# PALChain

下一个我要向你展示的是PAL Math Chain，这其实是使用了一个不同的大型语言模型。

我们基本上要做的是，当我们遇到某种数字问题时，我们将使用它。这是 Langchain 文档中的一个例子，Jan有宠物的数量是三倍。这是一个数学方程问题了。

#### 为什么用 PALChain ？

我们基本上在这里Prompts（提示）模型这样做的方式是将这个文字陈述转化为一个小型的Python函数，然后计算数学问题，而不是仅仅依靠语言模型猜测。

来看一个有趣的例子。这是非常简单的数学问题，食堂有23个苹果，如果他们用了20个来吃午饭，又买了6个，那么他们现在还剩下多少个苹果。

问题在于，如果你使用的是大型语言模型，它们可能会得到正确的答案，但如果你使用的是一些较小的模型，甚至只是像T5模型一样，大多数T5模型都会得到这样的错误答案，而不是依赖其中一个模型来做这个，我们可以使用这个方法，它基本上是获取这些数据并进行重写。

from langchain.chains import PALChain  
pal\_chain = PALChain.from\_math\_prompt(llm, verbose=True)  
  
question = "Jan has three times the number of pets as Marcia. Marcia has two more pets than Cindy. If Cindy has four pets, how many total pets do the three have?"

pal\_chain.run(question)

你可以看到它写了一个Python函数，它使用了文档字符串将我们之前的内容放在这里，我们从苹果的初始开始，所以它只是将这些变量赋值。然后苹果剩下的数量等于初始苹果减去使用的苹果加上购买的苹果，它确实给了我们准确的结果。

> Entering new PALChain chain...  
def solution():  
 """The cafeteria had 23 apples. If they used 20 for lunch and bought 6 more, how many apples do they have?"""  
 apples\_initial = 23  
 apples\_used = 20  
 apples\_bought = 6  
 apples\_left = apples\_initial - apples\_used + apples\_bought  
 result = apples\_left  
 return result  
  
 > Finished chain.  
9

然后我们可以将输出带入另一个大型语言模型中，然后以对话的方式重新表达它，这样它可以告诉你苹果剩下的数量是多少，或者我们可以直接从这个模块中获取输出。