

Ruby 类案例

下面将创建一个名为 Customer 的 Ruby 类，声明两个方法：

- `display_details`：该方法用于显示客户的详细信息。
- `total_no_of_customers`：该方法用于显示在系统中创建的客户总数量。

实例

```
#!/usr/bin/ruby
class Customer
  @@no_of_customers=0
  def initialize(id, name, addr)
    @cust_id=id
    @cust_name=name
    @cust_addr=addr
  end
  def display_details()
    puts "Customer id #@cust_id"
    puts "Customer name #@cust_name"
    puts "Customer address #@cust_addr"
  end
  def total_no_of_customers()
    @@no_of_customers += 1
    puts "Total number of customers: #@no_of_customers"
  end
end
```

`display_details` 方法包含了三个 `puts` 语句，显示了客户 ID、客户名字和客户地址。其中，`puts` 语句：

```
puts "Customer id #@cust_id"
```

将在一个单行上显示文本 Customer id 和变量 `@cust_id` 的值。

当您想要在一个单行上显示实例变量的文本和值时，您需要在 `puts` 语句的变量名前面放置符号（#）。文本和带有符号（#）的实例变量应使用双引号标记。

第二个方法，`total_no_of_customers`，包含了类变量 `@@no_of_customers`。表达式 `@@no_of_customers+=1` 在每次调用方法 `total_no_of_customers` 时，把变量 `no_of_customers` 加 1。通过这种方式，您将得到类变量中的客户总数量。

现在创建两个客户，如下所示：

```
cust1=Customer.new("1", "John", "Wisdom Apartments, Ludhiya")
cust2=Customer.new("2", "Poul", "New Empire road, Khandala")
```

在这里，我们创建了 Customer 类的两个对象，`cust1` 和 `cust2`，并向 `new` 方法传递必要的参数。当 `initialize` 方法被调用时，对象的必要属性被初始化。

一旦对象被创建，您需要使用两个对象来调用类的方法。如果您想要调用方法或任何数据成员，您可以编写代码，如下所示：

```
cust1.display_details()
cust1.total_no_of_customers()
```

对象名称后总是跟着一个点号，接着是方法名称或数据成员。我们已经看到如何使用 cust1 对象调用两个方法。使用 cust2 对象，您也可以调用两个方法，如下所示：

```
cust2.display_details()
cust2.total_no_of_customers()
```

保存并执行代码

现在，把所有的源代码放在 main.rb 文件中，如下所示：

实例

```
#!/usr/bin/ruby
class Customer
  @@no_of_customers=0
  def initialize(id, name, addr)
    @cust_id=id
    @cust_name=name
    @cust_addr=addr
  end
  def display_details()
    puts "Customer id #@cust_id"
    puts "Customer name #@cust_name"
    puts "Customer address #@cust_addr"
  end
  def total_no_of_customers()
    @@no_of_customers += 1
    puts "Total number of customers: #@no_of_customers"
  end
end
# 创建对象
cust1=Customer.new("1", "John", "Wisdom Apartments, Ludhiya")
cust2=Customer.new("2", "Poul", "New Empire road, Khandala")
# 调用方法
cust1.display_details()
cust1.total_no_of_customers()
cust2.display_details()
cust2.total_no_of_customers()
```

尝试一下 »

接着，运行程序，如下所示：

```
$ ruby main.rb
```

这将产生以下结果：

```
Customer id 1
Customer name John
Customer address Wisdom Apartments, Ludhiya
Total number of customers: 1
Customer id 2
Customer name Poul
Customer address New Empire road, Khandala
Total number of customers: 2
```

[← Ruby 类和对象](#)[Ruby 变量 →](#)[✎ 点我分享笔记](#)