

@TELITAB

The **string parameter** attribute @TELITAB is used to define that the value is [TeLiTab](#) data.

Syntax

Place @TELITAB in the data slot of the parameter

Remarks

- String parameters that have '#' as last character will always be considered to have Telitab data as value, except when the [@NOTELITAB](#) attribute is provided.
- The @TELITAB attribute gives a string parameter the properties of a [TeLiTab](#) data set, i.e. a [Quaestor](#) object in string format. [TeLiTab](#) strings can be used as 'dead' objects, they can be created only once and contain no model. It is possible to address data in a 'string' [TeLiTab](#) with constructions like $Y = A\#.X$. These [TeLiTab](#) values can be used as data source of many [intrinsic functions](#) of [Quaestor](#) like the [UNION#\(\)](#), [SPLIT\(\)](#) etc.
- In many cases, string parameters are recognised by [Quaestor](#) as [TeLiTab](#) strings, in an expression such as $A\# = \text{Curve}(@Y)$, $A\# = \text{Documents}$. $\text{ReportSet\$}$ or when functions like [UNION#\(\)](#), [SECTION#\(\)](#) or [QUERY#\(\)](#) are used in the right hand term, the parameter $A\#$ is immediately recognised as [TeLiTab](#), so without the @TELITAB attribute.
- The @TELITAB attribute is useful and sometimes necessary if [TeLiTab](#) sets are created by means of relations but not immediately recognised as such by [Quaestor](#). By adding the attribute, you are sure that this is the case and that, for instance, the [TeLiTab](#) editor in the [Workbase](#) will work to show the data inside the [TeLiTab](#)

Please note the suffix of a parameter can be used to implicitly determine the type of a parameter. The standard suffix for [TeLiTab](#) string parameters is "#". This is also reflected in the syntax of intrinsic function such as String functions ([LEFT\\$\(\)](#), [RIGHT\\$\(\)](#),...) and [TeLiTab](#) functions ([UNION#\(\)](#), [QUERY#\(\)](#),...). See also [QuaestorSyntax](#).

Member of functional group: [Presentation](#)

Member of knowledge base type: [Classic and general type](#), [Scenario type](#) and [Taxonomy type](#)

Quick links: [Function overview](#) | [Attribute overview](#) | [Constants overview](#) | [Dimensions overview](#)