

CADMATIC Outfitting

培训手册

Customization Development- Basic

客户开发 – 基础

文档编号：CADMATIC-SH-TM601-00.00.00

目录

[版权说明 3](#_Toc494120271)

[修订记录 4](#_Toc494120272)

[1 概述 5](#_Toc494120273)

[2 开发工具 6](#_Toc494120274)

[3 基本术语 7](#_Toc494120275)

[3.1 通用术语 7](#_Toc494120276)

[3.1.1 源代码 7](#_Toc494120277)

[3.1.2 编程语言 7](#_Toc494120278)

[3.1.3 二进制代码 7](#_Toc494120279)

[3.1.4 可执行文件 7](#_Toc494120280)

[3.1.5 函数（Function）和子程序（Subroutine） 7](#_Toc494120281)

[3.1.6 赋值语句Assignment statement 7](#_Toc494120282)

[3.1.7 表达式Expression 7](#_Toc494120283)

[3.1.8 变量Variable 7](#_Toc494120284)

[3.1.9 操作符Operators 7](#_Toc494120285)

[3.1.10 操作数Operands 7](#_Toc494120286)

[3.1.11 位Bit 7](#_Toc494120287)

[3.1.12 字节Byte 7](#_Toc494120288)

[3.1.13 数据类型DataType 7](#_Toc494120289)

[3.2 CADMATIC脚本术语 7](#_Toc494120290)

[3.2.1 内置函数Intrinsic functions 7](#_Toc494120291)

[3.2.2 外部函数External functions 7](#_Toc494120292)

[3.2.3 脚本库函数Scriptlib functions 7](#_Toc494120293)

[4 CADMATIC的基本语法 8](#_Toc494120294)

[4.1 第一个脚本程序 Hello World 8](#_Toc494120295)

[4.2 数据类型 9](#_Toc494120296)

[4.3 变量的声明及初始化 10](#_Toc494120297)

[4.3.1 显示声明 10](#_Toc494120298)

[4.3.2 隐式声明 10](#_Toc494120299)

[4.4 函数 Functions 11](#_Toc494120300)

[4.4.1 内部函数 11](#_Toc494120301)

[4.4.2 外部函数 11](#_Toc494120302)

[4.4.3 本地函数 11](#_Toc494120303)

[4.4.4 标准函数 11](#_Toc494120304)

[4.5 包含（Include）文件 12](#_Toc494120305)

[4.6 语句(Statement) 12](#_Toc494120306)

[4.7 算术表达式 12](#_Toc494120307)

[4.8 赋值语句 12](#_Toc494120308)

[4.9 逻辑表达式 12](#_Toc494120309)

[4.10 控制结构 12](#_Toc494120310)

[4.10.1 循环 12](#_Toc494120311)

[4.10.2 分支 12](#_Toc494120312)

[4.11 运算符优先级 12](#_Toc494120313)

[4.12 内置函数 12](#_Toc494120314)

[4.12.1 字符串函数 12](#_Toc494120315)

[4.12.2 算术及三角函数 12](#_Toc494120316)

[4.12.3 其它函数 12](#_Toc494120317)

[5 编程规则（风格）指南 13](#_Toc494120318)

[5.1 说明 13](#_Toc494120319)

[5.2 命名规则 13](#_Toc494120320)

[5.2.1 变量 13](#_Toc494120321)

[5.2.2 常量 13](#_Toc494120322)

[5.2.3 函数 13](#_Toc494120323)

[6 数据类型转换 13](#_Toc494120324)

[7 脚本钩子（Hooks） 13](#_Toc494120325)

[8 查找模型 14](#_Toc494120326)

[8.1 通过名称查找 14](#_Toc494120327)

[8.2 通过模型位号查找 14](#_Toc494120328)

[8.3 通过模型ID查找 14](#_Toc494120329)

[8.4 通过点选查找 14](#_Toc494120330)

[8.5 通过Query语句查找 14](#_Toc494120331)

[9 抽取信息 14](#_Toc494120332)

[9.1 抽取模型信息 14](#_Toc494120333)

[9.2 抽取零件信息 14](#_Toc494120334)

[9.3 抽取尺寸表信息 14](#_Toc494120335)

[9.4 抽取GDL信息 14](#_Toc494120336)

[10 创建模型 14](#_Toc494120337)

[10.1 创建设备模型 14](#_Toc494120338)

[10.2 创建型材模型 14](#_Toc494120339)

[10.3 创建板材模型 14](#_Toc494120340)

[10.4 创建标准零件 14](#_Toc494120341)

[10.5 创建管子模型 15](#_Toc494120342)

[11 创建图形 15](#_Toc494120343)

[11.1 概述 15](#_Toc494120344)

[11.2 基本流程 15](#_Toc494120345)

[11.3 生成图纸 15](#_Toc494120346)

[11.4 生成视图 15](#_Toc494120347)

[11.5 尺寸标注 15](#_Toc494120348)

[11.6 标签 Label 15](#_Toc494120349)

[11.7 二维符号 2D symbol 15](#_Toc494120350)

[11.8 二维图形 15](#_Toc494120351)

[11.9 图框智能字符串 Text Box 15](#_Toc494120352)

版权说明

修订记录

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 修订日期 | 修订版本号 | 修订内容 | 修订人 | 审核 | 审定 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# 概述

# 开发工具

# 基本术语

## 通用术语

### 源代码

### 编程语言

### 二进制代码

### 可执行文件

### 函数（Function）和子程序（Subroutine）

### 赋值语句Assignment statement

### 表达式Expression

### 变量Variable

### 操作符Operators

### 操作数Operands

### 位Bit

### 字节Byte

### 数据类型DataType

## CADMATIC脚本术语

### 内置函数Intrinsic functions

### 外部函数External functions

### 脚本库函数Scriptlib functions

# CADMATIC的基本语法

## 第一个脚本程序 Hello World

#include include/dmutil.h

Main()

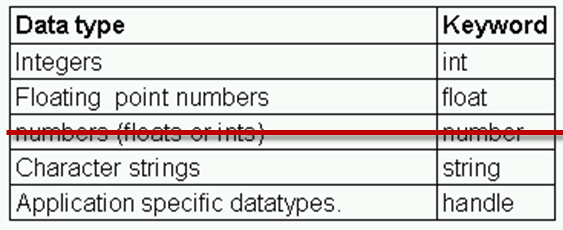
{

U\_MESSAGE(“Hello world”)

}

## 数据类型

CADMATIC脚本语言支持下列的数据类型：



## 变量的声明及初始化

任何一个变量都必须是属于某一种特定的数据类型，变量在使用前必须先声明。任何变量参与运算时必须已经具有确定的值（初始化），否则会引起错误。

### 显示声明

显示声明指先直接声明某个变量为某种数据类型的变量，然后再对这个变量进行初始化或者在声明的同时初始化这个变量。

1. 本地变量

格式：Type Name; 或者 Type Name = value;

例如：

int a; int a = 1;

float b; float b = 1.0;

string name; string name = “程序员”;

handle model;

1. 全局变量

格式：global Type Name; 或者 global Type Name = value;

例如：

global int a; global int a = 0;

### 隐式声明

直接初始化变量的值，根据赋值的类型来确定变量的数据类型，例如：

x = 0;

name = “Programer”;

## 函数 Functions

函数指的是用于完成某一特定功能的一组代码，这组代码可以被其它脚本程序重复使用。

每一个脚本程序至少必须包含一个main函数，这个函数是整个脚本程序的入口函数（脚本程序运行的起始点）。

### 内部函数

内部函数指脚本语言自带的函数，用户无需声明，可以直接使用。

比如： ITOASCII – 用于将整数转换成为字符串的函数。

### 外部函数

外部函数指系统提供的，可以供用户调用的函数。用户在自己的脚本程序中在使用这些函数之前，必须先声明要使用的函数，例如：

extern COS\_CHECK\_OUT;

系统提供的外部函数的声明已经全部包含在下面的目录：

…\include

用户要使用到某个外部函数时，只需要把包含对应的外部函数声明的头文件（\*.h）文件使用包含语句#include包含到自己的脚本文件中就可以了。

例如：

#include "include/dmutil.h"

### 本地函数

本地函数指在脚本文件自身中定义的函数。这些函数在脚本文件自身中可以直接调用，不需要声明。

### 标准函数

标准函数指由CADMATIC或者用户自己开发的脚本文件所定义的函数，用户要使用这些函数时，必须把定义这些函数的脚本文件使用包含语句#include包含到自己的脚本文件中。

例如：

#include "$Cadmatic\_Toolkit/EquipmentSeat/EquipmentSeat101.h"

## 包含（Include）文件

## 语句(Statement)

## 算术表达式

## 赋值语句

## 逻辑表达式

## 控制结构

### 循环

### 分支

## 运算符优先级

## 内置函数

### 字符串函数

### 算术及三角函数

### 其它函数

# 编程规则（风格）指南

## 说明

为了使脚本程序能够更容易读懂，方便程序代码的共享及维护（包括修订，升级等），编写程序时需要遵循一定的规则。

CADMATIC脚本语言的编程规则基本上与传统的C语言一致。

## 命名规则

脚本程序中存在着各种类型的标识符，为了让程序员更容易地区分各种标识符的类型及作用，对标识符的命名做如下的规定（注意字母的大小写是有区别的）：

### 变量

1. 本地变量

本地变量使用小写字母，为了能够更好理解，变量的名字中如果包含基本词语，可以用下划线将它们分开，比如：

part\_id (零件的ID号)

1. 全局变量

全局变量

### 常量

### 函数

# 数据类型转换

# 脚本钩子（Hooks）

# 查找模型

## 通过名称查找

PM\_FIND\_BY\_NAME

## 通过模型位号查找

PM\_FIND\_BY\_POSTION\_ID

## 通过模型ID查找

PM\_FIND\_BY\_OBJID

## 通过点选查找

PM\_PICK\_OBJECT

## 通过Query语句查找

query = PM\_INIT\_QUERY("EQUIPMENT");

equipments = PM\_EXECUTE\_QUERY(query);

# 抽取信息

## 抽取模型信息

PM\_GET\_OBJDATA

## 抽取零件信息

DM\_PARTID\_DATA

## 抽取尺寸表信息

## 抽取GDL信息

## 抽取连接点信息

# 创建模型

## 创建设备模型

## 创建型材模型

## 创建板材模型

## 创建标准零件

## 创建管子模型

# 创建图形

## 概述

## 基本流程

## 生成图纸

## 生成视图

## 尺寸标注

## 标签 Label

## 二维符号 2D symbol

## 二维图形

## 表格 Table

## 智能字符串 Text Box