# Leszynski Naming Convention

To make managing objects easier, it is important to use a naming convention. While not required by Microsoft Access, a naming convention will make your work easier to understand and easier to share.

Probably the most popular naming convention among Access designers is the Leszynski Naming Convention, also known as the LNC. The LNC, like most database naming conventions, makes use of tags.

These tags are usually three letters that precede (and are part of) names of entities in a database, including objects and fields.

In all honesty, it really doesn't matter which naming convention you choose, as long as you do pick one and stick to it! At the very minimum, use a convention for the names of objects (tables, queries, forms, reports, and macros) in your database.

In an employees database, it is likely you would need a table about employees, and a query about employees, and a form for employees, and a report... you get the idea.

A naming convention will allow you to know the type of object just by looking at the name, tblEmployees, qryEmployees, frmEmployees, rptEmployees, etc. If programming is ever done in the database, it will be unnecessarily complicated if a naming convention is not used.

Naming conventions also make your work look more professional.

The tags are always lowercase and the rest of the name begins with an uppercase letter. Object names and field names are "camel backed", meaning there are no spaces, and each word begins with an uppercase letter.

**Object names should:**

* Begin with the appropriate tag
* Be "camel-backed"
* Contain only letters or numbers (no spaces, special characters, punctuation, or underscores)

Avoid using the following characters:

,.~!@#$%^&\*()+-=|\:;'<>?/"`[]{}

in *any* name in a database!

## And the LNC tags are...

**Objects:**

|  |  |
| --- | --- |
| Table | tbl |
| Query | qry |
| Form | frm |
| Report | rpt |
| Macro | mcr |
| Module | bas or mdl or mod |

**Fields:**

|  |  |
| --- | --- |
| Binary | bin |
| Byte | byt |
| Currency | cur |
| Date/Time | dtm |
| Double | dbl |
| Integer | int |
| Long Integer | lng |
| Memo | mem |
| OLE | ole |
| Single | sng |
| Text | str |
| Yes/No | ysn |

**Controls:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | Label | lbl | | Text Box | txt | | Option Group | opt | | Toggle Button | tgb | | Option Button | opb | | Combo Box | cbo | | List Box | lst | | Command Button | cmd | | |  |  | | --- | --- | | Image | img | | Tab Control | tab | | Unbound Object Frame | uof | | Bound Object Frame | bof | | Subform | sbf | | Subreport | sbr | | Line | lin | | Rectangle | rct | |

Again, it isn't at all important *which* naming convention you use. But it is important that you use one.

-- SQL script is case insensitive, linefeed insensitive (except for the comments)

use Northwind

SELECT

--select top 2

-- some basic data reading plus some functions

EmployeeID,

UPPER (Firstname),

Upper(Firstname) As Firstname, --some SQL version need column tables to be unique

left(firstname, 3) as chap, --NOTE: function syntax can be variant in different versions of SQL

right(lastname, 4) as raast,

substring(lastname, 3,1) as vasat,

FirstName as GivenName, --ANSI STANDARD (GOOD, More popular)

Surname=LastName, --Transact propiertary aliasing (BAD)

FirstName +' '+LastName as Full\_Name, --Creates the new field. Des NOT modify the original data, just the result/output

City,

birthdate,

FORMAT(birthdate, 'yy/MM/dd') as nice\_birth,

format(hiredate, 'dddd') as week\_day,

DATEPART(year,hiredate) as saal,

year(hiredate) as saneh, --depends on SQL version

DATEADD(year,5,hiredate) as tarfi,

DATEDIFF(year,birthdate, hiredate) as senn,

DATEPART(MONTH,hiredate) as maah,

CONVERT(nvarchar,birthdate,109) as pretty,

CONVERT(nvarchar,birthdate,102) as pretty2,

CONVERT(nvarchar,birthdate,106) as pretty3,

CONVERT(nvarchar,DATEADD(year,5,hiredate),106) as pretty4,

hiredate,

Country as [keshvar], --arbitrary string can be put in brackets

City as "the city I was born", -- double quotation also works as brackets BAD STANDARD

-- a powerful tool for standardizing spelling variation

case

when(country in('USA','Canada')) then 'America'

when(country in ('UK','France')) then 'Europe'

else null

end

as Continent,

--conditional (such as @if in Excel)

iif(country in('USA','America'), 'domestic','foreign') as domestic,

-- arithmetics

3\*DATEPART(MONTH,hiredate)+5 AS maheno,

--subquery

'Earth' as planet,

(select count(\*) from customers) as shomaresh

FROM

Employees --this is table name

-- FROM [Tokyo\downtown]Northwind.dbo.Employees AS tabs 'when you are on different SQL databases

-- in SELECT section you can write SELECT tabs.Employees

--where

-- (Country='USA')

-- or (Country='Canada')

-- or (Country='Mexico')

--country IN ('USA', 'Canada', 'Mexico')

--country IN (SELECT country FROM Employees GROUP BY country) 'relational: will update the output accordingly

--where

--(firstname like '\_A%')

--where

--(firstname like '[m,n][a,e]%')

--GROUP BY

--HAVING

ORDER BY

continent,

Country,

city

use Northwind

select char(9)+char(9)+'Custs.'+[name]+', ' from sys.columns where ([object\_id]=object\_id (N'dbo.Customers')) order by column\_id;

select

Custs.CustomerID,

Custs.CompanyName,

Custs.ContactName,

Custs.ContactTitle,

Custs.Address,

Custs.City,

Custs.Region,

Custs.PostalCode,

Custs.Country,

Custs.Phone,

Custs.Fax

from dbo.Customers As Custs;