst'의 데이터 값을 사용한다.