|  |
| --- |
| <http://blog.naver.com/saintsad/100023879440> |
|  |
|  |
| |  |  |  |  | | --- | --- | --- | --- | | 고객명 | 나빈 | 작성일자 | 2003.04.22 | | 클라이언트 | 나빈 | 담당자 | 곽광철 | | Activity |  | 작성자 | 곽광철 |   ***기본 코딩 규칙.***  1 . 주석을 이용하여 반드시 설명을 달아두도록 한다.  향후 다른 사람이 작업을 하게 되어도 주석이 잘 되어 있으면 특별한 수고 없이 수정 및 보완이 용이  2 . GUI 툴은 가급적 사용하지 않는다  쓸데없는 테그로 인한 속도 저하 및 구조를 사전에 방지한다.  만일 사용한다고 해도 반드시 정리를 한다.  3 . 의미있는 파일명 및 체계적인 디렉토리 구조를 사용한다.  파일명만 보아도 어떤때 사용되는 파일인지 명확히 구분이 되어야 한다.  4 . 변수명 규칙에 따라 변수명을 사용한다.  변수명 규칙은 Hungarian 표기법과 PascalCasing Naming Convention을 따르는 것을 원칙으로 한다.  5 . include 파일을 사용한다.  Include을 사용하여 유지 보수 및 개발을 용이하게 한다.  6 . Option Explicit 문을 사용한다.  변수선언을 해야 중복된 변수 및 에러 발생시 디버깅이 용이하다.  7. ASP 페이지 상단에 모든 변수를 선언한다.  가장 처음 시작되는 부분에 변수를 선언해야 가독성 및 이해가 쉽다.  8 . 데이터베이스 접속은 인클루드 파일을 사용해야 한다.  접속때 마다 열어주고 닫아주게 되면 수정 및 개발에 불편이 따른다.  9 . 객체 인스턴스의 생성과 소멸은 명확히 한다.  자동으로 객체를 소멸해 주는 것을 기대하기 보다는 직접 생성과 소멸해 주어야 한다.   |  |  | | --- | --- | | Presentation Services(ASP) | ASP File Name, HTML File Name, Image File Name, Session Level  Variable Name(PascalCasing Naming Conversion or CamelCasing  Naming Conversion), Script Variable Name, Tag Name, Cookie  Name | | Business Services  (Component) | Project File Name(DLL Name), Class Name, Interface(Method,  Property) Name | | Data Services(Database) | Table Name, Field Name, Stored Procedure Name, View Name,  Trigger Name |   모든 이름은 Hungarian 표기법과 PascalCasing Naming Convention을 따르는 것을 원칙으로 한다.  (마지막장 표 참조)  ***Presentation Services Naming Rule***  1. ASP File Name  ASP 파일 명은 관리를 쉽게 할 수 있도록 업무에 관련된 이름을 부여한다.  [명명 규칙] 시스템단축명\_기능.asp  예) 웹메일 시스템의 자유 게시판의 ASP 파일 명은 wm\_FreeBoard.asp로 명명한다.  시스템 단축 명은 프로젝트 관리자가 부여한다.  2. HTML File Name  HTML 파일 명은 관리를 쉽게 할 수 있도록 업무에 관련된 이름을 부여하며, HTML 파일의 확장자는 htm로 명명한다.  [명명 규칙] 시스템단축명\_기능.htm  예) 웹메일 시스템의 자유 게시판의 HTML 파일 명은 wm\_FreeBoard.htm으로 명명한다.  시스템 단축 명은 프로젝트 관리자가 부여한다.  3. Image File Name  Image 파일 명은 다음과 같은 이름 부여 규칙을 사용한다.  [명명 규칙] 업무단축명\_위치\_기능.gif  업무 단축명은 가능하다면 ASP 파일의 단축 명을 사용한다.  위치 : 상(t), 중(m), 하(b), 좌(l), 중(m), 우(r)  예1) 웹메일 시스템의 자유 게시판의 우측상단에 있는 목록보기 이미지는 fb\_tr\_list.gif로 명명한다.  여기서 fb는 FreeBoard의 단축 명이다.  예2) 메인 페이지의 좌측 중앙에 있는 로그온 이미지는 main\_ml\_LogOn.gif로 명명한다.  웹 응용 프로그램 전반에 공통적으로 사용되는 이미지는 명명 규칙에 “위치”를 포함하지 않으며,  “업무단축명”으로 “cn\_”을 사용한다.  예3) 메인 화면에 사용하는 읽기에 대한 공통 이미지는 cn\_read.gif로 명명한다.  4. Application Level Variable Name  Application Level의 변수는 다음과 같은 이름 부여 규칙을 사용한다.  [명명 규칙] ap\_기능  예) 웹 사이트 방문자 수를 나타내는 Application Level 변수는 ap\_VisitCount로 명명한다.  5. Session Level Variable Name  Session Level의 변수는 다음과 같은 이름 부여 규칙을 사용한다.  [명명 규칙] se\_기능  예) 사용자 계정을 나타내는 Session Level 변수는 se\_UserID로 명명한다.  6. Script Variable Name  스크립트 변수 명은 변수의 Scope에 따라 접두어(“g\_” 전역, “l\_” 로컬, “p\_” 매개변수)를 붙인다.  단, 로컬 변수인 경우에는 Scope를 나타내는 접두어는 생략 가능하다.  [명명 규칙] 접두어\_기능  예) 사용자가 선택한 색상을 저장하는 로컬 변수는 “l\_Color” 이다.  스크립트에서는 모든 데이터 타입이 Variant이기 때문에, 변수 명에 데이터 형을 표기하지 않는다.  7. Tag Name  태그 명은 태그의 종류를 나타내는 접두어와 기능으로 이름을 부여한다.  [명명 규칙] 접두어\_기능  태그는 종류별로 다음과 같은 접두어를 붙인다.  접두어 태그  frm 폼  btn\_ 버튼  txt\_ 텍스트 박스  mtx\_ 멀티라인 텍스트 박스  lbl\_ 라벨  chk\_ 체크박스  opt\_ 옵션버튼  이미지 파일은 확장자로 알 수 있기 때문에 태그의 접두어를 사용하지 않는다.  여기서는 주로 많이 사용하는 태그들의 이름 부여 규칙만을 정의하며,  그 외의 태그들은 개발자가 임의로 이름을 부여해서 사용한다.  예) 사용자 이름을 입력하는 텍스트 박스의 명은 txt\_UserName으로 명명한다.  8. Cookie Name  Cookie 명은 관리를 쉽게 할 수 있도록 업무에 관련된 이름을 부여한다.  [명명 규칙] 시스템단축명\_기능  예) 웹메일 시스템의 사용자 이름을 저장하는 쿠키의 명은 wm\_UserName으로 명명한다.  서브 쿠키 명은 개발자가 임의로 부여한다.  ***Business Services Naming Rule***  1. Visual Basic Project File Name (DLL File Name)  DLL 파일 명은 관리를 쉽게 할 수 있도록 업무에 관련된 이름을 부여한다.  [명명 규칙] 시스템단축명\_기능.dll  예) 웹메일 시스템의 자유 게시판을 구현하는 DLL의 파일 명은 wm\_FreeBoard.dll로 명명한다.  시스템 단축 명은 프로젝트 관리자가 부여한다.  2. Class Name  비주얼 베이직 프로젝트에 포함된 클래스 파일의 이름은 기본적으로 Query와 Modify로 분류하고,  별도의 비즈니스 로직을 포함하는 클래스는 프로젝트 관리자와 협의 후 부여한다.  Query 클래스는 조회용 메서드를 포함하는 클래스의 명칭이며,  Modify 클래스는 삽입, 수정, 삭제와 같은 트랜잭션이 발생하는 메서드를 포함하는 클래스의 명칭이다.  [명명 규칙] Query, Modify, 업무명  예) 웹메일 시스템의 자유 게시판의 글을 읽는 클래스 파일의 이름은 Query로 명명하며,  ASP에서 사용하기 위한 이 클래스의 ProgID는 wm\_FreeBoard.Query이다.  일반적으로 Query와 Modify 이외의 클래스 명을 요하는 경우는 거의 발생하지 않는다.  3. Interface Name (Method, Property)  인터페이스 명은 메서드에 대해서만 기술한다.  MS의 COM+에 등록되는 컴포넌트는 Stateless한 컴포넌트이어야 하기 때문에,  일반적인 경우에는 Property를 사용하지 않는다.  개발팀 별로 메서드에 대한 공통 인터페이스를 정의해야 하며, 정의된 공통 인터페이스를  기준으로 메서드의 이름을 부여해야 한다.  ***Data Services Naming Rule***  데이터베이스 명은 별로의 이름 부여 규칙을 정하지 않고 프로젝트 관리자가 시스템에 적합하게  부여한다.  1. Table Name  테이블 명은 관리를 쉽게 할 수 있도록 업무에 관련된 이름을 부여하며,  레코드들의 집합이기 때문에 복수형으로 부여한다.  [명명 규칙] 기능의 복수형  예) 웹메일 시스템의 자유 게시판의 테이블 명은 FreeBoards로 명명한다.  2. Field Name  필드 명은 기능과 관련된 의미로 부여하며, 업무에 관련된 의미는 테이블 명에서 이미 표기하였기 때문에  업무에 관련된 의미는 부여하지 않는다.  [명명 규칙] 기능  예) 웹메일 시스템의 자유 게시판의 날짜 필드 명은 Date로 명명한다.  FreeBoardDate로 표기하지 않도록 주의한다.  3. Stored Procedure Name  저장 프로시저 명은 기준이 되는 테이블 명과 기능과 관련된 의미로 부여한다.  시스템 저장 프로시저와 구별하기 위해서 접두어 “up\_”를 붙인다.  [명명 규칙] up\_테이블명\_기능  예) 웹메일 시스템의 자유 게시판의 쓰기 기능의 저장 프로시저 명은 up\_FreeBoards\_Insert로 명명한다.  4. View Name  뷰 명은 기준이 되는 테이블 명을 단수형으로 부여한다. 뷰를 나타내기 위해서 접두어 “view\_”를 붙인다.  [명명 규칙] view\_테이블명의단수형  예) 웹메일 시스템의 자유 게시판의 데이터를 일자를 조건으로 조회하기 위해서 사용될 뷰의 명은  view\_FreeBoard로 명명한다.  5. Trigger Name  트리거 명은 기준이 되는 테이블 명과 기능으로 부여한다.  [명명 규칙] 테이블명\_기능  예) 웹메일 시스템의 자유 게시판의 데이터를 삭제할 경우에 발생할 트리거의 명은  FreeBoards\_Delete로 명명한다.  ***소스 파일 구성***  SQL 파일  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  /\*\*\*\* 제 목 : sample 정보 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  /\*\*\*\* 파일명 : tb\_sample.sql \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  /\*\*\*\* 작성일 : 2002/11/25 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  /\*\*\*\* 작성자 : 곽광철 [ [master@nabean.com](mailto:master@nabean.com) ] \*\*\*/  /\*\*\*\* 설 명 : Sample 테이블 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  /\*\*\*\* 수 정 : 2002/11/26 곽광철 컬럼 추가 \*\*\*\*\*/  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  CREATE TABLE tb\_sample /\* 테이블 생성 \*/  (  no INT IDENTITY(1,1) NOT NULL /\* 증가값 \*/  ,DATE DATETIME NOT NULL DEFAULT GETDATE() /\* 저장 일자 \*/  ,SAMPLE VARCHAR(20) NOT NULL DEFAULT 'SAMPLE' /\* 1번 컬럼 \*/  ,col1 VARCHAR(10) NULL /\* 2번 컬럼 \*/  ,col2 VARCHAR(10) NULL /\* 3번 컬럼 \*/  )  GO  Stored Procedure 파일  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  /\*\*\*\* 제 목 : 회원정보 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  /\*\*\*\* 파일명 : up\_sample.sql \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  /\*\*\*\* 작성일 : 2002/11/25 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  /\*\*\*\* 작성자 : 곽광철 [ [master@nabean.com](mailto:master@nabean.com) ] \*\*\*/  /\*\*\*\* 설 명 : Sample 테이블 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  /\*\*\*\* 수 정 : 2002/11/26 곽광철 변수 추가 \*\*\*\*\*/  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  CREATE PROC up\_sample  @sample1 VARCHAR(20),  @sample2 VARCHAR(20)  WITH ENCRYPTION  AS  SET NOCOUNT ON  BEGIN  SELECT sample FROM tb\_sample  WHERE col1 = @sample1 AND col2 = @sample2  END  GO  ASP 소스 파일  <%@ LANGUAGE="VBSCRIPT" %>  <% Option Explicit %>  <!--METADATA TYPE="typelib" UUID="00000205-0000-0010-8000-00AA006D2EA4" NAME="ADODB Type Library" -->  <!-- #include virtual="/conn/dbconn.asp" -->  <%  ‘ 제 목 : 샘플  ‘ 파일명 : sample.asp  ‘ 작성일 : 2002/11/26  ‘ 작성자 : 곽광철 [ master@nabean.com ]  ‘ 설 명 : 샘플  ‘ 수 정 : 2002/12/26 곽광철 요소 추가  %>  <%  Dim test, test2 ‘ 각종 변수선언  test = Request.Form(“f\_test”) ‘ 폼 값  %>  <HTML>  <HEAD>  <TITLE>::: Nabean :::</TITLE>  <meta http-equiv="Content-Type" content="text/html; charset=euc-kr">  <link rel="stylesheet" href="/css/style.css">  <script language="javascript" src="/css/script.js"></script>  </HEAD>  <BODY marginwidth=”0” marginheight=”0” topmargin=”0” leftmargin=”0” bgcolor=”#FFFFFF”>  <!-- 본문 시작 -->  <!-- 본문 끝 -->  </BODY>  </HTML>  ***Hungarian 표기법***   |  |  |  | | --- | --- | --- | | **Prefix** | **Type** | **Example** | | B | boolean | bool bStillGoing; | | C | character | char cLetterGrade; | | str | C++ String | string strFirstName; | | si | short integer | short siChairs; | | i | integer | int iCars; | | li | long integer | long liStars; | | f | floating point | float fPercent; | | d | double-precision floating point | double dMiles; | | ld | long double-precision floating point | long double ldLightYears; | | sz | Old-Style Null Terminated String | char szName[NAME\_LEN]; | | if | Input File Stream | ifstream ifNameFile; | | is | Input Stream | void fct(istream &risIn); | | of | Output File Stream | ofstream ofNameFile; | | os | Output Stream | void fct(ostream &rosIn); | | S | declaring a struct | struct SPoint { | | C | declaring a class | class CPerson { | | struct name or abbrev | declaring an instance of a struct | SPoint pointLeft;  SPoint ptLeft; // or abbrev. (be consistent) | | class name or abbrev | declaring an instance of a class | CPerson personFound; CPerson perFound; // or abbrev. (be consistent) | |

출처 : [Tong - ddakzzi님의 Document통](http://tong.nate.com/boxitem/post.do?@idx=3&@base=38912230&@rid=&@sw=1&_tongID=90103&_boxID=1936589&action=clist)