



Objectifs:

Par un ensemble d'exemples démontrer que le langage SQL est beaucoup plus simple et clair que le langage des tableurs.

Trouver au sein de ce document toutes les formulations logiques qui à l'issue de leur lecture donnent l'assurance au lecteur d'une maîtrise très rapide, et même immédiate du langage SQL (Structured Query Language).

Expertizers Copyright 2006-2009

LES FORMULES SQL DANS LE LOGICIEL

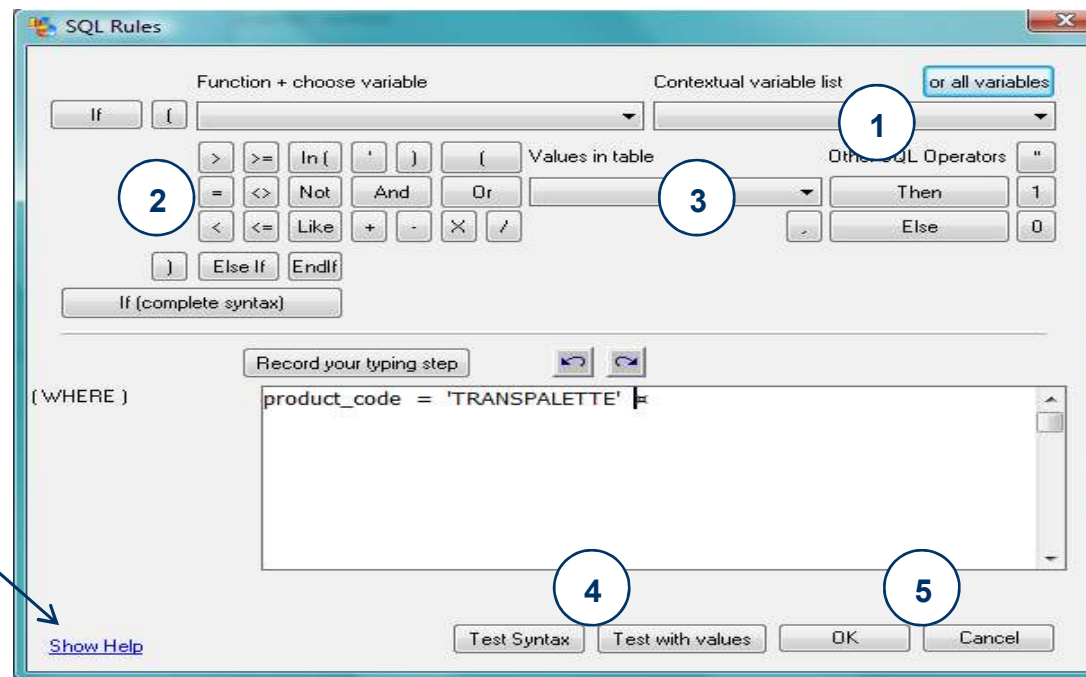
Le logiciel EXPERTIZERS repose exclusivement sur l'exécution de formules insérées dans un modèle et traitées séquentiellement dans la structure hiérarchique exposée dans le document préalable **'logique et fonctions'**.

Les formules (ou règles) sont de deux natures:

Les formules de condition (where ...)

Les formules de calcul (variable = ...)

Pour vous aider à rédiger une formule, double-cliquer sur sa zone de saisie. Vous afficherez alors le didacticiel qui dans la plupart des cas vous évite de saisir le moindre caractère.



pour retrouver les exemples de formules



EXEMPLES DE FORMULES DE CONDITION

WHERE...

Condition OK

Condition KO

country = 'USA'	USA	Autres valeurs
country in ('USA','GB','FR')	USA, GB, FR	Autres valeur
country like 'U%A'	USA, UBA	ASA, USB
city like 'P%%S' or city like 'N%%K'	PARIS, NEW-YORK	XXXPARIS
city like '%P%%S'	XXXPARIS	PARISXXX
city like '%P%%S%'	XXXPARISXXX	
UCASE(city) = 'PARIS'	PARIS,paris	
Country <> " (taper 2 simples quotes sans blanc intermédiaire)	un ou plusieurs caractères autres que des blancs	absence, même avec des blancs
diameter = 10 or diameter = 10.50	10 et 10.50	autres valeurs numériques
diameter in (10,10.50,30)	10 et 10.50 et 30	autres valeurs numériques
delivery_date < '2005/12/31'	date inférieure	date égale ou supérieure
(country = 'USA' and delivery_date < '2005/12/31') or (country = 'GB')	(USA et date inférieure) ou GB	FR ou (USA et date égale ou supérieure)
(country = 'USA' and (diameter in (10,10.50) or diameter > 100)) or (country = 'GB')	(USA et diamètre =10 ou10.50 ou > 100) ou GB	déduisez vous-même !
(if country ='FR' then 1 else 0.80 endif) * diameter > 150	FR et diamètre * 1 > 150, GB et diamètre * 0.80 > 150	déduisez vous-même !
machine = " or machine = 'LINE_1'	Machine LINE_1 ou absente	Machine présente différente de LINE_1



EXEMPLES DE FORMULES DE CALCUL (selon leur niveau)

Variable =

VARIABLES DEFINIES AU NIVEAU 'USER INTERNAL VARIABLES'

transport_price = 10.45

transport_price= if today() > '2008/11/30 then 10.70 else 10.45 endif

transport_price=if pays = 'FR' then 10.45 else if region = 'UE' and pays <> 'GB' then 10.55 else 11 endif endif

VARIABLES CALCULEES AU NIVEAU DES OBJETS D'OPERATION

cadence = 100

cadence = cadence * 0.7

cadence = if diameter = 20 then cadence * 0.7 else if diameter = 30 then cadence * 0.8 else cadence * 0.9 endif endif

quantity = if (diameter in (10,20,30,50) or diameter > 70) then (quantity * 1.1) else quantity endif

VARIABLES CALCULEES AU NIVEAU ACTIVITE

activity_value = if cadence_ope_PRODUCTION = 0 then 0 else (quantity_ope_PRODUCTION / cadence_ope_PRODUCTION) endif

activity_value = if cadence_ope_PRODUCTION = 0 then 1 else 0 endif

activity_cost = quantity_init * transport_price

activity_text = if pays = 'USA' then 'transport par dhl' else 'transport par voie normale'

Observations

Valeur calculée fonction de la date du jour du PC

Autant de endif que de if. En cas de valeurs multiples, fonction d'une même variable, utilisez la fonction decode (choix au menu 'utilitaires')

Initialise la cadence nominale d'un objet (machine par exemple)

Redéfinit la valeur de la cadence en utilisant la dernière valeur mémorisée

Ne pas hésitez à mettre des parenthèses pour clarifier la formule

Expression qui évite une division par zéro génératrice d'un arrêt malencontreux en phase de calcul. L'expression calcule ici le temps fonction de la quantité à produire au stade PRODUCTION.

Cette formule appliquée sur une activité de type ERROR_xxx générera une anomalie pour toute valeur calculée différente de 0.inale

Le coût calculé utilise la quantité de la ligne du fichier d'entrée et la multiplie par la valeur de transport_price calculée au niveau 'User Internal Variables'

Alimentation de la zone de texte prévue au niveau de l'activité

Attention: Ne jamais utiliser des constantes seules entre parenthèses, par exemple `quantity * (2/3)`, car le SQL utilisé calcule 2/3 sans décimale (soit 0) ! bug ? A la place utiliser `(quantity * 2) / 3` et pas de problème de précision !



LES FONCTIONS DE MANIPULATION D'UNE VARIABLE

Les fonctions de SQL sont nombreuses. Elles permettent d'analyser le contenu d'une variable pour en tirer un résultat. Nous nous contentons ici de vous indiquer la façon d'accéder à la liste explicative de chacune d'elles. La fonction help du didacticiel fournit tous les exemples utiles. Rassurez-vous, leur recours est plutôt rare !

NB: La fonction ne s'inscrit dans la zone de formule qu'après sélection de la variable

3 UCASE(pays)

Retour au didacticiel

SQL Functions	DEFINITION DE LA FONCTION	EXEMPLES
VARCHAR		
LCASE(variable)	convertit tous les caractères en minuscules. Peut être utilisé quand vous n'êtes pas sûr de la formalisation du contenu.	activity_text = activity_value then 1 else 0 e
LENGTH(variable)	retourne le nombre de caractères alphanumériques contenus dans la variable. Permet de tester si la variable contient une donnée ou non.	VIOLET AND ERROR_1 = IF 1 else 0 endif
LOCATE(variable,élément recherché, position de départ)	Recherche d'un caractère ou d'une chaîne de caractères au sein de la variable à partir d'une position donnée (facultative 1 par défaut). La fonction	LOCATE(pays: LOCATE (pay

Se rappeler: une fonction ne change pas le contenu de la variable utilisée: Fonction(variable). La fonction créé un résultat temporaire nommé par la fonction elle-même, utilisable directement dans vos règles (de condition ou de calcul). Exemple: IF LENGTH(event_code) < 15 Then / activity_value = LENGTH(event_code)



EN RESUME

Un langage beaucoup plus structuré que les macros d'un tableur,
donc plus clair.

Un langage commun avec les DSI,

donc partageable pour exprimer puis faire comprendre vos besoins vis-à-vis du système d'information, notamment si celui-ci a besoin d'être complété d'informations que vos modèles font apparaître comme indispensables à la qualité des analyses (prix de revient et rentabilités).

Un ensemble très complet d'exemples,

donc utilisable immédiatement pour exploiter tout le potentiel d'expertise du logiciel.



Avertissement

Ce document vous est présenté par DB.SOFTART.SERVICES. (« DB »).

Ce document et son contenu sont le produit de recherches et de développements importants conduits par DB. Ils vous sont présentés dans le seul but de vous permettre d'évaluer une partie des capacités du logiciel Expertizers.

Vous devez donc traiter le contenu de cette présentation, ainsi que toute autre information ou document en relation avec les solutions présentées au cours de cette conférence (les « Informations ») comme étant confidentiels et la propriété exclusive de DB.

En recevant ainsi les Informations, vous vous engagez à n'utiliser les informations que dans le seul but d'évaluer pour votre propre compte les capacités du logiciel Expertizers, à maintenir leur confidentialité et à ne transmettre à quiconque, totalité ou partie des Informations, sans autorisation écrite et préalable de DB.

- DB.SOFTART.SERVICES
Didier RICHE
Conseiller et chargé d'affaires
+33 (0)6.63.92.20.60

