

Vector 6 Database Variables

Database variables are used to populate text fields (and other objects such as barcodes) with information stored in a database. It is normal to have several tables within a database to contain all pre-defined information such as product data, country of origin, variety or size calibres. Almost all printable data that has been pre-defined can be stored in a database and multiple tables can be linked to a single label design allowing many data combinations to flow through data sources.

Why use a database variable?

Imagine you are printing a label for a strawberry product where you need to print the name of variety. There are three ways this can be achieved:

- Use fixed text You could insert a text field and simply type the name of the variety into this field. Every time the variety changes the label design will need to be edited. A process which is time consuming and is prone to spelling mistakes and poor label layout.
- 2. Use a Prompted Variable You could insert a prompted variable into the text field which will activate at print time, prompting the user answer a question/ prompt such as 'Enter Variety' and type in the name of the variety. The label design does not need to be edited, however the user still has to type in the variety every time the label is printed (every print job). This option is slightly more effective than option 1 as the label design requires no editing. However it is just as time consuming and prone to spelling mistakes.
- 3. Use a Database Variable This option links the text field to a pre-defined list of varieties that has already been populated. At print time the user selects the variety from a drop down list. Spelling mistakes are avoided and the process of a simple click of a mouse to select a variety is much quicker



SELECT VARIETY	Carrent Selection
ANNA BRIGITTA ELSANTA EVES DELIGHT	
SONATA SONATA SWEET EVE VIVA	
ОК	Cancel

DOCUMENT DETAILS Document version 2.0 Updated 1 December 2016

PRODUCT DETAILS

TECHNICAL SUPPORT www.progressive-id.co.uk support@progressive-id.co.uk Tel: 01732 850511



How to create a Database Variable

For the purpose of this guide we will make the assumption that the database with product tables already exists.

1. Create a label design and insert a text field.

2. Open the Edit Variables dialog box by clicking the Variables button from the toolbar then edit variables from the drop down list.

3	¶ A Picture Block → Block → Block → Block → Block → Block → Arrange	Varia	bles edit variables	9 Variable Sample Names Data View
		v	all 🕨	-
Ì	5 10 15 20	4	Prompted >	150 155 160 165 170 175 180 185 190 95
Î		1	Sequential Number 🔶	
		69	Database 🕨	
		۲.	Time/Date >	
		0	Macro 🕨	
		ID	Field >	
		1	Quantity •	
		8	Colour >	
			Block •	

3. On the left hand side select Database from the list of variable types and press the New button.

Edit Variable	×
View by Alphabetical Order	Variable Language Input Mask Prefix/Suffix Filter variable Dependent Variables Formatting
Wew by Type Order	Name
	Prompt Database Name DATAGUARD Tablese Name TESCO PRODUCT Search Field CATEGORY Description Field
	Vatiable Length Pecal Last Entry Use Fixed Key Use data transformations
\downarrow	Change DB
New Delete Export Import	Close Apply



4. Vector will automatically use the most recently used database as a source. The Database Name that appears on the right hand side is actually the name of the ODBC driver that connects to the database. If you need to use a different database click the Change DB button and select the relevant ODBC from the list of Machine Datasources. If the datasource is not present you will need to create an ODBC driver in Windows before creating any database variables.

Edit Variable								×
Edit Variable Vew by Aphabetical Order Wew by Type Order Prompted Prompted Mono Source and Anoro D Field Colour Colou	Variable Name Prompt Databa Table N Search Data Fit	Language se Name lame Field eld tion Field	DA1	k Prefix/Suffix TAGUARD SCO PRODUCT TEGORY	Filter variable	Depender	nt Variables	Formatting
				anable Length Recall Last Entry Ise Fixed Key Ise data transfom	nations			

6. Select the table that contains the data you want to print onto the label design.

7. Next select the search field for the table. The contents of this field is used at print time for the user to select from. Often the search field is prefixed with the word SEARCH in the table you have selected.

Edit Variable					×
Vew by Alphabetical Order Wew by Type Order	Variable Language Inp Name	out Mask Prefix/Suffix	Filter variable	Dependent Variables	Formatting
O Sequertial Number Database Time/Date Output Nacro Output Output Fond Output Format	Database Name Table Name Search Reld Data Field Description Reld	DATAGUARD TESCO VARIETY ST SEARCH VARIETY VARIETY VARIETY CASE SEN CATEGORY FLESH TYPE FLESH TYPE CASE S RETAILER Variable Length Recal Last Entry	RAWBERB	Key	

8. Next select the data field which contains the data you wish to print onto the label design. Sometimes the search field and data field will be exactly the same. This is always the case if the table being used only has one field within it.



Edit Variable					×
View by Aphabetical Order Wew by Type Order Prophed Description	Variable Language Inp Name Prompt	put Mask Prefix/Suffix	Filter variable	Dependent Variab	les Formatting
Colourity Colour Colourity Colourity	Database Name Table Name Search Reld Data Field Description Reld	DATAGUARD TESCO VARIETY STR SEARCH VARIETY VARIETY SEARCH VARIETY VARIETY	RAWBERF -	Кеу	
G Output Format		VARIETY CASE SENS CATEGORY FLESH TYPE FLESH TYPE CASE SI RETAILER Vanable Length Recal Last Entry	ENSITIVE		

9. Next give the variable a unique name. Typically this will be the same as the data field as the name should be relevant to the content of the data field.

Edit Variable	
Wew by Aphabetical Order Wew by Type Order Prompted Of Prompted Of Securital Number	Verlable Language Input Mask Prefix/Suffix Filter variable Dependent Variables Formatting Name VARIETY Prompt Database DataGuidaD DataGuidaD
- 123 Database - 37 Time/Date - 10 Field - 10 Field - 10 Countly - 20 Colour - 0 Output Format	Database Name DATABUAND Table Name TESCO VARIETY STRAWBERF • Search Reid SEARCH VARIETY • Key Data Field VARIETY • Description Reid •

10. Next enter a prompt for the variable which is also displayed at print time.

Edit Variable	
View by Aphabetical Order View by Type Order	Variable Language Input Mask Prefx/Suffix Filter variable Dependent Variables Formatting
Prompted Decomplet Osciential Number Sciential Number Sciential Number Sciential Number Osciential Number Osciential Number Colour Osciential Number Colour Osciential Number Osciential Numb	Prompt SELECT VARIETY Prompt SELECT VARIETY Database Name DATAGUARD Table Name TESCO VARIETY STRAWBERF • Search Field SEARCH VARIETY • Key Data Field VARIETY • Description Field •
-of Output Format	

11. Variable Length (optional)

Variables are represented by blue Vs, one for each character, determined by the length of the field in the database. As this is 255 by default it can be very confusing and trigger data overflow warnings. Enabling variable length will display only a single blue V and is useful for data such as variety or product descriptions. However it should not be used if the length of the data is always the same. For example an EAN8 Barcode is always 7 digits in length (the 8th digit is a check digit and is calculated automatically) so would not be suitable.



VARIABLE LENGTH DISABLED

VARIABLE LENGTH ENABLED



12. Recall Last Entry (optional)

This feature should be used with caution as it causes the system to remember the previous data selections when printing. It can be useful to apply recall last entry when a variable seldom changes – perhaps one grower always packs a certain product – as it saves having to select it for every print job. However most data will change with every print job and enforcing the user to make a conscious choice helps ensure accuracy.

13. When all settings for the database variable have been made click Apply to create the new variable. If there is only a single field in the table the new variable will now be added to the lis of variables on the left hand side.

Edit Variable	
⊘ View by Alphabetical Order	Variable Languag
Wiew by Type Order	Name
Prompted	Prompt
	Database Name
	Table Name
	Search Field
Macro	Data Data
Quantity	Data Held
Colour	Description Field
Block	
······of Output Format	

If the table has more than one field, Vector will ask whether you want to create just the one variable (click No) or more variables for each field in the table (click Yes).





To create more variables click Yes and Vector will show you all the available fields in that table. Highlight the fields you want to create variables for in the Available list and click Add-> followed by OK.

Selected Item	s						×
Available:				Selected:			
SEARCH VA VARIETY CA FLESH TYPE RETAILER	RIETY SE SENSITIVE CASE SENSITIVE	*	Add-> <-Remove	CATEGORY FLESH TYPE		*	Move Up Move Down
•	+	Ŧ	ОК	< Cancel	4	Ŧ	

Vector will then create all of these variables (including the one you were originally creating) using the field name as the variable name.





Database variables when used with DataGuard 3

DataGuard 3 has been developed to present the label data in a systemised manner. In some cases it is important to observe some specific rules in relation to the prompt field on database variables.

It is also important to have a link to a table that contains product data, even if you are not actually printing this data. The primary purpose of DataGuard 3 is to calculate Display Until, Best Before and Use By dates. It handles this by using offset ShelfLife fields in the product data tables for each product record.

At print time the user needs to make a product selection so that the system can reference either ShelfLife and/or ShelfLife2 fields to calculate the relevant dates on the label design. The product table variable being used as a link to the product data table must use the prompt SELECT PRODUCT. It is also important only variables linking to the product table use this prompt. For example if connecting to a variety table use SELECT VARIETY instead.