



# PaaS平台函数使用手册

Function Manual For PaaS Platform

发行版本：1707

发布时间：2017 / 08 / 04

## 声明

您购买的产品、服务或特性等应受北京仁科互动信息科技有限公司商业合同和条款的约束，本文中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，北京仁科互动对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。非经本公司书面许可复制文档内容的部分或全部内容，不得以任何形式传播。

版权所有 © Copyright 2012 - 2017 | 仁科互动（北京）网络技术有限公司

## 联系我们

销售咨询：400-050-0907

渠道加盟：400-832-0718

客户服务：400-012-2980

销售热线：400-050-0907

渠道热线：400-832-0718

技术热线：400-012-2980

邮箱地址：info@xiaoshouyi.com, service@xiaoshouyi.com

官方网址：www.xiaoshouyi.com

# 阅读指南

## 手册目标

本文档主要介绍销售易 CRM PaaS 平台 4.6 版本中支持的所有函数，这些函数并非销售易的 API。在 PaaS 平台中设置一些值时，需要使用“插入函数”功能来辅助值的设置，比如为文本等类型字段设置默认值、为计算公式型字段设置值以及在验证规则中设置校验条件等。本文档主要对“插入函数”功能中支持的所有函数进行详细说明。

## 手册约定

本文档中所有标识为 **NOTE** 的部分表示需要您特别关注的内容。



# 目录

函数在 PaaS 平台中的应用位置.....	3
函数说明.....	9
日期和时间类函数.....	9
逻辑类函数.....	14
数学类函数.....	19
文本类函数.....	25
特殊情况说明.....	37
操作符 / 表达式对于 NULL 的处理方式.....	37
空白值参与运算出现非期待结果的解决方案.....	38



## 函数在 PaaS 平台中的应用位置

本文档介绍的所有函数应用于 PaaS 平台中的以下位置：

- 默认值

为字段设置“默认值”时，可通过函数辅助值的设置，如下图红框部分所示。关于默认值的详细介绍请参考《销售易 CRM\_PaaS 平台使用手册》自定义业务对象一章中的“设置字段”部分。

新建字段

选择字段类型：

- 文本
- 单选
- 多选
- 文本域
- 整数
- 实数
- 货币
- 日期

示例：

字段详细信息：

字段名称

API 名称：customItem8

最大长度：300

最小长度

帮助文本

默认值

插入运算符 插入函数 语法检查

**NOTE** 选择的字段类型不同，支持的函数类型也不相同，“插入函数”部分将根据当前选择的字段类型显示对应的函数。

- 计算公式

为计算公式类型字段设置值时，可通过函数辅助值的设置，如下图红框部分所示。关于计算公式的详细介绍请参考《销售易 CRM\_PaaS 平台使用手册》自定义业务对象一章中的“设置字段”部分。

新建字段

选择字段类型：

- 自动编号
- 关联关系
- 主子明细
- 引用字段
- 计算公式
- 电话
- 邮箱
- 网址

示例：

报价金额=单价\*数量\*折扣+运费

根据您定义的公式表达式派生其值的只读字段，任何来源字段有所更改，该计算型字段均将更新。

数字 计算公式

计算公式 =

插入字段 插入运算符 插入函数 语法检查

描述

参与计算的字段值为空时如何处理？

- 空白字段认为是0
- 空白字段认为是空白

小数点位数

0

**NOTE** 1. 函数不支持计算公式中“汇总累计”的计算类型。

2. 当为计算公式型字段设置值时，可能会出现参与计算的字段值为 NULL 的情况，关于此情况的处理方式及解决方案请参考[特殊情况说明](#)。



- 校验规则

为校验规则设置校验条件时，可通过函数辅助校验条件的设置，如下图红框部分所示。关于校验规则的详细介绍请参考《销售易 CRM\_PaaS 平台使用手册》自定义业务对象一章中的“设置校验规则”部分。

### 新建校验规则 关闭

规则名称： \*

状态：

描述：

**校验条件**  
示例：商机折扣 > 0.3。如果商机折扣大于30%，则显示错误信息。

**错误原因**  
示例：商机折扣不能大于30%。此消息将在满足错误条件时显示。

**NOTE** 由于校验条件的返回类型只能为布尔型，因此当使用函数辅助校验条件的设置时，请确保最终返回类型为布尔型。

- 触发事件 - 更新字段

在触发事件 - 更新字段中设置更新字段值时，可通过函数辅助更新字段值的设置，如下图红框部分所示。关于更新字段值的详细介绍请参考《销售易 CRM\_PaaS 平台使用手册》流程管理一章中的“添加更新字段事件”部分。

### 添加更新字段 关闭

**基本信息**

事件名称  \*

关联对象  \*

对象类型  \*

事件说明

通用事件     审批流引用事件

**规则**

更新字段  \*

更新字段值 字段名称 =  \*

- 触发事件 - 更新字段

在触发事件 - 更新字段中设置更新字段值时，可通过函数辅助更新字段值的设置，如下图红框部分所示。关于更新字段值的详细介绍请参考《销售易 CRM\_PaaS 平台使用手册》中流程管理一章中的“添加更新字段事件”部分。

### 添加更新字段 关闭

**基本信息**

事件名称  \*

关联对象  \*

对象类型  \*

事件说明

通用事件     审批流引用事件

**规则**

更新字段  \*

更新字段值 字段名称 =  \*

- 触发规则 - 设置条件

在触发规则中设置条件时如果公式类型选择“特殊公式”，则可通过函数辅助条件的设置，如下图红框部分所示。关于设置条件的详细介绍请参考《销售易 CRM\_PaaS 平台使用手册》中流程管理一章中的“设置触发规则”部分。

添加触发规则

基本信息 设置条件 引用事件

设置条件 (最多可设置20个)

公式类型 特殊公式

插入字段 插入运算符 插入函数 语法检查

上一步 保存并继续

## 函数说明

---

### 日期和时间类函数

销售易 CRM PaaS 平台 V1707 版本支持以下日期和时间类函数：

- [Date](#)
- [DateValue](#)
- [Day](#)
- [Month](#)
- [Now](#)
- [Today](#)
- [Year](#)

Date	
<b>描述</b>	根据输入的年月日的值返回一个日期类型的数据
<b>使用</b>	<p>DATE(<i>year,month,day</i>)</p> <p>返回值为日期型</p> <p><i>year</i>为四位数字，表示年</p> <p><i>month</i>为两位数字，表示月</p> <p><i>day</i>为两位数字，表示日</p>
<b>样例</b>	比如创建一个日期数值 2017-03-24，函数为 Date(2017,03,24)
<b>注意点</b>	<ol style="list-style-type: none"><li>1. 使用该函数时，支持的日期范围为 1990 年 1 月 1 日到 9999 年 12 月 31 日，如果是该范围之外的日期，系统计算异常，无法正常使用。范围之外的日期推荐使用 DateValue 函数来进行日期处理</li><li>2. 如果该函数中输入的不是合法的日期，则前台页面上显示为空，比如输入 2017-02-29，2017 不是闰年，该日期无效，前台显示为空</li></ol>

DateValue	
<b>描述</b>	根据输入的日期 / 时间或者文本类型的表达式返回一个年月日格式的日期值
<b>使用</b>	<p>DATEVALUE(<i>expression</i>)</p> <p>返回值为年月日格式的日期型</p> <p><i>expression</i>为日期 / 时间、文本类型的表达式或者字段</p>
<b>样例</b>	比如创建一个日期数值 2017 年 3 月 24 日，函数为 DateValue("2017-03-24")

## DateValue

### 注意点

1. 如果函数引用的字段不是有效的日期 / 时间或文本字段，则该日期无效，前台显示为空
2. 当输入日期时，需要使用引号将日期括起来，并使用以下格式：YYYY-MM-DD，YYYY 为四位数年份，MM 为两位数的月份，DD 为两位数的天
3. 如果表达式的值不在有效的日期范围内，比如 MM 为 01~12 以外的值，则该日期无效，前台显示为空
4. 此函数只能被用于日期型字段，不能被用于日期 + 时间型字段

## Day

### 描述

根据输入的日期返回该日期在一个月中的哪一天

### 使用

`DAY(date)`

返回值为整数型，范围为 1~31

*date* 为日期型字段或者返回结果为日期型的函数

返回指定日期的天：Day(DateValue("2017-03-24")) 返回的就是 24

### 样例

返回当天：Day(Today())

### 注意点

1. *date* 只能为日期型字段或者返回结果为日期型的函数，不支持以下形式的日期表达式 Day("2017-3-24")
2. *date* 不支持日期 + 时间类型

## Month

<b>描述</b>	根据输入的日期返回该日期在一年中的那个月份
<b>使用</b>	MONTH( <i>date</i> ) 返回值为实数型，范围为 1~12 <i>date</i> 为日期型字段或者返回结果为日期型的函数
<b>样例</b>	返回当月：Month(Today())
<b>注意点</b>	1. <i>date</i> 只能为日期型字段或者返回结果为日期型的函数，不支持以下形式的日期表达式 Month("2017-3-24") 2. <i>date</i> 不支持日期 + 时间类型

## Now

<b>描述</b>	返回触发对应元素的当前时刻的时间
<b>使用</b>	NOW() 返回值为日期时间类型 无输入参数
<b>样例</b>	比如返回当前时刻可以使用该函数 Now()
<b>注意点</b>	1. 此函数无参数，但括号必须保留 2. 该函数返回的是触发时的当前时间，比如在计算公式中只有一个 Now() 函数，那么这个计算公式返回的值就是当前数据的创建时间 3. 比如校验规则中配置 Now() 函数作为判断的条件，返回值为触发该校验规则时的当前时刻的时间



Today	
<b>描述</b>	返回当前日期
<b>使用</b>	TODAY() 返回值为日期类型 无输入参数
<b>样例</b>	Today()
<b>注意点</b>	<ol style="list-style-type: none"><li>1. 此函数无参数，但括号必须保留</li><li>2. 此函数可被用于日期型字段，不能被用于日期 + 时间型字段</li><li>3. 当此函数与其他日期型字段相减时，结果为实数型数据，表示两个日期之间的天数</li><li>4. 当此函数与数值进行加减运算时，结果为日期型数据，表示距离今天之前或者之后几天的日期</li></ol>

Year	
<b>描述</b>	根据输入的日期返回该日期所属的那个年份
<b>使用</b>	YEAR( <i>date</i> ) 返回值为实数类型，四位数字表示的年份，范围为 1900~9999 <i>date</i> 为日期型字段或者返回结果为日期型的函数
<b>样例</b>	<ol style="list-style-type: none"><li>1. 返回当年：Year(Today())</li><li>2. 返回指定日期的年份：Year(DATE(2017,03,24)) 返回值为 2017</li></ol>
<b>注意点</b>	<ol style="list-style-type: none"><li>1. <i>date</i> 只能为日期型字段或者返回结果为日期型的函数，不支持以下形式的日期表达式 Year("2017-3-24")</li><li>2. <i>date</i> 不支持日期 + 时间类型</li></ol>

## 逻辑类函数

销售易 CRM PaaS 平台 V1707 版本支持以下逻辑类函数：

- Case
- If
- IsNull
- IsNumber
- Not
- NullValue
- Or
- And

## Case

<b>描述</b>	将表达式的值与给定的一组值进行比较，如果表达式的值等于其中的某个值，则返回相应的结果，如果表达式的值不等于其中的任何一个值，则返回设定的其他值（ <i>else_result</i> ）
<b>使用</b>	<p><code>CASE(expression, value1, result1, value2, result2,..., else_result)</code></p> <p>返回值无固定类型，根据 <i>result</i> 或者 <i>else_result</i> 的类型而定</p> <p><i>expression</i> 为字段或者用于比较的任何值</p> <p><i>value</i> 和 <i>result</i> 成对出现，<i>value</i> 用于与 <i>expression</i> 进行比较，<i>result</i> 为相应的返回结果</p> <p><i>else_result</i> 为当 <i>expression</i> 不等于任何 <i>value</i> 时，需要返回的值</p>
<b>样例</b>	<p>客户级别描述为计算公式类型字段，其值根据客户级别自动计算，如下（<i>Customer</i> 为业务对象客户的 API 名称，<i>customer_level</i> 为业务对象客户中的字段客户级别的 API 名称）</p> <p>Case(<i>Customer.customer_level</i>, "A", "VIP 客户", "B", "普通客户", "C", "潜在客户", "其他")</p>
<b>注意点</b>	所有 <i>value</i> 的数据类型必须相同，所有 <i>result</i> 的数据类型必须相同

## If

<b>描述</b>	判断表达式的值为真还是假，条件为真（TRUE），返回前一个值，条件为假（FALSE），返回后一个值
<b>使用</b>	<p><code>IF(logical_test, value_if_true, value_if_false)</code></p> <p>返回值无固定类型，根据 <i>value_if_true</i> 或者 <i>value_if_false</i> 的类型而定</p> <p><i>logical_test</i> 为要判断的表达式</p> <p><i>value_if_true</i> 为表达式为真时返回的结果</p> <p><i>value_if_false</i> 为表达式为假时返回的结果</p>
<b>样例</b>	<p>合同状态为计算公式类型字段，其值根据合同到期时间自动计算，即显示合同是否过期，如下（<i>Contract</i> 为业务对象合同的 API 名称，<i>expiration_time</i> 为业务对象合同中的字段合同到期时间的 API 名称）</p> <p>If(<i>Contract.expiration_time</i>&lt;Today(), "已到期", "未到期")</p>

**If**

**注意点** *value\_if\_true* 和 *value\_if\_false* 的数据类型必须相同

**IsNull**

**描述** 判断表达式是否为空 (null) 或者空白 (blank), 如果为空或者空白, 则返回 TRUE, 否则返回 FALSE

**使用** ISNULL(*expression*)  
返回值为布尔型, 返回 TRUE 或者 FALSE  
*expression* 为要判断的表达式

**样例** 判断业务对象订单中的字段订单描述是否为空, 如下 ( Order 为业务对象订单的 API 名称, description 为业务对象订单中的 字段描述的 API 名称 ):  
IsNull(Order.description)

**注意点** 对于实数或者整数类型的字段, 如果该字段上值为空, 该函数返回值为 true

**IsNumber**

**描述** 判断文本值是否为数字, 如果为数字, 则返回 TRUE, 否则返回 FALSE

**使用** ISNUMBER(*Text*)  
返回值为布尔型, 返回 TRUE 或者 FALSE  
*Text* 为文本型字段或者包含在双引号中的字符串

**样例** 判断业务对象订单中的字段账号是否为数字, 如下 ( Order 为业务对象订单的 API 名称, account\_number 为业务对象订单中的字段账号的 API 名称 ):  
IsNumber(Order.account\_number)

**注意点** 无

Not	
<b>描述</b>	如果表达式为 TRUE，则返回 FALSE，如果表达式为 FALSE，则返回 TRUE
<b>使用</b>	<p>NOT(<i>logical</i>)</p> <p>返回值为布尔型，返回 TRUE 或者 FALSE</p> <p><i>logical</i>为逻辑表达式</p>
<b>样例</b>	<p>显示所有的非潜在客户，如果为非潜在客户，显示客户名称，否则为空，如下（Custom 为业务对象客户的 API 名称，customer_level 为业务对象客户中的字段客户级别的 API 名称，customer_name 为业务对象客户中的字段客户名称的 API 名称）：</p> <pre>if(Not(IsPickVal(Custom.customer_level,"C")),Custom.customer_name,"")</pre>
<b>注意点</b>	无

NullValue	
<b>描述</b>	根据参数中是否为空值来返回原值或者特定的值，从而保证没有空值 (Null) 值的出现
<b>使用</b>	<p>NULLVALUE(<i>expression, substitute_expression</i>)</p> <p>返回值类型为输入参数中的返回值类型</p> <p><i>expression</i> 为用于判断的字段或者字符串</p> <p><i>substitute_expression</i> 表示的是如果第一个参数为空值时返回的表达式值</p>
<b>样例</b>	比如 NULLVALUE(Order.OrderDueDate,Order.CreateAt + 5), 如果 OrderDueDate 为空值，返回 CreateAt + 5 的日期；如果 OrderDueDate 不为空值，返回 OrderDueDate
<b>注意点</b>	<ol style="list-style-type: none"> <li>1. 日期、数字等类型字段是只有空值 (Null) 或者有值</li> <li>2. 如果文本类型字段值为 " "，该函数判断为 False，只有是空值或者 Null 值，该函数判断为 True</li> </ol>

## Or

<b>描述</b>	判断多个逻辑表达式中是否有表达式的结果为 TRUE，并根据判断结果返回 TRUE 或者 FALSE。仅当所有表达式全部为 FALSE 时才返回 FALSE
<b>使用</b>	OR( <i>logical1, logical2...</i> ) 返回值为布尔型，返回 TRUE 或者 FALSE <i>logical</i> 为逻辑表达式
<b>样例</b>	比如建立一个验证规则，当商品折扣小于 10% 或者大于 90% 时显示错误信息，如下（PromotionCommodity 为业务对象促销商品的 API 名称，commodity_discount 为业务对象促销商品中的字段折扣的 API 名称）： OR(PromotionCommodity.commodity_discount<0.1,PromotionCommodity.commodity_discount<0.9)
<b>注意点</b>	无

## And

<b>描述</b>	判断多个逻辑表达式中是否有表达式的结果为 FALSE，并根据判断结果返回 TRUE 或者 FALSE。仅当所有表达式全部为 TRUE 时才返回 TRUE
<b>使用</b>	AND( <i>logical1, logical2...</i> ) 返回值为布尔型，返回 TRUE 或者 FALSE <i>logical</i> 为逻辑表达式
<b>样例</b>	比如建立一个验证规则，当商品折扣大于 90% 且小于 99% 时显示错误信息，如下（PromotionCommodity 为业务对象促销商品的 API 名称，commodity_discount 为业务对象促销商品中的字段折扣的 API 名称）： AND(PromotionCommodity.commodity_discount>0.9,PromotionCommodity.commodity_discount<0.99)
<b>注意点</b>	无

## 数学类函数

销售易 CRM PaaS 平台 V1707 版本支持以下数学类函数：

- ABS
- Ceiling
- LN
- Log
- Max
- Min
- Mod
- Round
- SQRT
- Floor

**ABS**

<b>描述</b>	计算一个数字的绝对值
<b>使用</b>	$ABS(number)$ 返回值为整数型或者实数型 $number$ 为整数、实数型字段或者其他要计算的数值
<b>样例</b>	$ABS(-7.7)$ 此公式返回结果为 7.7 $ABS(-7)$ 此公式返回结果为 7
<b>注意点</b>	无

**Ceiling**

<b>描述</b>	将一个数字向上取整，即取整为大于等于自身的最小整数
<b>使用</b>	$CEILING(number)$ 返回值为实数型 $number$ 为整数、实数型字段或者其他要计算的数值
<b>样例</b>	$Ceiling(7.1)$ 此公式返回结果为 8.0 $Ceiling(-7.75)$ 此公式返回结果为 -7.0
<b>注意点</b>	此函数取整时不进行四舍五入，直接向上取整



**LN**

<b>描述</b>	计算一个数字的自然对数
<b>使用</b>	LN( <i>number</i> ) 返回值实数型 <i>number</i> 为整数、实数型字段或者其他要计算的数值
<b>样例</b>	LN(10) 此公式返回结果为 2.302585092994046
<b>注意点</b>	1. <i>number</i> 必须为大于 0 的数值，如果为非法值，则前台显示为空 2. 返回值的小数位根据配置的字段类型上的小数位的配置

**Log**

<b>描述</b>	计算 n 的以 10 为底的对数
<b>使用</b>	LOG( <i>number</i> ) 返回值为实数型 <i>number</i> 为整数、实数型字段或者其他要计算的数值
<b>样例</b>	Log(10) 此公式返回结果为 1.0
<b>注意点</b>	1. <i>number</i> 必须为大于 0 的数值，如果为非法值，则前台显示为空 2. 返回值的小数位根据配置的字段类型上的小数位的配置

**Max**

<b>描述</b>	返回一组数据中的最大值
<b>使用</b>	$MAX(number, number, \dots)$ 返回值为整数型或者实数型 <i>number</i> 为整数、实数型字段或者其他要计算的数值
<b>样例</b>	$Max(1, 15, 100, 100.6)$ 此公式返回结果为 100.6 $Max(-1.5, -2, -3)$ 此公式返回结果为 -1.5
<b>注意点</b>	无

**Min**

<b>描述</b>	返回一组数据中的最小值
<b>使用</b>	$MIN(number, number, \dots)$ 返回值为整数型或者实数型 <i>number</i> 为整数、实数型字段或者其他要计算的数值
<b>样例</b>	$Min(-2.1, 4, 2, 5, 34)$ 此公式返回结果为 -2.1
<b>注意点</b>	无

**Mod**

<b>描述</b>	取余，返回一个数值除以除数后的余数
<b>使用</b>	$MOD(number, divisor)$ 返回值为整数型 <i>number</i> 为被除数，可以为整数、实数型字段或者其他要计算的数值 <i>divisor</i> 为除数，可以为整数、实数型字段或者其他要计算的数值

## Mod

<b>样例</b>	<p>MOD(15,4) 此公式返回结果为 3</p> <p>MOD(-9,-3) 此公式返回结果为 0</p> <p>比如建立一个验证规则，活动推广开始日期都安排在周末，当安排在工作日时，显示错误信息，如下（ActivationRecord 为业务对象活动记录的 API 名称，start_date 为业务对象活动记录中的字段推广开始日期的 API 名称）：</p> <p>MOD(ActivationRecord.start_date-DATEVALUE("1900-1-7"),7)&lt;&gt;0&amp;&amp;MOD(ActivationRecord.start_date-DATEVALUE("1900-1-7"),7)&lt;&gt;6</p>
<b>注意点</b>	<i>divisor</i> 不能为 0，如果为非法值，则前台显示为空

## Round

<b>描述</b>	按指定的位数舍入数值
<b>使用</b>	<p>ROUND(<i>number,num_digits</i>)</p> <p>返回值为实数型</p> <p><i>number</i> 为整数、实数型字段或者其他要计算的数值</p> <p><i>num_digits</i> 为需要保留的小数位数（不含小数点），范围为 0~5</p>
<b>样例</b>	<p>Round(123.678,0) 此公式返回结果为 124</p> <p>Round(-123.678,0) 此公式返回结果为 -124</p> <p>Round(123.6789,2) 此公式返回结果为 123.68</p> <p>Round(-123.6789,2) 此公式返回结果为 -123.68</p>
<b>注意点</b>	舍入数值时自动按照四舍五入的方式取整

## SQRT

<b>描述</b>	计算一个数字的正平方根
<b>使用</b>	SQRT( <i>number</i> ) 返回值为实数型 <i>number</i> 为整数、实数型字段或者其他要计算的数值
<b>样例</b>	SQRT(25) 此公式返回结果为 5.0
<b>注意点</b>	1. <i>number</i> 不能为负数，如果为非法值，则前台显示为空 2. 返回值的小数位根据配置的字段类型上的小数位的配置

## Floor

<b>描述</b>	将一个数字向下取整，即取整为小于等于自身的最大整数
<b>使用</b>	FLOOR( <i>number</i> ) 返回值为实数型 <i>number</i> 为整数、实数型字段或者其他要计算的数值
<b>样例</b>	Floor(7.7) 此公式返回结果为 7.0 Floor(-7.35) 此公式返回结果为 -8.0
<b>注意点</b>	此函数取整时不进行四舍五入，直接向下取整

## 文本类函数

销售易 CRM PaaS 平台 V1707 版本支持以下文本类函数：

- [Begins](#)
- [Contains](#)
- [Find](#)
- [Includes](#)
- [IsPickVal](#)
- [Left](#)
- [Len](#)
- [Lower](#)
- [LPAD](#)
- [MID](#)
- [Right](#)
- [RPAD](#)
- [Substitute](#)
- [Text](#)
- [Trim](#)
- [Upper](#)
- [Value](#)
- [HyperLink](#)

## Begins

<b>描述</b>	检查文本是否以特定字符或者字符串开头，如果是，则返回 TRUE，否则返回 FALSE
<b>使用</b>	<p><code>BEGINS(text, compare_text)</code></p> <p>返回值为布尔型，返回 TRUE 或者 FALSE</p> <p><i>text</i> 为要检查的文本，可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数 "</p> <p><i>compare_text</i> 为要比较的文本字段或者其他文本值</p>
<b>样例</b>	<p>比如检查业务对象订单中的字段订单号是否以 " SO#" 开头，如下（ Order 为业务对象订单的 API 名称， order_id 为业务对象订单中的字段订单号的 API 名称）：</p> <p><code>Begins(Order.order_id, "SO#")</code></p>
<b>注意点</b>	<ol style="list-style-type: none"> <li>1. 此函数中的参数值区分大小写字母</li> <li>2. 如果要检查的文本字段（ <i>text</i>）为空，则此函数返回结果为 FALSE</li> </ol>

## Contains

<b>描述</b>	检查文本是否包含特定字符或者字符串，如果包含，则返回 TRUE，否则返回 FALSE
<b>使用</b>	<p><code>CONTAINS(text, compare_text)</code></p> <p>返回值为布尔型，返回 TRUE 或者 FALSE</p> <p><i>text</i> 为要检查的文本，可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数</p> <p><i>compare_text</i> 为要比较的文本，可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数 "</p>
<b>样例</b>	<p>比如检查业务对象商机报备中的字段行业是否包含 " 教育 "，如下（ OpportunityRegister 为业务对象商机报备的 API 名称， profession 为业务对象商机报备中的字段行业的 API 名称）：</p> <p><code>Contains(OpportunityRegister.profession, " 教育 ")</code></p>

## Contains

### 注意点

1. 此函数中的参数值区分大小写字母
2. 如果要检查的文本字段 ( *text* ) 为空, 则此函数返回结果为 FALSE

## Find

### 描述

返回一个字符串在另一个字符串中的位置

### 使用

`FIND(search_text, text [, start_num])`

返回值为整数型

*search\_text* 为要查找的字符串, 可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数

*text* 为要搜索的字符串, 可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数

*start\_num* 为搜索字符串的起始位置, 即从 *text* 的第几个字符开始从左到右查找 *search\_text*。此参数为可选, 如果缺省, 则默认从 *text* 的第一个字符开始搜索

### 样例

`Find("@xiaoshouyi", "xsy@xiaoshouyi.com")` 此公式返回结果为 4  
`Find("@xiaoshouyi", "xsy@xiaoshouyi.com", 3)` 此公式返回结果为 2

### 注意点

1. 返回的位置为相对值, 即相对于开始搜索的位置, 开始搜索的位置从 1 开始计
2. 当 *search\_text* 在 *text* 中不存在时, 返回 0
3. 此函数中的参数值区分大小写字母
4. 如果要检查的文本字段 ( *text* ) 为空, 则此函数返回结果为 0
5. *start\_num* 参数左右的括号 “[” 和 “]” 仅用于说明此参数为可选参数, 不属于参数的一部分, 实际使用时应去掉

## Includes

<b>描述</b>	判断多选选项列表字段中当前选择的任何值是否等于指定的文本文字，如果等于，返回 TRUE，否则返回 FALSE
<b>使用</b>	INCLUDES( <i>multiselect_picklist_field</i> , <i>text_literal</i> ) 返回值为布尔型，返回 TRUE 或者 FALSE <i>multiselect_picklist_field</i> 为多选类型字段 <i>text_literal</i> 为要比较的文本，可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数
<b>样例</b>	比如判断客户的个人兴趣的当前所有选择项中是否有“体育”的选项，如下（Account 为业务对象客户的 API 名称，Favorite 为业务对象客户中的字段 个人兴趣的 API 名称）： Includes(Account.Favorite, " 体育 ")
<b>注意点</b>	此函数判断的范围为多选类型字段当前所有选择的项，而不是多选类型可选的所有项

## IsPickVal

<b>描述</b>	判断单选选项列表字段中当前选择的值是否等于指定的文本文字，如果等于，则返回 TRUE，否则返回 FALSE
<b>使用</b>	ISPICKVAL( <i>picklist_field</i> , <i>text_literal</i> ) 返回值为布尔型，返回 TRUE 或者 FALSE <i>picklist_field</i> 为单选类型字段 <i>text_literal</i> 为要比较的文本，可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数
<b>样例</b>	比如判断业务对象客户中的单选类型字段国家的当前选择项是否为“俄罗斯”，如下（Customer 为业务对象客户的 API 名称，country 为业务对象客户中的字段国家的 API 名称）：IsPickVal(Customer.country, " 俄罗斯 ")
<b>注意点</b>	此函数判断的范围为单选类型字段当前选择的项，而不是单选类型可选的所有项



## Left

<b>描述</b>	对一个字符串从左侧起截取指定长度的字符串	
<b>使用</b>	LEFT( <i>text</i> , <i>num_chars</i> ) 返回值为文本型，返回截取的字符串 <i>text</i> 为需要截取的字符串，可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数 <i>num_chars</i> 为从字符串左侧起需要截取的字符串长度（字符数）	
<b>样例</b>	Left("销售易", 2)	此公式返回结果为销售
	Left("www.xiaoshouyi.com", 2)	此公式返回结果为 ww
<b>注意点</b>	<ol style="list-style-type: none"> <li>1. 在此函数中，一个英文字母和一个汉字都被认为是一个字符</li> <li>2. 当 <i>num_chars</i> 为 0 或者负数时，此函数的返回结果在前台显示为空</li> <li>3. 当要截取字符串的字段为空时，此函数的返回结果在前台也显示为空</li> </ol>	

## Len

<b>描述</b>	返回一个文本字符串所包含的字符数	
<b>使用</b>	LEN( <i>text</i> ) 返回值为整数型 <i>text</i> 为需要计算字符数的字符串，可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数	
<b>样例</b>	Len("销售易")	此公式返回结果为 3
	Len("www.xiaoshouyi.com")	此公式返回结果为 18
<b>注意点</b>	<ol style="list-style-type: none"> <li>1. 在此函数中，一个英文字母和一个汉字都被认为是一个字符</li> <li>2. 当要计算字符数的字段为空时，此函数返回结果为 0</li> </ol>	

## Lower

**描述** 将一个文本字符串中的所有字母转换为小写字母

**使用** LOWER(*text*)  
返回值为文本型，返回转换后的文本字符串  
*text* 为需要转换的字符串，可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数

**样例** Lower("WWW.xiaoshouyi.COM")  
此公式返回结果为 www.xiaoshouyi.com  
Lower(" 销售易：WWW.xiaoshouyi.COM")  
此公式返回结果为销售易：www.xiaoshouyi.com

**注意点**

1. 当 *text* 中包含非英文字母时，仅对字母部分进行转换，非英文字母部分不做任何处理，原样返回
2. 当要转换的字段为空时，此函数的返回结果在前台也显示为空

## LPAD

**描述** 使用空格或者指定字符串从左侧填充另外一个字符串，使被填充字符串达到指定长度

**使用** LPAD(*text*, *padded\_length* [, *pad\_string*])  
返回值为文本型，返回填充后的文本字符串  
*text* 为需要被填充的字符串，可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数  
*padded\_length* 为填充后字符串需要达到的长度（字符数）  
*pad\_string* 为指定的填充字符串。此参数为可选，如果缺省，则默认为一个空格

## LPAD

<b>样例</b>	LPAD("xsy", 10, "a")	此公式返回结果为 aaaaaaaxsy
	LPAD("xsy", 10, " 销售易 ")	此公式返回结果为销售易销售易销 xsy
	LPAD("xsy", 3, "a")	此公式返回结果为 xsy
	LPAD("xsy", 2, "a")	此公式返回结果为 xs
	LPAD("xsy", 10)	此公式返回结果为 xsy

### 注意点

1. 在此函数中，一个英文字母和一个汉字都被认为是一个字符
2. 当填充一次 *pad\_string* 仍然达不到指定长度时，将会多次填充（见第一个样例）
3. 当多次填充 *pad\_string* 时，在最后一次填充时 *pad\_string* 可能被截取（见第二个样例）
4. 当 *padded\_length* 的值与 *text* 的长度相同时，不做任何处理（见第三个样例）
5. 当 *padded\_length* 的值小于 *text* 的长度相同时，自动截取（见第四个样例）
6. 当使用空格填充时，会自动忽略左侧的空格（见第五个样例）
7. 当被填充的字段为空时，此函数的返回结果在前台也显示为空

## MID

**描述** 对一个字符串从指定的起始位置开始从左到右截取指定长度的字符串

### 使用

MID(*text*, *start\_num*, *num\_chars*)

返回值为文本型，返回截取的字符串

*text* 为需要截取的字符串，可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数

*start\_num* 为要截取的字符串起始位置

*num\_chars* 为要截取的长度（字符数）

**样例** MID(" 销售易 www.xiaoshouyi.com", 2, 5) 此公式返回结果为售易 www

## MID

<b>注意点</b>	<ol style="list-style-type: none"> <li>1. 在此函数中，一个英文字母和一个汉字都被认为是一个字符</li> <li>2. 当 <i>start_num</i> 为 0 或者负数时，不会报错，将直接从字符串第一个位置开始截取</li> <li>3. 当 <i>num_chars</i> 的值大于 <i>text</i> 的长度时，不会报错，将直接截取到字符串最后一个字符</li> <li>4. 当要截取字符串的字段为空时，此函数的返回结果在前台也显示为空</li> </ol>
------------	---

## Right

<b>描述</b>	对一个字符串从右侧起截取指定长度的字符串
<b>使用</b>	<p><code>RIGHT(text, num_chars)</code></p> <p>返回值为文本型，返回截取的字符串</p> <p><i>text</i> 为需要截取的字符串，可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数</p> <p><i>num_chars</i> 为从字符串右侧起需要截取的字符串长度（字符数）</p>
<b>样例</b>	<p><code>Right(" 销售易 ", 2)</code>                      此公式返回结果为 售易</p> <p><code>Right("www.xiaoshouyi.com", 2)</code>      此公式返回结果为 om</p>
<b>注意点</b>	<ol style="list-style-type: none"> <li>1. 在此函数中，一个英文字母和一个汉字都被认为是一个字符</li> <li>2. 当 <i>num_chars</i> 为 0 或者负数时，此函数的返回结果在前台显示为空</li> <li>3. 当要截取字符串的字段为空时，此函数的返回结果在前台也显示为空</li> </ol>

## RPAD

**描述** 使用空格或者指定字符串从右侧填充一个字符串，使其达到指定长度

**使用** `RPAD(text, padded_length [, pad_string])`  
 返回值为文本型，返回填充后的文本字符串  
*text* 为需要被填充的字符串，可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数  
*padded\_length* 为填充后字符串需要达到的长度（字符数）  
*pad\_string* 为指定的填充字符串。此参数为可选，如果缺省，则默认认为一个空格

**样例**

<code>RPAD("xsy", 10, "a")</code>	此公式返回结果为 xsyaaaaaaaa
<code>RPAD("xsy", 10, "销售易 ")</code>	此公式返回结果为 xsy 销售易销售易销
<code>RPAD("xsy", 3, "a")</code>	此公式返回结果为 xsy
<code>RPAD("xsy", 2, "a")</code>	此公式返回结果为 xs
<code>RPAD("xsy", 10)</code>	此公式返回结果为 xsy

**注意点**

1. 在此函数中，一个英文字母和一个汉字都被认为是一个字符
2. 当填充一次 *pad\_string* 仍然达不到指定长度时，将会多次填充（见第一个样例）
3. 当多次填充 *pad\_string* 时，在最后一次填充时 *pad\_string* 可能被截取（见第二个样例）
4. 当 *padded\_length* 的值与 *text* 的长度相同时，不做任何处理（见第三个样例）
5. 当 *padded\_length* 的值小于 *text* 的长度相同时，自动截取（见第四个样例）
6. 当使用空格填充时，会自动忽略左侧的空格（见第五个样例）
7. 当被填充的字段为空时，此函数的返回结果在前台也显示为空

## Substitute

**描述** 使用指定文本替换文本字符串中的特定文本

**使用** `SUBSTITUTE(text, old_text, new_text)`  
 返回值为文本型，返回替换后的文本字符串  
*text* 为需要替换的字符串，可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数  
*old\_text* 为需要被替换的文本，可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数  
*new\_text* 为替换文本，可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数

**样例** 比如将业务对象促销商品中的字段促销方式中的 " 打折 " 更改为 " 满减 " ，如下（PromotionCommodity 为业务对象促销商品的 API 名称，commodity\_way 为业务对象促销商品中的字段促销方式的 API 名称）：  
`Substitute(PromotionCommodity.commodity_way, " 打折 ", " 满减 ")`

**注意点**

1. 此函数中的参数值区分大小写字母
2. 如果 *new\_text* 为 "" ，则不做任何处理
3. 如果在需要替换的字符串中包含多个需要被替换的文本，则将被替换多次

## Text

**描述** 将实数、整数、日期等其他类型的数据转换为文本类型

**使用** `TEXT(value)`  
 返回值为文本型  
*value* 为需要转换的数据

**样例** 比如将业务对象订单中的字段数量与单位进行拼接，如下（Order 为业务对象订单的 API 名称，quantity 为业务对象订单中的字段数量的 API 名称，unit 为业务对象订单中的字段单位的 API 名称）：  
`Text(Order.quantity)&Order.unit`

## Text

### 注意点

1. 当需要转换的数据为百分比类型时，则转换后的文本为实际值（不带百分号），比如转换的数据为 10%，则转为文本后显示为 0.1
2. 当需要转换的数据为日期类型时，则转换后的文本显示的日期格式为 YYYY-MM-DD，其中 YYYY 为四位数表示的年，MM 为两位数表示的月份，DD 为两位数表示的天，连接符 - 也将被保留

## Trim

### 描述

删除文本字符串的开头结尾所有空格和制表符，对于文本字符串中间的空格和制表符不进行删除

### 使用

TRIM(*text*)

返回值为文本型，返回删除完成后的字符串  
*text* 为需要转换的数据

### 样例

TRIM(" 销 售 易 www.xiaoshouyi.com ")  
此公式返回结果为销 售 易 www.xiaoshouyi.com

### 注意点

无

## Upper

### 描述

将一个文本字符串中的所有字母转换为大写字母

### 使用

UPPER(*text*)

返回值为文本型，返回转换后的文本字符串  
*text* 为需要转换的字符串，可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数 "

### 样例

Upper("WWW.xiaoshouyi.COM")  
此公式返回结果为 WWW.XIAOSHOUYI.COM  
Upper(" 销售易 : WWW.xiaoshouyi.COM")  
此公式返回结果为销售易 : WWW.XIAOSHOUYI.COM

## Upper

- 注意点**
1. 当 *text* 中包含非英文字母时，仅对字母部分进行处理，非英文字母部分不做任何处理，原样返回
  2. 当要转换的字段为空时，此函数的返回结果在前台也显示为空

## Value

**描述** 将表示数值的文本字符串转换为数值

**使用** VALUE(*text*)  
返回值为整数型或者实数型  
*text* 为需要转换的字符串，可以为文本字段、双引号括起来的文本字符串或者返回值为文本类型的函数

**样例** Value(12.34567) 此公式返回结果为 12.34567

**注意点** 参数 *text* 的内容应为数字，如果包含除小数点以外的其他非法字符，则前台显示为空

## HyperLink

**描述** 创建超链接

**使用** HYPERLINK(*url*, *friendly\_name* [, *target*])  
返回为超链接类型的文本  
*url* 为需要链接到的 Web 地址  
*friendly\_name* 为显示的链接文本，即点击此文本将链接到设置的 Web 地址  
*target* 为点开超链接的位置，参数值包含 "\_blank", "\_parent", "\_self", "\_top"，该参数的含义与 HTML 中的链接中的参数函数一致

**样例** HyperLink("http://www.xiaoshouyi.com", "销售易") 或者  
HyperLink("http://www.xiaoshouyi.com", "销售易", "\_blank")

**注意点** *target* 参数左右的括号 “[” 和 “]” 仅用于说明此参数为可选参数，不属于参数的一部分，实际使用时应去掉



## 特殊情况说明

### 操作符 / 表达式对于 NULL 的处理方式

当为计算公式型字段设置值时，可能会出现参与计算的字段值为 NULL（字段值为空白）的情况，下面主要介绍各运算符 / 表达式在 NULL 值参与运算时的处理方式，如下表所示。

运算符 / 表达式	处理方式
加法 (+)	数字 + NULL = NULL
	NULL + 数字 = NULL
	日期 + NULL = NULL
	日期时间 + NULL = NULL
	NULL + NULL = NULL
减法 (-)	数字 - NULL = NULL
	NULL - 数字 = NULL
	日期 - NULL = NULL
	日期时间 - NULL = NULL
	NULL - 日期 = NULL
	NULL - 日期时间 = NULL
NULL - NULL = NULL	
乘法 (*)	两个参数中任意一个为 NULL，结果为 NULL
除法 (/)	两个参数中任意一个为 NULL，结果为 NULL
幂表	两个参数中任意一个为 NULL，结果为 NULL
字符串链接 (&)	假设参与运算的为 v1 和 v2
	1. 如果 v1 和 v2 都为 NULL，则结果为 NULL
	2. 如果 v1 为 NULL，v2 不为 NULL，则结果为 v2
	3. 如果 v1 不为 NULL，v2 为 NULL，则结果为 v1

运算符 / 表达式	处理方式
布尔表达式	<p>操作符包括 =、==、&gt;、&lt;、&gt;=、&lt;=、!= 和 &lt;&gt;</p> <p>假设参与运算的为 v1 和 v2</p> <ol style="list-style-type: none"> <li>如果 v1 和 v2 都为 NULL，则 v1 和 v2 通过上述运算符进行比较的结果如下： <ul style="list-style-type: none"> <li>当操作符为 = 或者 == 时，结果为 TRUE</li> <li>当操作符为 &gt;、&lt;、&gt;=、&lt;=、!= 或者 &lt;&gt; 时，结果为 FALSE</li> </ul> </li> <li>如果 v1 和 v2 中任意一个 NULL，而另一个非 NULL，则 v1 和 v2 通过上述任意一个运算符进行比较的结果都为 FALSE</li> </ol>
逻辑表达式	<p>操作符包括 &amp;&amp; 和   </p> <p>假设参与运算的为 v1 和 v2</p> <ol style="list-style-type: none"> <li>如果 v1 和 v2 都为 NULL，则 v1 和 v2 通过上述任意一个运算符进行运算的结果都为 FALSE</li> <li>如果 v1 和 v2 中任意一个 NULL，而另一个非 NULL，则 v1 和 v2 通过上述运算符进行运算的结果如下： <ul style="list-style-type: none"> <li>当操作符为 &amp;&amp; 时，结果为 FALSE</li> <li>当操作符为    时，结果以另一个非 NULL 的值的结果为准</li> </ul> </li> </ol>

## 空白值参与运算出现非期待结果的解决方案

当值为空白的字段参与计算时，可能会出现非期待的结果。目前有两种方案可提前对此情况进行处理。

1. 在销售易 PaaS 平台上进行设置，表中的 NULL 对应界面的设置为“空白字段认为是空白”项，如下图红框部分所示，即当勾选“空白字段认为是空白”时，按照上表中描述的处理方式进行运算。

新建字段

选择字段类型：

- 自动编号
- 关联关系
- 主子明细
- 引用字段
- 计算公式
- 电话
- 邮箱
- 网址

示例：

报价金额=单价\*数量\*折扣+运费

根据您定义的公式表达式派生其值的只读字段，任何来源字段有所更改，该计算型字段均将更新。

数字 计算公式

计算公式 =

插入字段 插入运算符 插入函数 语法检查

描述

参与计算的字段值为空时如何处理？

- 空白字段认为是0
- 空白字段认为是空白

小数点位数

0

2. 通过函数 `IsNull` 或者 `NullValue` 判断参与运算的字段是否为空白，然后进行处理。

