

3

Kapitel 3 Übersicht über reservierte Wörter

Es gibt eine Anzahl von reservierten Wörtern in RPG. Diese reservierten Wörter verweisen auf spezielle Namen. Alle speziellen Namen beginnen mit einem einheitlichen Symbol (z. B. ein Asterisk oder Prozentzeichen). Ausnahmen davon sind die UDATE-, UDAY-, UMONTH- und UYEAR-Felder. Das PAGE(Seiten)-Feld jedoch, da ein in der Technik reserviertes Wort, kann mit einer alternativen Länge definiert werden und wie jedes andere Feld manipuliert werden.

Für Informationen zu Datum- und Zeitcodes beachten Sie bitte DATUM- und ZEIT-Typen beginnend auf Seite 69.

Es gibt viele verschiedene Gruppierungen von reservierten Wörtern in RPG, sie enthalten z. B.:

1. *Reserverd Fields (Reservierte Felder)* – Spezielle Feldnamen, die einen bestimmten Wert beinhalten. Diese Werte können typischerweise während der Laufzeit des Programmes geändert werden.
2. *Figurative Constants (Figurative Konstanten)* – Spezielle Feldnamen, deren Wert konstant ist oder vor dem Programmablauf geändert werden kann. Der Inhalt dieser Felder kann nicht verändert werden.
3. *Control Values (Kontrollwerte)* – Spezielle Werte, die das Funktionieren von Operationscode oder das Definieren von Ausgabewerten kontrollieren.
4. *Routines (Routinen)* – Spezielle Routinen innerhalb des RPG-Zyklus.
5. *Built-in Functions (Eingebaute Funktionen)* – Spezielle Funktionen, die zusätzliche Zeichen-, Mathematik- oder Ausdrucksfunktionen bereitstellen.

Diese Typen von reservierten Wörtern sind in den Tabellen der folgenden Seiten beschrieben.

Reserved Field	Description
NOT	Used in expressions to reverse the result of the expression. Note, the word NOT is the only reserved word in RPG. No other value (including fields) can be named NOT.
PAGE	Page number. This 4-digit numeric field is used as a page counter. It is incremented each time it is output. You can change the defined length of this field by simply declaring it in your program with another length.
PAGEn	Additional page counters. These 4-digit numeric fields are used as additional page counters. PAGEn, where n can be 1 to 7, offers 7 additional page counters. You can change the defined length of these fields by simply declaring it in your program with another length.
UDATE	Session date. This 6-digit numeric field is initialized to the run-date when the program is started. The format of this date is based on the format specified in the Header specification for the program.
UDAY	Session day. This 2-digit numeric field contains the day of the month.
UMONTH	Session month. This 2-digit numeric field contains the month.
UYEAR	Session year. This 2-digit numeric field contains the year.
*DATE	Used to retrieve the current date. This date value represents the date with 8 positions. For example, if the date format is *MDY, *DATE will contain mmddccyy, where cc=century; yy=year; mm=month; dd=day. Unlike UDATE, *DATE cannot be modified at program runtime. The format of this date is based on the format specified in the Header specification for the program
*MONTH	Used to retrieve the current month. This date value represents the month in mm format. Where mm=month.
*YEAR	Used to retrieve the current century and year. This date value represents the year in ccyy format, where cc=century; yy=year.
*DAY	Used to retrieve the current day. This date value represents the day of the month in dd format, where dd=day of the month.

Tabelle 5: Reservierte Wörter

Figurative Konstanten

Figurative Konstanten können mit den meisten Operationscodes verwendet werden und überall, wo ein Ausdruck erlaubt ist. Figurative Konstanten können innerhalb der Definitions- und der Rechenbestimmung verwendet werden.

Figurative Value	Description
*ALL'...'	Repeating pattern. Automatically adjusts to the size of the corresponding field that it is being compared with or moved to. For example: *ALL'abcd' moves 'abcdabcd' etc. to the Result field for the length of the Result field.
*ALLX'...'	Repeating hexadecimal pattern. Automatically adjusts to the size of the corresponding field that it is being compared with or moved to. For example: *ALLX'00' moves binary zeros to the Result field for the length of the Result field.
*ALLG'so...si'	Repeating graphic character set (DBCS) pattern. Automatically adjusts to the size of the corresponding DBCS field.
*BLANK *BLANKS	Blanks. Automatically adjusts to the size of corresponding field that it is being compared with or moved to them.
*END	Use the special value *END in Factor 1 of the SETLL operation to position the file <i>cursor</i> to the end of the file.
*HIVAL	Represents the highest possible value for the corresponding data type. It can be compared with or moved to a field.
*LOVAL	Represents the lowest possible value for the corresponding data type. It can be compared with or moved into a field.
*NULL	Use to set or compare pointer data types to "no value". Currently, RPG supports *NULL with the pointer data type.
*OFF	Logical off ('0'). Functionally similar to *ALL'0'. Typically, *OFF is used with the IFxx and MOVE operations to test or set the status of an indicator.
*ON	Logical on ('1'). Functionally similar to *ALL'1'. Typically, *ON is used with the IFxx and MOVE operations to test or set the status of an indicator.
*OMIT	Use this value on parameters that are being skipped during a bound call (i.e., CALLB or CALLP) operation.
*START	Use the special value *START in Factor 1 of the SETLL operation to position to the beginning of the file.
*ZERO *ZEROS	Used to represent a repeating pattern of zeros. Can be used with any data type, except <i>pointer, date, time, and timestamp</i> variables.
*ASTFILL	Used with the %EDITC built-in function to indicate that the edited value is to be padded on the left-side with leading asterisks.
*CURSYM	Used with the %EDITC built-in function to indicate that the currency symbol is to be inserted into the edited value.

Tabelle 6: Figurative Konstanten

Control Value	Operation Codes	Description
*ENTRY	PLIST	The *ENTRY parameter list identifies the parameter list used to pass parameters into and return parameters from the program.
*INZSR	BEGSR	The *INZSR subroutine, if specified in the program, is called by the RPG cycle before 1P output.
*TERMSR	BEGSR	The *TERMSR subroutine, if specified, is called by RPG when the program is ending—after all other processing is completed.
*LIKE	DEFINE	The *LIKE DEFINE operation code is used to define a new field, based on the attributes of another field. The types of fields that can be defined with *LIKE DEFINE are character and packed decimal.
*LOCK	IN	The *LOCK IN operation code is used to read a data area, then place an object-lock on that data area.
	OUT	The *LOCK OUT operation code is used to write a data area and retain the object-lock.
*LDA	DEFINE	The *DTAARA DEFINE *LDA operation code is used to assign a variable to receive the contents of the local data area.
*PDA	DEFINE	The *DTAARA DEFINE *PDA operation code is used to assign a variable to receive the program initialization parameters.
*DTAARA	DEFINE	The *DTAARA DEFINE operation code is used to declare the entry in Factor 2 as a data area. An optional field name can be specified in the Result field. If Factor 2 is not specified, the field name in the Result field is used as the data area name.
	IN	The IN *DTAARA operation code is used to read all data areas defined in the program. If *LOCK IN *DTAARA is specified, all data areas defined in the program are read and an object-lock is placed on each one.
	OUT	The OUT *DTAARA operation is used to write (i.e., output) all data areas defined in the program. If *LOCK OUT *DTAARA is specified, all data areas are written and any object-locks are retained.
*PSSR	BEGSR	The *PSSR subroutine, if specified in the program, is called by the RPG exception/error handling routine whenever an unmonitored error occurs.

Tabelle 7: Kontrollwerte, die in der Rechenbestimmung verwendet werden

Ausgabe-Kontrollwerte

Control Field	Description
*ALL	Output all fields. This control field causes all fields from an externally defined file to be output. It is used on Output specifications controlled by an EXCEPT operation code.
*PLACE	<p>Asterisk-Place. This control field replicates Output specifications, within the specific output line up to the position of the *PLACE. The replicated output is positioned at the end position specified for the *PLACE.</p> <p>*PLACE is (or more accurately, "was") used in label printing programs to print 2-, 3-, and 4-up labels. It was created primarily to automate printed output from 80-column card-based programs. It is virtually worthless today considering the on-line source editors available.</p>

Tabelle 8: Kontrollfelder, die in den Ausgabebestimmungen verwendet werden

Routine	Description
*CANCL	Cancel the program.
*DETC	Return to detail-time calculations.
*DETL	Return to detail-time lines (i.e., detail output).
*GETIN	Return to the next "get in" cycle.
*NEXT	Return to the statement following the one in which the error occurred.
*OFL	Return to the overflow output-time portion of the cycle.
*TOTC	Return to total-time calculations.
*TOTL	Return to total-time lines (i.e., total-time output).
Blanks	<p>If the *PSSR or INFSR subroutines were called by the EXSR or CASxx operation, control returns to the statement following the EXSR or CASxx operation. If the RPG exception/error handler called the subroutine, the following applies:</p> <p style="padding-left: 40px;">If the error status code is 1121 to 1126, control returns to the operation where the error occurred.</p> <p style="padding-left: 40px;">Any other error status code causes an exception to be issued and the requester is notified (i.e., a message is sent to the user).</p>

*Tabelle 9: Umkehrpunkte innerhalb *PSSR- und INFSR-Unter-routinen*

