# Series25 Pricing Formula Variables

Click here to view related articles.

Pricing formulas can be as simple as a flat rate or as complicated as a web of nested logic. This page shows you all the tools you can use to create formulas in your Rate Schedules. To see more examples, read Series25 Pricing Formula Examples.

#### How Formulas Are Calculated

For each line item in an invoice, 25Live chooses a rate schedule and a price sheet based on the organization's rate group and the rate schedule's billable items. If a rate schedule has multiple price sheets, the one with the latest effective date is used (as long as it is not after the event's pricing date).

If a price sheet has multiple formulas, all of them are calculated and added together, then their total is reflected on the line item in 25Live's pricing page (and on an invoice). Some formulas may be calculated as 0, especially when using breakpoints or conditional statements.

## Pricing Formulas and Repeating Events

When an event has multiple occurrences, pricing formulas are calculated separately for each occurrence and summed together to give a line item total. This behaves slightly differently for each type of rate schedule:

- For locations and resources, the calculations are only made on occurrences where the object is assigned. (Example: if a location is only assigned to a single day on an event that meets five times, then pricing is only calculated for that day.)
- For event types, calculations are made for every occurrence. (Example: if an event has five occurrences, its event type formula will be performed five times.)
- For requirements, calculations are only made once regardless of the number of occurrences. This makes requirements ideal for adding a flat fee to an event.

#### Flat Fees and Multi-Occurrence Events

If you want a rate schedule to charge a flat fee regardless of the number of occurrences, there are two variables that can help you out:

Variable	Description	Example Usage	Explanation
NumberOfOccurrences	Returns the total number of occurrences in the event	50/NumberOfOccurrences	Charges a flat \$50 because each occurrence contributes a fraction of that fee

Variable	Description	Example Usage	Explanation
FirstOccurrence	Returns 1 when calculating the first occurrence, returns 0 otherwise	50 * FirstOccurrence	Charges a flat \$50 because every occurrence but the first contributes \$0

## **Pricing Formula Variables**

The tables below list all the variables that can be used in price sheet formulas. Some variables are restricted and can only be used with certain types of rate schedules. (For example, the Capacity variable only applies to locations.)



#### **Requirement Rate Schedules**

The most common use case for event requirements in pricing is to apply a flat fee, so most formulas for these rate schedules are very simple. The only variable which applies to requirements is Quantity.

#### **Basic Information Variables**

These variables are used to return data about the event or current occurrence being calculated.

When calculating price for a segment, the time calculation only looks at that segment. Each segment adds up pricing independently. For example, a surcharge for events that are more than 8 hours long might not be triggered if it's split up into multiple segments, each less than 8 hours.

Note that variables referencing event times or reservation times are not affected by the Pricing Times setting.

Variable Name	Description
NumberOfOccurrences	Returns the total number of occurrences in the event
ExpHeadCount	Expected headcount for the event
RegHeadCount	Registered headcount for the event
ActHeadCount	Actual headcount for each occurrence (Location rate schedules only)
Quantity	Quantity assigned (Resource and requirement rate schedules only)
Capacity	Either capacity of layout or max capacity (Location rate schedules only)

Variable Name	Description
OccurrenceNumber	The position of the occurrence in a repeating list  Example: First occurrence = 1, second occurrence = 2
FirstOccurrence	Returns 1 when calculating the first occurrence, returns 0 otherwise

### **Event and Reservation Duration Variables**

These variables give absolute values for an occurrence's duration, unaffected by the Pricing Times setting.

Note that the "event time" of an occurrence refers to just the start and end times that appear on a calendar, while "reservation time" refers to the full setup/takedown/pre/post-event time.

Variable Name	Description
EventDuration	Event duration, expressed as a numeral: (EV_END_DT - EV_START_DT)  Note: The value is expressed in terms of the number of days.
ReservationDuration	Total reserved duration, expressed as a numeral: (RSRV_START_DT - RSRV_END_DT)  Note: Includes setup/takedown/pre/post times. The value is expressed in terms of the number of days. One and a half days is 1.5 and six hours is 0.25.
SetupDuration	Setup duration, expressed as a numeral  Note: The value is expressed in terms of the number of days.
PreEventTime	Pre-event duration, expressed as a numeral  Note: The value is expressed in terms of the number of days.
PostEventTime	Post-event duration, expressed as a numeral  Note: The value is expressed in terms of the number of days.
TakedownDuration	Takedown duration, expressed as a numeral  Note: The value is expressed in terms of the number of days.

#### Occurrence-Based Variables

These variables all return information based on the length of the occurrence, which can either be equivalent to event time (the start and end times that appear on a calendar) or reservation time (the full duration including

### CollegeNET Series25 Help and Customer Resources

setup/takedown/pre/post time). This is controlled by the Pricing Times setting.

Note that several variables include "Part" in the name, such as "OccurrencePartHours". These variables are intended for use with price sheet breakpoints. They only return values for parts of the occurrence which fall within the current breakpoint.

Variable Name	Description
OccurrenceDuration	Length of the occurrence in days  Equal to either ReservationDuration or EventDuration, depending on Pricing Time settings  Note: Used to break occurrences into blocks of less than 24 hours  For example: an event longer than 2 hours = OccurrenceDuration > 2 / 24 an event of 4 hours = OccurrenceDuration = 4 / 24
OccurrenceDays	Note: Equivalent to OccurrenceDuration rounded up to the next whole number  For example: a six-hour event = 1 4 p.m. Fri - 2 p.m. Sat = 1 4 p.m. Fri- 6 p.m. Sat = 2
OccurrenceHours	Number of hours, or part of, in this occurrence  Note: Equivalent to (OccurrenceDuration * 24) rounded up to the next whole number
OccurrenceMinutes	Number of minutes in this occurrence  Note: Equivalent to (OccurrenceDuration * 1440)
OccurrenceNights	Number of midnights spanned by the occurrence, or 1 if the occurrence doesn't span midnight  Note: Fri 6 p.m Sun 10 a.m. = 2 9 a.m 11 p.m. (same day) = 1
OccurrencePartDays	Number of days, or part of, that fall within the current breakpoint

/ariable Name	Description
OccurrencePartHours	Number of hours, or part of, that fall within the current breakpoint  For example:  Wed 9 a.m 2 p.m. OccurrenceHours = 5  for 0-3h, OccurrencePartHours = 3  for 4-8h, OccurrencePartHours = 2  for 0-10h, OccurrencePartHours = 5
OccurrencePartMinutes	Number of minutes, or part of, that fall within the current breakpoint
OccurrencePartDuration	Duration (in days) of the portion of the occurrence that falls within the current breakpoint
OccPartSunDuration OccPartMonDuration OccPartTueDuration OccPartWedDuration OccPartThuDuration OccPartFriDuration OccPartSatDuration	Durations of those portions of the occurrence that occur on the indicated days within this breakpoint  Note: These variables reflect just the portion of an occurrence that falls in the breakpoint period.
OccPartSunDays OccPartMonDays OccPartTueDays OccPartWedDays OccPartThuDays OccPartFriDays OccPartSatDays	The total number of each day within the current breakpoint  Note: These variables reflect just the portion of an occurrence that falls in the breakpoint period.
OccPartSunHours OccPartMonHours OccPartTueHours OccPartWedHours OccPartThuHours OccPartFriHours OccPartSatHours	The total number of hours in each day within the current breakpoint

Variable Name	Description
OccPartSunMinutes OccPartMonMinutes OccPartTueMinutes OccPartWedMinutes OccPartThuMinutes OccPartFriMinutes OccPartSatMinutes	The total number of minutes in each day within the current breakpoint
OnSun OnMon OnTue OnWed OnThu OnFri OnSat	The number of times the occurrence occurs on that day  Note: Several days could be set to 1 if the occurrence spans multiple days. If the occurrence spans several weeks, each day is counted.  For example:  Mon the 3rd - Tue the 11th, OnMon=2, OnTue=2, OnWed=1  These variables reflect just the portion of an occurrence that falls in the breakpoint period.
In00Hour, In01Hour, In12Hour, In13Hour, In23Hour  Note military time	The number of times the occurrence occurs in that hour  Note: Several hours could be set to 1 if the occurrence spans multiple hours. If the occurrence spans multiple days, each hour is counted.  For example:  Mon 10:30 a.m 2:45 p.m. In10Hour=1, In14Hour=1  These variables reflect just the portion of an occurrence that falls in the breakpoint period.
	An indicator of whether the occurrence occurs in a part of that hour, relative to the occurrence start  Note: If the occurrence spans multiple days, the second day rolls over from 1 to 2, and so on.
In01stHour,In02ndHour,In03rdHour In21stHour, In22ndHour,In23rdHour	For example:  Monday 10:30 a.m 2:45 p.m.
Note military time	In01stHour = 1 In02ndHour = 1 In03rdHour = 1 In04thHour = 1 In05thHour = 1 These variables reflect just a portion of an occurrence that falls in the
	These variables reflect just a portion of an occurrence that falls in the breakpoint period.

## Date/Time Variables

These variables return a date, time, or both. The most common usage of these variables is to use **extraction codes** to isolate specific values such as the day of the week. You can also compare these variables to a specific date or time values using **conversion functions**.

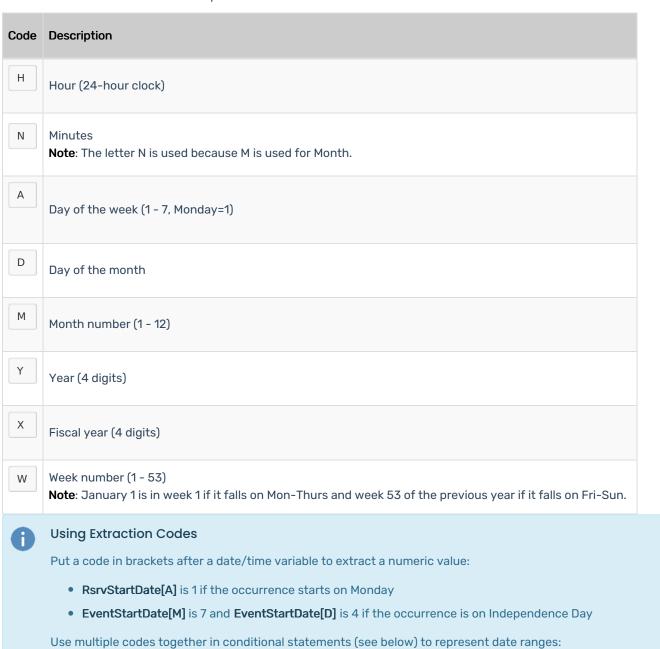
Note that like other variables, all date and time variables refer to an individual occurrence rather than the event as a whole. "Event time" of an occurrence refers to just the start and end times that appear on a calendar, while "reservation time" refers to the full setup/takedown/pre/post-event time. "Occurrence time" is equivalent to either event time or reservation time based on the Pricing Times setting.

Variable Name	Description
RsrvStartDateTime RsrvStartDate RsrvStartTime	Reservation start date, start time, or both
RsrvEndDateTime RsrvEndDate RsrvEndTime	Reservation end date, end time, or both
EventStartDateTime EventStartDate EventStartTime	Event start date, start time, or both
EventEndDateTime EventEndDate EventEndTime	Event end date, end time, or both
PreEventStartDateTime PreEventStartDate PreEventStartTime	Pre-event start date, start time, or both
PostEventEndDateTime PostEventEndDate PostEventEndTime	Post-event start date, start time, or both
OccStartDateTime OccStartDate OccStartTime	Occurrence start date, start time, or both  Note: An occurrence is defined as either an occurrence or an event, depending on how your system definition is set.

Variable Name	Description
OccEndDateTime OccEndDate OccEndTime	Occurrence end date, end time, or both  Note: An occurrence is defined as either an occurrence or an event, depending on how your system definition is set.

#### **Extraction Codes**

Date/time variables are used with brackets and extraction codes to produce numbers that can be used in formulas. The most common use case is to compare them



EventStartDate[M] >= 6 & EventStartDate[M] <= 8 ? 75 \* OccurrenceHours will charge \$75 per hour
during the months of June, July, and August</li>

#### **Conversion Functions**

Extraction codes allow you to work with date/time variables in terms of regular numbers. Conversion functions do the opposite: converting numbers into date or time values that can be compared directly to date/time variables.

Function	Description	Example
\$datim	converts a date and time (24-hour format)	4/27/2025 16:00
\$date	converts a date	4/27/2025
\$clock	converts a time (24-hour format)	16:00



### **Using Conversion Functions**

Conversion functions are most often used to compare date/time variables in conditional statements.

For example, to set a condition that \$50 per hour should be charged if the reservation ends after 9:00 p.m., you'd enter:

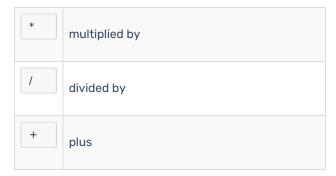
RsrvEndTime > \$clock("21:00") ? 50 \* ReservationDuration

There is some overlap here with using "specific times" breakpoints for a price sheet. The difference is that a formula using breakpoints only applies to the portion of the occurrence that falls within the specified times, while the example above applies to the whole occurrence (no matter how long it is) as long as its end time meets the condition.

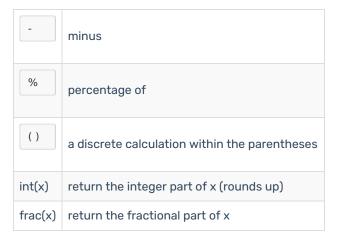
### Other Formula Functions

#### **Basic Math**

When writing formulas, you can use any of these standard operators:



### CollegeNET Series25 Help and Customer Resources



For example, with a flat fee, plus an hourly rate that rounds to the closest hour, use...

200+(100 \*(Int((EventDuration+PreEventTime+PostEventTime)24)))

## Conditions ("If" Statements)

You can create more complex formulas by specifying conditions. When a condition is specified in a formula, the rest of the formula is calculated only if the condition is true.

To specify a condition, add a question mark at the beginning of the formula and then put a comparison statement before it. Comparison statements include one of the following operators:



You can use any of the pricing formula variables described above in your comparisons.

For example:

ExpHeadCount > 100 ? 10 \* OccurrenceDuration

This formula will only apply the price (10 \* OccurrenceDuration) if the expected headcount is more than 100.

You can also include multiple comparison statements in a single condition if you use logical relationships between them:



For example

ExpHeadCount > 50 & ExpHeadCount < 100 ? 10 \* OccurrenceDuration</li>

This formula will only apply the price (10 \* OccurrenceDuration) if the expected headcount is between 50 and 100.

## Referencing Other Rate Groups

To make it easier to create a new formula, you may want to reference an existing Rate Group pricing formula and then make the needed adjustments. To do so, you enclose the existing Rate Group formula name in curly brackets. For example, if the Corporate rate is twice the Student Organizations rate, your formula for the Corporate rate would be:

2 \* {Student Organizations}

This formula would use the formula for Student Organizations to calculate the rate (such as overtime, staff fees, and so on) and then double it.



If you reference a Rate Group that doesn't exist (or doesn't have a pricing formula associated with it), the part of your formula that references that group resolves to 0 (zero).

## **Referencing Custom Attributes**

You can also reference custom attributes in your pricing formula. To do so, you enclose the custom attribute name in "at" symbols.

For example, if you have created a custom location attribute named SquareFootage, and you want to create a pricing formula where the price is determined by the square footage of the room, say \$10 per square foot, your formula can refer to the attribute as follows:

• 10 \* @SquareFootage@

If you have the same custom attribute name in multiple places, the pricing formula gives them precedence as follows:

- First precedence: event custom attribute
- · Second precedence: organization custom attribute
- Third precedence: location/resource custom attribute (only applicable when pricing a location or resource)



If your custom attribute has a space in the name (for example, "Square Footage"), you must include the space in your formula:

## CollegeNET Series25 Help and Customer Resources

• 10 \* @Square Footage@

PREVIOUS: Rate Schedules

**UP NEXT: Formula Examples**