

# The Keystroke Report Engine

## Overview

The **Keystroke** Report Engine controls the way that **Keystroke** prints Reports and Labels. Users can customize Reports and Labels, using filters, ranges and a variety of selectable options. It is not, however, entirely free-form; the user's selection of database fields and options is limited to those that would be most appropriate for the type of Report or Label in question. Each type of Report and Label comes with one or more standard forms, which the user can either customize or use without modification.

The Report Engine includes a built-in Report Form Editor, which allows users to create and edit Report and Label forms. Forms can include user-defined variables, which in turn can be based on a variety of logical Operators, functions and data fields.

Users may add Reports to both the Local and Shared Settings list.



Selecting a Report

## Generating and Printing Reports

Under most circumstances, users can create and print suitable Reports using the standard Report setup options and forms. Select the appropriate type of Report, either in the Reports module or from a Reports menu in one of the other **Keystroke** modules. If there is only one setting for the Report, **Keystroke** will display the Report setup dialog box. If there is more than one setting available, **Keystroke** will display a pop-up menu of the available settings. Select a setting and press **ENTER** to bring up the setup dialog box.

The options available in the Report setup dialog box vary, depending on the **Keystroke** module and the type of Report. For most reports, the user may group items, and select one or more ranges, filters and levels of detail, along with a variety of checkbox options. The user may also select the Report form and the print options (screen / file / printer). Highlight Form and press **ENTER** to bring up a list of the Report forms. Select a form from the list and press **ENTER** or **F10**. (See *Report Options* and *Report Form Editor*, below.)

---

The new **Keystroke** Report Engine described herein is available in **Keystroke POS Version 5** and later, and with the *Matrix Manager* module in releases earlier than Version 5 (including *Classic*).

---

## ***Saving Report / Label Settings***

There are two types of Report / Label settings:

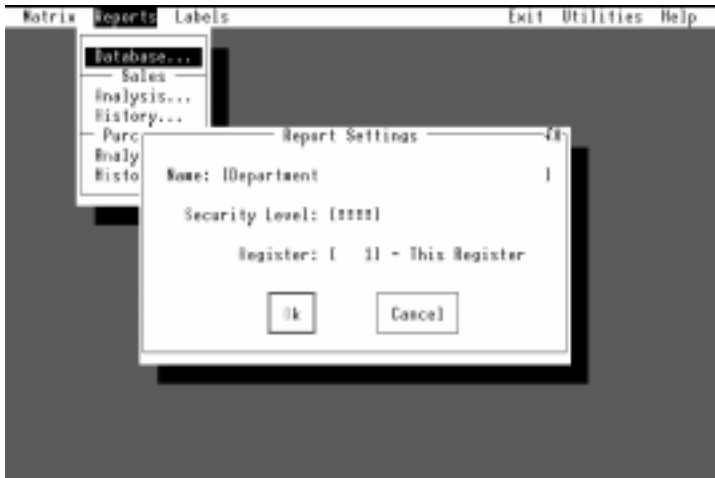
*Shared* (or Common) Report / Label settings are available to all registers. Users can temporarily change these settings for one-time use; to save the changes, however, the user must press **SHIFT+F10**.

*Register* (or Local) Report / Label Settings are available to the current register only. They will not show up on the Report / Label Settings pop-up menu for other registers. Changes to Register Report / Label Settings are saved automatically.

To create a new Report / Label Setting, select Add New Settings from the Report / Label Settings pop-up menu and enter a name for the new Report / Label in the Report / Label Settings dialog box. Enter a Security Level for the Report / Label setting, or accept the default Security Level, then enter a Register number. The choices are:

- Register 0* Available at all registers.
- Current Register #* Available only at the current register.
- Other Register #* Available only at the designated register.

Report / Labels assigned to other registers will disappear from the current register after they are saved. Make any changes to such Report / Labels at the designated register.



*Report settings can be assigned to the local Register, to another Register, or to all Registers (by entering 0 in the Register field).*

*The Report Settings Dialog Box*

# Report Options

Some of the options listed below are common to most Reports and Labels. Others are available only with a few Reports or Labels. For Report / Label options that are specific to an individual module, see the documentation for that module.

*The actual options available will depend on the module and the type of report.*



*Options for Matrix Database Report*

## Group By Dialog Box

May include the following options (see *Option Descriptions*, below.):

<b>Group By</b>	<b>Sort By</b>	<b>Detail</b>	<b>Print</b>
Range	Filter	Don't Print if Empty	Print Comments

## Inventory Dialog Box

May include the following options (see *Option Descriptions*, below.):

<b>Sort By</b>	<b>Detail</b>	<b>Print</b>	<b>Range</b>	<b>Filter</b>
Include Hidden Items		Don't Print if Empty		Print Comments

## Report On Dialog Box

May include the following options (see *Option Descriptions*, below.):

<b>Report On</b>	<b>Filter</b>	<b>Don't Print if Empty</b>	<b>Print Comments</b>
------------------	---------------	-----------------------------	-----------------------

## ***Option Descriptions***

### ***Add Allocated to QOH***

Add the Allocated Quantity to the Quantity On Hand.

### ***Add Layaway to QOH***

Add the Layaway Quantity to the Quantity On Hand.

### ***Based On***

- Based On** Used when selecting Price Tags by Purchase. Select the type of transaction to supply the information for the price tags.
- Sort By** Sort the data by one of the listed fields.

### ***Detail***

The level of detail in the Report:

- Summary** Reports the number of items etc., with no breakdown.
- Detailed** Lists individual items.
- Detailed with Page Breaks** Lists individual items, with page breaks after each group.

### ***Don't Print if Empty***

Available when the Report is Grouped or Sorted By one of the database fields (Department or Vendor, for instance). When Don't Print if Empty is checked, and one of the Sort / Group categories is empty, it won't be included on the Report. For example, if The Report is Grouped by Department, and there are no items in one of the Departments, that Department won't be included.

### ***Exclude Service Items***

Do not include items that have "Service" checked in the Inventory Database. **Keystroke** does not keep track of Inventory Quantities for Service items.

### ***File Name***

When the user selects File from the Print To list, The File Name option will appear in the Report / Label setup dialog box. Select it to bring up the Print To (File Name) box:

- File Name** Enter a name for the file, and an optional DOS path:  
i.e., **REPORT1.TXT** or **C:\REPORTS\REPORT1.TXT**.
- Delimited By** To Print To a file, then import it into a spreadsheet or database application, select a delimiter from the list. Options are:  
**Quote/Comma      Tab      Comma Only**

### ***Filter***

Use a filter to select the items for which **Keystroke** will print a Report / Labels. See *Filters and Formulas*.

## **Group By**

- None** Do not group items. All items will be included, unless they are excluded by other settings.
- [Fields]** Group items in the Report by the selected field. If an item does not contain any data in that field, it will not be included in the Report. For example, if a Report is Grouped By Department, and an Inventory item's record has a blank Department field, that item will not be included in the Report.

## **Include Hidden Items**

Include items marked hidden using the Hide/Show function.

## **In Stock Only**

Include only items currently in stock.

## **Label Per Item**

The number of labels that will be printed for each Inventory item. The options are:

- 1** One Label
- Qty in Stock** The current Quantity in Stock (QOH plus Allocated) for the item.

## **Leave QOH Blank**

Leave the Quantity On Hand blank. A Report with blank Quantities is useful when taking a physical Inventory, since it prevents employees from simply entering the QOH without making an actual count.

## **Line Item Comments**

- ON** Include comments for each line item.
- OFF** Do not include line item comments.

## **Method**

The Method is the way of calculating the total value of the transactions being Reported.

- Accrual (All)** All transactions, regardless of payment status.
- Weighted Cash** Only paid portions of transactions.
- Cash** Only transactions which are fully paid.


## **On Sale Only**

Include *only* items that currently have "On Sale" checked in the Inventory Database.

## **Price Changes Only**

Prints tags *only* for items for which the price has changed since the Last Printed Price.

## **Price Date**

If the printed price is based on a Price Table, and if that Price Table includes Price Formulas with different Date Ranges, the Price Date tells the Price Table which price to use. Enter a date using any of the acceptable **Keystroke** date formats, or press  to select a date from the pop-up calendar.

## Price Table

Select the Price Table on which to base the printed price. The table uses the Price Date in determining which price to print.

## Print

**All** Print all records or items, unless excluded by other settings.  
**Individual** Select a category or item from a list, and add it to the output. The program will then return to the list, allowing the user to make another selection. To exit the list, press **ESC**.

## Print Box Tags

**Keystroke** prints one Label for each box. For boxed or bulk items with “No Price Tag” marked in Inventory.

## Print Comments

Include Comments on the Report. The location of the Print Comment box determines which Comments will be printed. If it is in a Group By or Sort By dialog box, for instance, the Comments will be for the field that the records are grouped or sorted by.

## Printed Price

Some price tag forms allow a different price to be printed on them. Select the price from the list. Current Price is what a Cash customer would pay for the item. The options are:

<b>Current Price</b>	<b>Base Price</b>	<b>Sale Price</b>	<b>List Price</b>	
<b>Level 1 Price</b>	<b>Level 2 Price</b>	<b>Level 3 Price</b>	<b>Last Cost</b>	<b>Average Cost</b>

## Printer Name

Select a printer from the list. To print barcodes (on Labels) from the Windows version of **Keystroke**, select a Windows printer (listed as type **WINDOWS** in the third column of the list).

See *Using Windows Printer Drivers* in **Printing Labels and Price Tags**.

## Print To

Print to a Screen, Printer, or File. For Printer or File, select Printer Name or File Name.

## Range

If the Report is Grouped By one of the database fields (Department or Vendor, for instance), the user can select a Range within that grouping.

**Range** Press **INS** in the Range fields to select the beginning and the end of the Range. The Range must be continuous.

*First to Last* The default Range. Includes everything not excluded by other settings.

*Set by user* When the user enters a Range, only items in the Range will be included.

For some groupings, such as Vendors, the Range will also depend on the Sort By field.

## ReOrder Only

Include only items for which the Quantity On Hand is below the Minimum Quantity (as set in the Inventory Database).

### ***Sold Items Only***

- ON** Report only on items that were sold during the period indicated (recommended).
- OFF** Include unsold items.

### ***Sort By***

Sort the data by one of the listed fields. The Sort order affects the Range: If the data is sorted by Name, for instance, the Range will be a series of names, in alphabetical order. If the data is sorted by Code, the Range will be determined by the Codes, etc.

### ***Subtract Discount***

- ON** Distribute Invoice discount between all items on the Invoice when calculating the price charged.
- OFF** Does not distribute discount when calculating price charged (recommended).

### ***Transaction Comments***

- ON** Include comments for each transaction.
- OFF** Do not include transaction comments.

### ***Update Last Printed Price***

Saves the printed Price with the Inventory records (used with Price Changes Only field).

### ***Use Adjusted Cost***

- ON** Include adjustments resulting from the distribution of discounts and shipping (from Purchase Manager Parameters).
- OFF** Do not include adjustments.

### ***Use Last Cost***

Use the most recent Purchase Price for the Cost.

# Report Form Editor

**Keystroke** includes a variety of built-in Report and Label forms, which are designed to meet the needs of most users. Under some circumstances, however, it may be necessary to edit one of the forms, or to create a new one entirely. The Report Form Editor allows **Keystroke** users to (within certain limits) alter Report and Label form layout, add, edit or remove fields, and create custom variables.

## Opening the Report Form Editor

To use the Report Form Editor, press **CTRL F3**, or select the Form field from the appropriate Report dialog box and press **ENTER**. In the Report Form Files list, select the form to be modified and press **F3**. To create a new form, press **INS** in the Report Form Files list.

## The Edit Report Form Dialog Box

### File Name

#### Report Forms

To save a new Report Form, or an edited Form under a new name, enter four digits in the bracketed section of the File Name field. The full file name consists of a four-character code specifying the module and type of Report, followed by four user-defined digits (also known as the Form File Number), followed by the extension **KSR**.

For example, a standard Matrix Report Form could be named **XRD10060.KSR**:

**XRD1** Module / Report code (i.e., MatriX Reports-Database).  
**0060** Form File Number.

For standard **Keystroke** Report Forms, the first digit of the Form File Number is “0”. To prevent custom Forms from being overwritten by subsequent **Keystroke** installations, please enter numbers that do not begin with “0”.

In the above example, for instance, the “0” indicates that it is a standard Report Form and not a user-created Report, which would have a Form File Number that does not begin with “0” (i.e., **XRD12260.KSR**).

#### Price Tags and Labels

The procedure for saving Price Tag and Label form files is similar to the one outlined above, but the system for naming them is different. See *Price Tag and Label Form Files* in *Printing Labels and Price Tags*.

### Description

The Description will appear in the Report Form File list. It should tell the user what the form’s function is, and clearly set it apart from other forms on the list. Maximum length: 40 characters.

### Title

The Title appears at the top of the Report when it is printed. On most forms, the effective maximum length will be about 78 characters.

## Labels Box

Mainly for use with Labels. See *Printing Labels and Price Tags*.

## Table of Fields

The Table of Fields lists the fields and variables that will appear in the Report.

It is divided into sections. Most of the sections represent different levels of organization within the Report. They are numbered, with higher numbers representing higher levels of organization.

On a Matrix Report, for instance, Section 3 might represent the way that Inventory items are grouped (Department/Category/Vendor), Section 2 might represent individual Inventory items, while Section 1 could be for the Matrix items attached to an Inventory item. Many Reports also include Section 0, for totals, and a special (non-numbered) section for user-defined Variables.

Ln#	Field	Track	Width	Align	Dec	Flags	PCodes
1	[INV:Number]	2	5	L			
1	[INV:Code]	3	20	L			
1	[INV:Description]	1	30	L			
1	[INV:Selected QOH]	1	5	R		-2	
1	[INV:Selected Price]	1	10	r		-39	
2	[INV:Comment]	10	40	L			1

The form information section and the Table of Fields are used for both Reports and Labels, The Labels section is used primarily for labels.

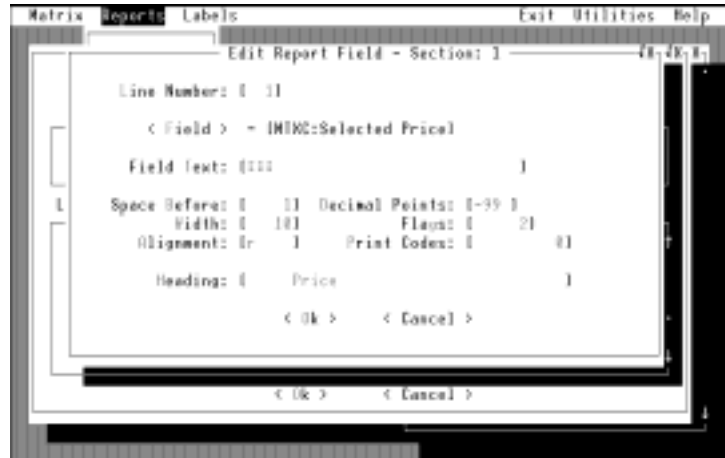
*The Edit Report Form Dialog Box*

The user cannot change the sections in a Report form. Users can, however, define Variables, and control what fields and Variables appear in each section. See *Edit Report Field Dialog Box*, below.

## Edit Report Field Dialog Box

The Edit Report Field Dialog box allows the user to edit Report and Label fields.

The user can set both the content and the format of the Report field.





The Edit Report Field Dialog Box

### Line Number

The Line Number determines the line on which the field will be printed. The lines are numbered within each section. Use Line 0 for section subtotals. If Space Before field (see below) contains a negative number, the field will print on more than one line.

### Field


Select Field and press  to see a list of the data sets available for use in the Report. Select a data set, press , then select an element from the data set. It will be displayed in this format: **[DATA SET:field]**. For example, the Inventory List Price field would be listed as **[INV>ListPrice]**. Select **None** to print the Field Text (see below) instead of the contents of a Database field or Report Variable.

**Definition: Data Set.** A data set is a collection of fields, variables and other types of data, grouped together either by function or by where they are used or generated within the program. The Report Engine can use data from several data sets on a single Report. For example, basic Inventory fields might constitute one data set, while calculated Inventory fields might form another set, and variables that are specific to the current Report would go into another data set. For a given Report, however, only certain data sets will be available, depending on the nature of the Report, and the module in which it originates.

## Data Set Names

<u>Code</u>	<u>Name</u>	<u>Notes</u>
SYS	System Fields	always List #0
RVAR	Report Variables	always List #1
INV	Inventory	
INVC	Inventory	Calculated
CST	Customer	
VEN	Vendor	
CLK	Clerk	
DPT	Department	
CAT	Category	
MTX	Matrix Item	
MTXC	Matrix Item	Calculated
ISA	Inventory Sales Analysis	
MSA	Matrix Sales Analysis	
STH	Sales Transaction	Header
STL	Sales Transaction	Line Item

## Field Text

To use a field as a label, or for other non-variable text, set the Field to **None** (see above) and enter the appropriate Field Text. **Keystroke** will print the Field Text if the Field is set to **None**, or if the field data is blank (or 0 in the case of numeric fields). The Field Text can also contain control codes (such as printer escape codes) or other special characters. Enter the ASCII decimal number of that character enclosed in braces. For example, to print the  character (ASCII 27), enter {27}.

## Space Before (or Track)

The number of blank spaces between the end of the last field (determined by that field's Width setting, and not by the length of the text in the field) and the beginning of the current field. If the Space Before is a negative number, The current field will be printed on the following line, moved back by the number of spaces indicated.

## Width

The space available (measured in characters) for printing the field. If the text in the field is longer than the assigned Width, the extra characters will be cut off. The default Width is 10 characters.

If the Width is set to 0, or is left blank, **Keystroke** will print all of the text in the field.

When a field is set to Width 0, it affects the location of subsequent fields on the same line, as do the Space Before and Width settings of those fields.

In the following chart, Field 1 has a Width of 0, and Field 2 follows it on the same line:

<u>Field 2: Space Before</u>	<u>Width</u>	<u>Where Printed</u>
Less than length of Field 1	= 0	Immediately after preceding field.
Less than length of Field 1	>0	On the next line.
Greater than / equal to Field 1	= 0	Immediately after preceding field.
Greater than / equal to Field 1	> 0	Location indicated, counting from left of page.

## Alignment

The Alignment determines how the text will be printed within the space assigned to the field (as determined by the Space Before and Width settings). Possible Alignment settings are:

<b>L</b>	Left
<b>C</b>	Centered
<b>R</b>	Right
<b>r</b>	Right with trailing spaces. For use with 3 digit price fields.
<b>B</b>	Barcoded Should be followed by the barcode format and size codes. See <i>Barcodes</i> , in the <b>Printing Labels and Price Tags</b> section, below.
<i>Any number</i>	Print only on the Label indicated. See <i>Print Only On Label #</i> , in <b>Printing Labels and Price Tags</b> section.

## Decimal Points

Entering a number in Decimal Points sets the number of decimal points to print, and identifies the field as a number (rather than text) when exporting. If the field contains 0, it will be replaced by the contents of the Text field (see above) when printing.

A negative number sets the maximum number of decimals, without trailing zeroes. Enter -99 to use the same number of decimals as the Price/Cost field: 3 digits if Enable 3 digit Cost/Pricing is ON, and 2 digits if it is OFF.

## Flags

Flags are special instructions regarding the content or formatting of the field. To select a flag, highlight Flags in the Edit Report Field dialog box and press **INS** to bring up the list of flags. To turn a flag ON or OFF, double-click on it or select it and press **ENTER**. A flag may have multiple fields turned ON at the same time.


<b>Comment Field</b>	Should always be checked if the field is a Comment field.
<b>Detail Only</b>	Do not print the field if section Detail is set to Summary.
<b>List Fields</b>	Serial #s, AltCodes, etc. Prints each item on a separate line.
<b>Skip if Blank</b>	Do not print if field is blank or 0 (for numeric fields).
<b>Line Feed Before Field</b>	Go to beginning of next line first. Mainly for subtotals.
<b>Line Feed After Field</b>	Go to beginning of next line after. Mainly for subtotals.
<b>Add CheckSum digit</b>	Add CheckSum digit at end of field. Use with Stock Numbers on Labels, when Stock Number CheckSum (in Transaction Entry parameters) is ON. See <i>Stock Number CheckSum</i> , in the <b>Printing Labels and Price Tags</b> section.
<b>Cost Coding</b>	Print the cost in encrypted format. See <i>Cost Coding</i> , in the <b>Printing Labels and Price Tags</b> section.
<b>Print Backwards</b>	Print the contents of the field backwards. Combine this flag with Cost Coding (or use it by itself) to prevent customers from reading cost (or price) data. For use on Labels and Price Tags.
<b>Add Comma Separators</b>	Use commas in long numbers: 1,000.00, 1,000,000, etc.

## Print Codes

Select print styles for the current data field or text. Not all printers are capable of printing all of the styles listed; refer to the printer's manual and/or Configuration Printer Drivers to determine which styles will work. Multiple Print Codes are allowed; some trial and error, however, may be necessary to find combinations that work properly.

For a Printer Code to apply to a field, both the ON and OFF settings must be selected. The ON setting turns the Code on before the field is printed, and the OFF setting turns it off afterwards.

To set a field to print in Bold, for instance, Bold ON and Bold OFF should both be turned on. Then, when **Keystroke** prints the field, it will turn Bold ON, print the field, and turn Bold OFF.

To turn a Print Code ON or OFF, for the current field: select it and press . If a check mark appears next to a Printer Code description, it is currently turned ON.

## Headings

Enter a heading for the field. It will appear at the top of the main body of the Report. Each section's headings appear in a row above the headings for the sections that follow. Headings may be cut off or overwritten by subsequent headings in the same section, depending on the Width setting of the fields with which they are associated.

## Variables

The **Keystroke** Report Engine gives the user access to all of the of database fields and calculated fields that most Reports would require. If the available fields do not provide all of the data needed for a Reports, the user can, under most circumstances, create a variable that will provide the necessary data.

A typical variable might tell the program to print the word "Reorder" next to any item for which the QOH is no more than five units above the Minimum Quantity, and which has an Average Cost per order unit of less than ten dollars.

The formula for the variable would look like this:

```
@IF ( [ INV:QOH ] <= [ INV:MinQty ] + 5 && [ INV:COST ] < 10 , Reorder , )
```

Which means: "If the Inventory Quantity on Hand is less than or equal to the Inventory Minimum Quantity plus five, and if the Inventory Cost is less than ten dollars, then print 'Reorder.' Otherwise, don't print anything."

It includes a function (If-Then-Else), which in turn includes filter conditions ("if the Inventory Cost is less than ten dollars"), calculations ("Inventory Minimum Quantity plus five") and logical operations (&&, the symbol for "and").

To create a new variable, scroll to the Variable section at the bottom of the Field list in the Edit Report Form dialog box, select Add New Variable and press .

See *Filter and Formula Operators* and *Filter and Formula Functions*, below.

## Edit Report Variable Dialog Box

The Edit Report Variable dialog box allows the user to create or edit variables for use in a Report. The user can give the variable a name, and create the formula that defines it.

**Name** Enter a name for the variable. It is the name that will appear in the Report Variables field list when the user selects a field or sets up a Formula.

**Formula** The formula defines the variable. It is made up of calculations, logical operations, functions, database fields, other variables, numbers and text strings. When used in the Report Editor, formulas affect the printed output in Reports and on Labels, but they do not change the contents of any databases. In other parts of **Keystroke**, such as the Database Manager, users can change database information by means of formulas.



Press **INS** to enter a field in a Report variable.

*The Edit Report Variable Dialog Box*

To add a field or variable to a formula, press **INS**, select a data set from the list, press **ENTER** and select the it from the list. To enter an Operator, function, number or text string, simply type it in the appropriate location. A formula can use multiple fields, Operators, functions, etc. in various combinations. See *Filter and Formula Operators* and *Filter and Formula Functions*, below, to find out what specific Operators and Functions can do, and how they can be combined.

## Filters and Formulas

An Operator tells **Keystroke** what to do with the data in a filter or formula. There are three basic types of Operator:

*Arithmetic Operators* add, subtract, multiply, divide and raise numbers to a power.

*Logical Operators* compare values (numbers, text, checkbox selections, other database fields) and return a result based on the comparison (greater than, less than, equals, etc.).

*General Operators* allow the user to group operations, format data and use complex functions.

**Operator:** +  
**Type:** Arithmetic  
**Context:** Any  
**Operation:** Add  
**Example:** **[INV:Price]+.50**  
**Action:** Adds \$.50 to the Price of each Inventory item.

**Operator:** -  
**Type:** Arithmetic  
**Context:** Any  
**Operation:** Subtract  
**Example:** **[ INV:Price] - . 50**  
**Action:** Subtracts \$.50 from the Price of each Inventory item.

**Operator:** /  
**Type:** Arithmetic  
**Context:** Any  
**Operation:** Divide  
**Example:** **[ INV:ListPrice]/2**  
**Action:** Divides the List Price by 2.

**Operator:** \*  
**Type:** Arithmetic  
**Context:** Numbers on both sides  
**Operation:** Multiply  
**Example:** **[ INV:SalePrice]\*[ INV:QtyOrdered]**  
**Action:** Multiplies Sale Price times Quantity Ordered.

**Operator:** \*  
**Type:** Arithmetic  
**Context:** Numbers and spaces on both sides  
**Operation:** Multiply  
**Example:** **[ INV:ListPrice] \* [ INV:QOH]**  
**Action:** Multiplies List Price by Quantity on Hand.

**Operator:** ^  
**Type:** Arithmetic  
**Context:** Any  
**Operation:** To the power of  
**Example:** **[ INV:V1Cost ] ^2**  
**Action:** Vendor 1 Cost squared (to the power of 2).

**Operator:** \

**Type:** General

**Context:** Any

**Operation:** (Literal) Treat character that follows as a character, and not a symbol or Operator

**Example:** **[INV:SalePrice] \\* [INV:QtyOrdered]**

**Action:** The Sale price, followed by “\*”, followed by the Quantity Ordered.  
i.e., “2.99 \* 100”, as opposed to “299”

**Operator:** [ ]

**Type:** General

**Context:** Any

**Operation:** **[ListName:FieldName]** Automatically entered when selecting field. Do not type brackets, ListName or FieldName.

**Example:** Press **[INS]** in the Formula field and select Matrix Item. Press **[ENTER]** and select Table Code.

**Action:** **[MTX:Table Code]** appears in the Formula field.

**Operator:** ( )

**Type:** General

**Context:** Any

**Operation:** Group

**Example:** **( [MTX:QOH]+[MTX:QLayaway] ) \* 2**

**Action:** Add Matrix Quantity on Hand to Matrix Layaway Quantity, then multiply the result by 2

**Operator:** @

**Type:** General

**Context:** Any

**Operation:** Precedes a function name

**Example:** **@DOW(100200)**

**Action:** Runs the Day of the Week function for 10/02/00.  
See *Filter and Formula Functions*.

**Operator:** &

**Type:** General

**Context:** Any

**Operation:** (Concatenate) Combine strings of text (or numbers) together, end-to-end.

**Example:** **[INV:Category]& [INV:Description]& [MTX:Element Codes]**

**Action:** Combines the Category field, followed by a space, the Inventory Description, another space, and the Matrix Element Codes. The & is optional. **[INV:Category] [INV:Description] [MTX:Element Codes]** will produce the same result.

**Operator:** ~

**Type:** General

**Context:** Operand (field, number or text) on either side

**Operation:** If first operand is blank, use the second operand.

**Example:** **[MTX:Description]~[INV:Description]**

**Action:** Prints the Matrix item Description, if there is one.  
If there is not, prints the main Inventory item Description.

**Operator:** ~  
**Type:** Logical  
**Context:** No operand in front  
**Operation:** (Is Blank) True if character after it is blank, False if it is not  
**Example:** ~[**INV:Description**]  
**Action:** Returns True (-1) if Inventory Description is blank, False (0) if it is not.

**Operator:** \*  
**Type:** Logical  
**Context:** Space before, but not after  
**Operation:** Contains  
**Example:** [**INV:Description**] \***Paper**  
**Action:** Returns True if the Inventory Description contains the word “Paper”, False if it does not.  
Works with numeric and non-numeric data: [**INV:ListPrice**] \***95** returns True if the Inventory List Price contains “95”.

**Operator:** \*  
**Type:** Logical  
**Context:** Non-numeric character on one side  
**Operation:** Contains  
**Example:** [**MTX:Description**]\***4**  
**Action:** Returns True if the Matrix item Description contains “4”, since the Description field is non-numeric (even if a particular Description consists entirely of numbers).  
Compare this to [**INV:ListPrice**]\***95** (without a space), which multiplies the Inventory list Price by 95.

**Operator:** \*  
**Type:** Logical  
**Context:** Space after  
**Operation:** Begins with  
**Example:** [**INV:Description**] **Steno**\*  
**Action:** Returns True if the Inventory item Description begins with “Steno”.  
Works for numeric fields and data.

**Operator:** !  
**Type:** Logical  
**Context:** Any  
**Operation:** Does not contain  
**Example:** [**INV:Description**]**!Paper**  
**Action:** Returns True if the Inventory Description does not include the word “Paper”, False if it does.

**Operator:** >  
**Type:** Logical  
**Context:** Any  
**Operation:** Is greater than  
**Example:** [**INV:QOH**]**>100**  
**Action:** Returns True if Inventory Quantity on Hand is greater than 100.  
If QOH = 100, returns False.

**Operator:** <  
**Type:** Logical  
**Context:** Any  
**Operation:** Is less than  
**Example:** [ **INV:QOH** ] < 100  
**Action:** Returns True if Inventory Quantity on Hand is less than 100.  
If QOH = 100, returns False.

**Operator:** >=  
**Type:** Logical  
**Context:** Any  
**Operation:** Is greater than or equal to  
**Example:** [ **INV:QOH** ] >= 100  
**Action:** Returns True if Inventory Quantity on Hand is greater than or equal to 100.

**Operator:** <=  
**Type:** Logical  
**Context:** Any  
**Operation:** Is less than or equal to  
**Example:** [ **INV:QOH** ] <= 100  
**Action:** Returns True if Inventory Quantity on Hand is less than or equal to 100.

**Operator:** =  
**Type:** Logical  
**Context:** Any  
**Operation:** Is equal to  
**Example:** [ **INV:QOH** ] = 100  
**Action:** Returns True if Inventory Quantity on Hand equals 100.  
If QOH is greater or less than 100, returns False.

**Operator:** <>  
**Type:** Logical  
**Context:** Any  
**Operation:** Is not equal to  
**Example:** [ **INV:QOH** ] <> 100  
**Action:** Returns True if Inventory Quantity on Hand does not equal 100.  
If QOH = 100, returns False.

**Operator:** !=  
**Type:** Logical  
**Context:** Any  
**Operation:** Is not equal to  
**Example:** [ **INV:QOH** ] != 100  
**Action:** Returns True if Inventory Quantity on Hand does not equal 100.  
If QOH = 100, returns False.

**Operator:** &&  
**Type:** Logical  
**Context:** Any  
**Operation:** And  
**Example:** [ **INV:QOH** ] > 100 && [ **INV:Price** ] < 5  
**Action:** Returns True if Inventory Quantity on Hand is greater than 100 and Price is less than \$5.00.

**Operator:** AND  
**Type:** Logical  
**Context:** Space before and after  
**Operation:** And  
**Example:** [INV:QOH]>100 AND [INV:Price]<5  
**Action:** Returns True if Inventory Quantity on Hand is greater than 100 and Price is less than \$5.00.  
Converts AND to &&.

**Operator:** ||  
**Type:** Logical  
**Context:** Any  
**Operation:** Or  
**Example:** [INV:QOH]>100 || [INV:Price]<5  
**Action:** Returns True if either Inventory Quantity on Hand is greater than 100 or Price is less than \$5.00.

**Operator:** OR  
**Type:** Logical  
**Context:** space before and after  
**Operation:** Or  
**Example:** [INV:QOH]>100 OR [INV:Price]<5  
**Action:** Returns True if either Inventory Quantity on Hand is greater than 100 or Price is less than \$5.00.  
Converts OR to | |.

## Filter and Formula Functions

A function performs a more complex type of operation than the simple Operators listed above. Functions can be used like Operators in filters and variable definitions. A function, however, must begin with the special function Operator @, and its parameters (field names, Operators, numbers and text strings) must be contained within a set of parentheses.

### If-Then-Else

<b>Syntax</b>	<b>@IF ( Expression , TruePart , FalsePart )</b>
<b>Parameters</b>	
<i>Expression</i>	Any logical expression (see <i>Filter and Formula Operators</i> ).
<i>TruePart</i>	Any Operator, function, field or literal text.
<i>FalsePart</i>	Any Operator, function, field or literal text.
<b>Action</b>	If <i>Expression</i> = 0 (or blank), returns the <i>FalsePart</i> . Otherwise, returns the <i>TruePart</i> .
<b>Example</b>	<b>@IF ( [ INV:Price ] &gt; 0 , [ INV:Price ] , Needs Price )</b>
<b>Result</b>	If the Price is greater than 0, prints the price. If the Price is 0 (or less), prints the message, "Needs Price".

### Left Characters

<b>Syntax</b>	<b>@LEFT ( Text , NumCharacters )</b>
<b>Parameters</b>	
<i>Text</i>	Any text or numeric field.
<i>NumCharacters</i>	Any integer.
<b>Action</b>	Returns a string consisting of the leftmost characters in <i>Text</i> , with the length determined by <i>NumCharacters</i> .
<b>Example</b>	<b>@LEFT ( [ INV:Description ] , 5 )</b>
<b>Result</b>	Prints the first five characters of the Description.

### Middle Characters

<b>Syntax</b>	<b>@MID ( Text , NumCharacters1 , NumCharacters2 )</b>
<b>Parameters</b>	
<i>Text</i>	Any text or numeric field.
<i>NumCharacters1</i>	Any integer.
<i>NumCharacters2</i>	Any integer.
<b>Action</b>	Returns a string consisting of characters in <i>Text</i> , starting at <i>NumCharacters1</i> , with the length determined by <i>NumCharacters2</i> .
<b>Example</b>	<b>@MID ( [ INV:Description ] , 5 , 10 )</b>
<b>Result</b>	Prints ten characters of the Description, starting with the fifth character.

### Middle Characters (to end)

<b>Syntax</b>	<b>@MID ( Text , NumCharacters )</b>
<b>Parameters</b>	
<i>Text</i>	Any text or numeric field.
<i>NumCharacters</i>	Any integer.
<b>Action</b>	Returns a string consisting of the rightmost characters in <i>Text</i> , starting at <i>NumCharacters</i> , counting from the left.
<b>Example</b>	<b>@MID ( [ INV:Description ] , 5 )</b>
<b>Result</b>	Prints the Description, starting with the fifth character.

## **Word Wrap Middle Characters**

**Syntax** @WRAP ( *Text* , *NumCharacters1* , *NumCharacters2* )

**Parameters**

*Text* Any text or numeric field.

*NumCharacters1* Any integer.

*NumCharacters2* Any integer.

**Action** Returns a string consisting of characters in *Text*, starting just after the first space character to the left of *NumCharacters1*, with the length determined by *NumCharacters2*.

**Example** @WRAP ( [ INV:Description ] , 5 , 10 )

**Result** Prints ten characters of the Description, starting just after the first blank space before the fifth character.

## **Word Wrap Middle Characters (to end)**

**Syntax** @WRAP ( *Text* , *NumCharacters* )

**Parameters**

*Text* Any text or numeric field.

*NumCharacters* Any integer.

**Action** Returns a string consisting of the rightmost characters in *Text*, starting just after the first space character to the left of *NumCharacters*.

**Example** @WRAP ( [ INV:Description ] , 5 )

**Result** Prints the Description, starting just after the first blank space before the fifth character.

## **Right Characters**

**Syntax** @RIGHT ( *Text* , *NumCharacters* )

**Parameters**

*Text* Any text or numeric field.

*NumCharacters* Any integer.

**Action** Returns a string consisting of the rightmost characters in *Text*, starting at *NumCharacters*, counting from the right.

**Example** @RIGHT ( [ INV:Description ] , 5 )

**Result** Prints the last five characters of the Description.

## **Day Of Week**

**Syntax** @DOW ( *DateString* )

**Parameters**

*DateString* Date in the format MMDDYY, with no separation

**Action** Returns the day of the week represented by the date, as a number.  
Monday = 1 through Sunday = 7

**Example** @DOW ( 100300 )

**Result** Returns 2, representing Tuesday, the day of the week for Oct. 3, 2000.

## **Day Of Month**

**Syntax** @DOM ( *DateString* )

**Parameters**

*DateString* Date in the format MMDDYY, with no separation

**Action** Returns the day of the month represented by the date, as a number.

**Example** @DOW ( 100300 )

**Result** Returns 03, which represents the 3rd day of the month.

## Printing Labels and Price Tags

The procedure for printing Labels and Price Tags is generally the same as the procedure for printing Reports. Some features, however, apply only to Labels and Price Tags. They are covered in this section.



Some printing options apply only to Labels and Price Tags.

Printing Price Tags and Labels

## Price Tag and Label Form Files

Price Tag and Label forms can be created and edited in the same manner as Report form files. They do not, however, use the same system for naming files (see *The Edit Report Form Dialog Box*). Price tag and label files can have any name (up to 8 characters). They are identified by their extensions instead:

<u>Form</u>	<u>Extension</u>
Price Tag	.TAG
Matrix Price Tag	.XTG
Customer Label	.CLB
Vendor Label	.VLB

## The Labels Section (in the Edit Report Form Dialog Box)

These fields are mainly for use with labels, although most of them can be used on reports if necessary.

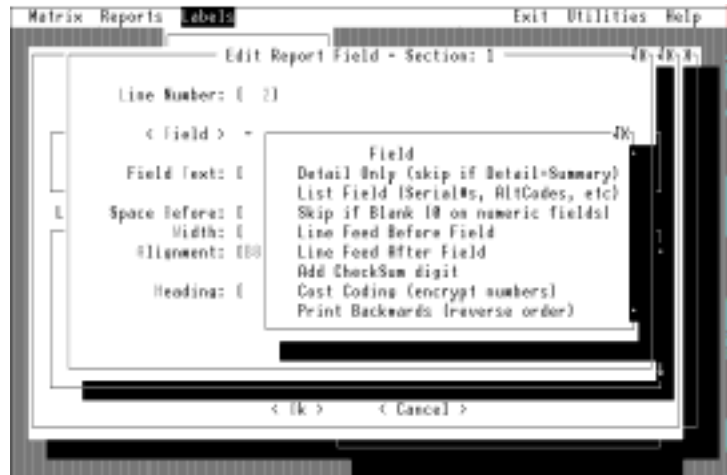
<b>Num Lines</b>	The number of lines that each section should fill. If the actual text to be printed has fewer lines, the program will print blank lines to make up the difference.
<b>Number Across</b>	The number of labels to be printed across the page. For reports, leave this setting at its default value.
<b>Top Margin</b>	Number of lines in the top margin (Windows printer drivers only).
<b>Width</b>	Only used for printing barcodes. Enter the width of the label, in tenths of an inch. <b>Keystroke</b> will use it to determine the spacing of barcodes on a page of labels. For example: If a label is 1 1/2 inches wide, enter 15 (15 tenths of an inch = 1 1/2 inches).
<b>No Header</b>	Prints with no header.

## Stock Number CheckSum (in Report Field Flags)

What happens if a Clerk enters the wrong Stock Number on an Invoice? **Keystroke** can use a CheckSum digit to minimize data entry errors of this type. When the CheckSum feature is enabled, **Keystroke** uses the Stock Number to calculate a CheckSum digit, which it then places at the end of the Stock Number when printing price tags. Stock Number 122, for instance, might be printed as 1229 -- 122 followed by 9, the CheckSum digit.

When the Clerk enters Stock Number 1229, the program separates the CheckSum digit (9 in this case), calculates what the CheckSum should be from the remaining digits (the original Stock Number, 122), and compares the calculated CheckSum with the CheckSum digit as entered by the Clerk. If the two digits are the same, The program will accept the Stock Number as valid and use it (after removing the CheckSum digit) to identify the Inventory item (in this case, Stock # 122). If the two digits are not the same, or if the Clerk enters the Stock Number without the CheckSum digit, the program will display an "Invalid Number" message.

Press **INS** in the Edit Report Form Flags field to bring up the Flags list box.



Report Field Flags List Box

To enable the CheckSum feature when printing price tags: Select the Price Tag / Label form from the Report Form Files list in the Price Tag setup dialog box and press **F3** to bring up the Edit Report Form screen. Select the field that contains the Stock Number and press **INS**. In the Edit Report Field dialog box, select Flags and press **INS**. Select Add CheckSum digit from the list and press **ENTER** to turn it ON (so that it has a check mark next to it). Press **F10** or **ESC** to exit the list and save the changes. **Keystroke** will now add a CheckSum digit to that field when printing a price tag or label.

To enable the CheckSum feature for Sales or Purchase entry: In the Sales or Purchase module, select Parameters from the Transaction menu. In the Sales Manager, select Line Item from the pop-up menu. In the Purchase Manager, select Entry from the pop-up menu. In the Line item Entry Parameters (or Entry Parameters, in the Purchase Manager) dialog box, select Stock Number CheckSum and press **Space** so that it marked with an X.

Now, when the user enters a Stock Number on a Sales or Purchase transaction, **Keystroke** will interpret the last digit as a CheckSum, and process it accordingly.

## Cost Coding (in Report Field Flags)

Cost Coding is for use on labels. It prints the cost in encrypted format (which customers can not decipher), using the following formula:

0 = A   1 = O   2 = W   3 = R   4 = U  
5 = F   6 = X   7 = V   8 = G   9 = N

All other characters (decimal, dollar sign etc.) are removed.

To customize the encryption, create a text file named **COSTCODE.TXT**, consisting of at least ten characters, and place the file in the main **Keystroke** directory. The program will use the first ten characters in the file to represent the digits 0 through 9 respectively. If this file does not exist, the program will use the default code (AOWRUFXVGN).



Press **[F5]** in the Edit Report Field Alignment field to see the Alignment Code list box.

Alignment Code List Box

## Print Only on Label # (in Report Field Alignment Codes)

This option is required for some specialized printers, which do not automatically drop down to the next row of labels after printing a row. For these printers, **Keystroke** must send the appropriate commands (typically a carriage return and line-feed) to the printer at the end of the row.

To use this feature, check the printer's documentation (or contact the printer manufacturer) to find out what commands to send the printer. Add a field at the end of the label form, set the Field to None and enter the appropriate printer commands in the Field Text field. Count the rows of labels per page and enter that number as the first character in the Alignment field (i.e., for 3-across labels, enter 3). **Keystroke** will now send the correct command to the printer after printing the last label in each row.

## Null Fields

When you print labels for Matrix items, if a field contains the Null character, the program will replace it with the value in the corresponding Inventory database record. If the Base Price for a Matrix item is set to Null, for example, and the main Inventory item's Base Price is \$9.95, the program will print the Matrix item's labels with a Base Price of \$9.95.

## Barcodes (in Report Field Alignment Codes)

---

The following instructions are intended for use with laser and dot-matrix printers. Contact an Authorized **Keystroke** dealer for help in printing price tags with a thermal label printer. Inkjet printers generally do not follow a precise, uniform standard for spacing lines and characters, making it difficult to align labels properly when printing.

---

**Keystroke** can print barcode price tags using all common barcode formats. Most Barcode readers can automatically recognize different formats, making it possible to use more than one barcode format on price tags. The Code 128 and Code 39 formats are among the most versatile. They can include both alphabetic and numeric characters, and vary in length, up to 30 characters. Code 39 is best for alphanumeric fields, while Code 128 is best if the field contains only numbers.

To print a field as a barcode, select Alignment in the Edit Report Field dialog box and enter B, followed by the code representing the barcode format. After the B and the format code, enter the barcode size information.

### Barcode Formats

Format	Code	Notes
Codabar	9	
Code 128	8	Automatic Subset Switching
Code 128A	J	Upper Case / Control Characters
Code 128B	K	Upper Case / Lower Case
Code 128C	L	Double Density
Code 16K	N	Automatic Subset Switching
Code 16KA	O	Upper Case / Control Characters
Code 16KB	P	Upper Case / Lower Case
Code 16KC	Q	Double Density
Code 39	5	3 of 9
Code 49	M	
Code 93	C	
EAN/JAN-13	3	13 digits
EAN/JAN-8	4	8 digits
Extended 3 of 9	6	
Extended 93	D	
HIBC	F	HIBC compliant 3 of 9
Interleaved 2 of 5	7	
Maxicode	R	
MSI Plessey	B	
Portable Data File 417	G	
UCC-128	E	
UPC-A	1	11 digits
UPC-E	2	11 digits
UPC-E	H	6 digits, 0 type
UPC-E	I	6 digits, 1 type
Zip+4 Postnet	A	5, 9, or 11 digits

For example, B8 in the Alignment field tells **Keystroke** to print a Code 128 barcode: The B means “print a barcode”, and 8 tells the program to use the Code 128 format, with automatic subset switching.

## Barcode Size Codes

Enter the size codes after the format code. Each size code is a single character (1-9, followed by A-Z) representing the numbers 1-35.

Code	Unit	Default
Height	10ths of an inch	2
Bar Width	144ths of an inch	2
Bar Ratio	1 to 3	1

For example, B8FC2 is the code for a (large) barcode: B8 tells the program to print a Code 128 barcode with automatic subset switching. The height code is F: a height of 1.5 inches (15/10ths of an inch, since F is the format code for 15). The Bar Width Code is C: a bar width of .0833 inches (12/144ths of an inch, since C is the format code for 12). The bar ratio is 2.

---

*Why do we use 1-9, A-Z? Why don't we just use 1-35? When **Keystroke** interprets the format codes for a barcode, it expects each one to be represented by a single character. If we used numbers, we would be restricted to 1-9 (or 0-9). Adding A-Z allows us to have 35 single-character codes.*

---

## Using Windows Printer Drivers

The Windows version of **Keystroke** can use Windows printer drivers, with some limitations. A Windows driver may not accept all of the formatting commands that **Keystroke** sends to the printer, so output may lack some features (condensed type, etc.) that would be present when using a built-in **Keystroke** driver.

In general, we recommend using one of **Keystroke**'s drivers. It may be necessary, however, to use a Windows driver under the following circumstances:

- To print barcodes from the Windows version of **Keystroke**, or
- If the printer is a Windows-only printer.

To use a Windows driver, select Printers from the Configuration Manager's Setup menu. In the Printers list, select a printer and press <Enter>, or press <Ins> to create a new one. In the Printer Setup dialog box, select Driver, then select Windows from the Drivers. Exit the Drivers list and select Port in the Printer Setup dialog box. In the Port list, select the printer's actual Windows driver.

## Windows Printer Settings - PRD.TXT

When you print from Keystroke for Windows, you can either use one of Keystroke's built-in printer drivers, or select the Windows driver for your printer.

Keystroke needs to send Printer Codes, such as Bold, Condensed, Underlined, etc., to the printer. If you use a Keystroke's printer driver, the Codes are set in the driver's Printer Codes dialog box. If you use a Windows driver, the Print Codes are controlled by a file called PRD.TXT, in the main Keystroke directory (i.e., C:\KEYSTROK). You can edit it using Notepad, Wordpad, or any other text editor or word processor.

PRD.TXT is organized like a table, made up of rows and columns. The rows represent Printer Codes; the columns represent the settings that you can assign to the Printer Codes. Each column has a heading, a set position and width (in characters), which must not be changed. Any Printer Code settings that are blank will not be changed when the program uses that Code.

Heading	Description	Parameters	Start	Width
Code	Printer Code number/name.	Any characters	1	4
Font Name	The font	A Windows font name	7	32
CPI	Characters Per Inch	Integer or dec. fraction	40	9
Size	Font size <sup>1</sup>	Integer or dec. fraction	50	9
Char Set	Character set <sup>2</sup>	Integer (0 - 255)	60	4
Bold	Turn bold ON / OFF	1 = ON, 0 = OFF	65	4
Italic	Turn italic ON / OFF	1 = ON, 0 = OFF	70	4
Underline	Turn underline ON / OFF	1 = ON, 0 = OFF	75	4
Strk Thru	Turn strikethrough ON / OFF	1 = ON, 0 = OFF	80	4
Color	Color of text (Color printers only)	Integer (0 - 15)	85	4

---

*NOTE: The following settings are not affected by PRD.TXT. They are set in the Printer Codes dialog box of Keystroke's Windows printer driver: Page Width, Page Length, Single Line and Double Line*

---

### **Printer Code Names**

The first Code in the list is named Def, for Default. All of the codes that follow are in numbered pairs (i.e., 1+, 1-). The + Code should turn the settings ON. The - Code should turn them OFF. Code pairs 1 through 9 correspond to the named settings in the Printer Codes list:

- 1+ Pica ON
- 1- Pica OFF
- 2+ Condensed ON
- 2- Condensed OFF
- 3+ Emphasized ON
- 3- Emphasized OFF
- 4+ Enhanced ON
- 4- Enhanced OFF
- 5+ Bold ON
- 5- Bold OFF
- 6+ Underline ON
- 6- Underline OFF
- 7+ Italicized ON
- 7- Italicized OFF
- 8+ DoubleWidth ON
- 8- DoubleWidth OFF
- 9+ DoubleHeight ON
- 9- DoubleHeight OFF
- 10+ Code 10 ON
- 10- Code 10 OFF
- etc...

---

*NOTE: The setting names are for your convenience; you can assign any combination of settings to any of the Printer Codes.*

---

<sup>1</sup> Use either CPI or Size to control the font size, but not both. If you enter a CPI setting, the program will use it and ignore the Size setting.

<sup>2</sup> This setting should be left at 255, so line drawing and other extended characters can appear properly.

## Text Color Codes

If you have a color printer, you can use these color codes:

Code	Color
0	Black
1	Dark Blue
2	Green
3	Teal
4	Dark Red
5	Violet
6	Dark Yellow
7	Gray - 25%
8	Gray - 50%
9	Blue
10	Bright Green
11	Turquoise
12	Red
13	Pink
14	Yellow
15	White

---

*NOTE: These codes match the order of the colors in the Configuration Manager's Colors settings.*

## PRD.TXT: Sample

This is what the default PRD.TXT looks like, when Keystroke creates it. To restore the default settings, exit Keystroke, delete the current PRD.TXT, then restart the program.

Printer Codes for Windows Printers

Code	Font Name	CPI	Size	Char Set	Bold	Ital ic	Undr Line	Strk Thru	Color
Def	Courier New	10	12	255	0	0	0	0	0
1 +		10	12						
1 -		10	12						
2 +		16.666667	7.2						
2 -		10	12						
3 +									12
3 -									0
4 +							1		
4 -							0		
5 +					1				
5 -					0				
6 +							1		
6 -							0		
7 +						1			
7 -						0			
8 +									
8 -									
9 +									
9 -									
10+									
10-									
11+									
11-									
12+									
12-									
13+									
13-									
14+									
14-									
15+									
15-									