

每日頭條

首頁 | 健康 | 娛樂 | 時尚 | 遊戲 | 3C | 親子 | 文化 | 歷史 | 動漫 | 星座 | 健身 | 家居 | 情感

動態語言和靜態語言的區別

找不到想看的？搜尋看看！

2018-07-12 由 程序員小新人學習 發表于程式開發

編譯型語言和解釋型語言

1、編譯型語言

需通過編譯器（**compiler**）將原始碼編譯成機器碼，之後才能執行的語言。一般需經過編譯（**compile**）、連結（**linker**）這兩個步驟。

編譯是把原始碼編譯成機器碼，

連結是把各個模塊的機器碼和依賴庫串連起來生成可執行文件。

優點：編譯器一般會有預編譯的過程對代碼進行優化。因為編譯只做一次，運行時不需要編譯，所以編譯型語言的程序執行效率高。可以脫離語言環境獨立運行。

缺點：編譯之後如果需要修改就需要整個模塊重新編譯。編譯的時候根據對應的運行環境生成機器碼，不同的作業系統之間移植就會有問題，需要根據運行的作業系統環境編譯不同的可執行文件。

代表語言：C、C++、Pascal、Object-C以及最近很火的蘋果新語言swift

2、解釋型語言

解釋性語言的程序不需要編譯，相比編譯型語言省了道工序，解釋性語言在運行程度的時候才逐行翻譯。

優點：有良好的平台兼容性，在任何環境中都可以運行，前提是安裝了解釋器（虛擬機）。靈活，修改代碼的時候直接修改就可以，可以快速部署，不用停機維護。

缺點：每次運行的時候都要解釋一遍，性能上不如編譯型語言。

代表語言：JavaScript、Python、Erlang、PHP、Perl、Ruby

3、混合型語言

既然編譯型和解釋型各有缺點就會有人想到把兩種類型整合起來，取其精華去其糟粕。就出現了半編譯型語言。比如C#，C#在編譯的時候不是直接編譯成機器碼而是中間碼，.NET平台提供了中間語言運行庫運行中間碼，中間語言運行庫類似於Java虛擬機。.net在編譯成IL代碼後，保存在dll中，首次運行時由JIT在編譯成機器碼緩存在內存中，下次直接執行（博友回復指出）。我個人認

為拋開一切的偏見**C#**是這個星球上最好的程式語言。可惜微軟的政策限制了**C#**的推廣。

Java先生成字節碼再在**Java**虛擬機中解釋執行。嚴格來說混合型語言屬於解釋型語言。**C#**更接近編譯型語言。

動態語言和靜態語言

1、動態語言

是一類在運行時可以改變其結構的語言：例如新的函數、對象、甚至代碼可以被引進，已有的函數可以被刪除或是其他結構上的變化。通俗點說就是在運行時代碼可以根據某些條件改變自身結構。

主要動態語言：Object-C、C#、JavaScript、PHP、Python、Erlang。

2、靜態語言

與動態語言相對應的，運行時結構不可變的語言就是靜態語言。如Java、C、C++。

3、注意：

很多人認為解釋型語言都是動態語言，這個觀點是錯的！**Java**是解釋型語言但是不是動態語言，**Java**不能在運行的時候改變自己結構。反之成立嗎？動態語言都是解釋型語言。也是錯的！**Object-C**是編譯型語言，但是他是動態語言。得益於特有的run time機制（準確說run time不是語法特性是運行時環境，這裡不展開）OC代碼是可以在運行的時候插入、替換方法的。

C#也是動態語言，通過**C#**的反射機制可以動態的插入一段代碼執行。所以我說**C#**是這個星球最好的程式語言。

有三個名詞容易混淆：

Dynamic Programming Language (動態語言或動態程式語言)

Dynamically Typed Language (動態類型語言)

Statically Typed Language (靜態類型語言)

動態類型語言和靜態類型語言

1、動態類型語言

很多網上資料把動態類型語言和動態語言混為一談，簡直是誤人子弟。動態類型語言和動態語言是完全不同的兩個概念。**動態類型語言**：是指在運行期間才去做數據類型檢查的語言，說的是數據類型，

動態語言：說的是運行是改變結構，說的是代碼結構。

動態類型語言的數據類型不是在編譯階段決定的，而是把類型綁定延後到了運行階段。

主要語言：Python、Ruby、Erlang、JavaScript、swift、PHP、Perl。

2、靜態類型語言

靜態語言的數據類型是在編譯期間（或運行之前）確定的，編寫代碼的時候要明確確定變量的數據類型。

主要語言：C、C++、C#、Java、Object-C。

3、注意：

相當一部分程式設計師，也包括曾經的我，認為解釋型語言都是動態類型語言，編譯型語言都是靜態類型語言。這個也是錯的。**swift**是編譯型語言但是它也是動態類型語言。**C#**和**Java**是解釋型語言也是靜態類型語言。



 點我分享到Facebook

相關文章



編譯型語言、解釋型語言、靜態類型語言、動態類型語言概念與區別

2018-03-11

編譯型語言和解釋型語言1、編譯型語言需通過編譯器將原始碼編譯成機器碼，之後才能執行的語言。一般需經過編譯、連結這兩個步驟。



Kotlin系列（一）：Kotlin初探，揭開Kotlin的神秘面紗

2017-10-23

一、前言：1、Kotlin 是什麼？



《Inside C#》筆記（一）.NET平台

2017-06-10

C# 基於.NET運行時，所以有必要首先對.NET以及C#與.NET平台的關係有一定的了解。一 .NET平台.NET背後的基本思想是將原本獨立工作的設備、網絡服務整合在一個統一的平台，從而可以為用戶提供豐富全面的解決方案。平時提到.Net一般默認指的是。

微軟.Net CLR，速度和可伸縮性均會提高

2017-11-20

微軟的公共語言運行時正在進行一系列的改造，公司宣布這個改造計劃將使CLR更加高效、可擴展性更強。微軟首席軟體工程師BenWatson表示，目前.Net的最大問題是運行時自身可伸縮性的固有限



制，CLR正在超越設計初衷。



Python簡介

2017-04-25

學習交流群：[581935586](#) Python簡介介紹python之前，先來說一下編譯型語言和解釋型語言，面向對象和面向過程的區別：計算機高級語言按照程序的執行方式可以分為編譯型語言和解釋型語言。



零基礎怎麼學習Python?新手入門Python前先看

2018-09-11

python是一種面向對象的，解釋型的計算機語言，它的特點是語法簡介，優雅，簡單易學。1989年誕生，Guido開發。



體驗.NET Core 2.0

2017-10-26

上圖描述了 .NET Core的系統構成，最上層是應用層，是開發基於UI應用的框架集，包括了ASP.NET Core(用於創建web app)，和UWP(用於創建Windows10 app)。中間層是公共庫(CoreFX),實現了.

程式語言相關概念

2016-08-23

編譯、解釋型語言：編譯型語言需要首先由編譯器將原始碼編譯為機器代碼，再由機器執行，即運行前有一個編譯過程，C++是典型的編譯語言，GCC是C++的最流行的編譯器；解釋型語言不需要編譯，運行時直接由解釋器解釋執行，典型代表如shell、Python；也有語言是這兩種類型的混合，即既



計算機語言不同用法的五大分類及優缺點

2018-08-20

計算機語言種類繁多，根據不同用法又有不同的分類，那現在有哪些分類方式方式呢，下面我們來看下。1.高級語言和低級語言低級語言：彙編語言高級語言：c、c++、java、python、PHP、c#。



困擾多年的問題，Python到底是語言，還是工具，哈佛教授完美詮釋

2018-08-03

Python是出類拔萃的然而，這是一句非常模稜兩可的話。上面提到的那些技術經常被提起和引用。如果你對機器碼，虛擬機之類的很熟

了，你可以跳過開頭，直接從"即時編譯:PyPy和它的未來"這部分開始看起。



很多人想學習Python，但首先得搞清楚這幾個問題

2018-10-23

Python的概念Python可以應用於眾多領域，如：數據分析、組件集成、網絡服務、圖像處理、數值計算和科學計算等眾多領域。

憑啥Java的運行環境稱虛擬機，Python的只能稱解釋器

2015-03-06



看到Stackoverflow上有個問題在討論Java和Python的對比，其中就有人問答為啥Java的運行環境被稱之為JVM，而Python的只能叫做Interpreter。（題圖來自：gfi.com）這個問題估計想過的人不多，先找維基百科看一下虛擬機的定義。

帶你全新認識，為什麼世界排行第一的程式語言是java？

2018-08-15

根據Tiobe官方發布，排行前三的分別是java、C、C++，java以14.9%市場占有率遙遙領先其他語言。

10款小眾的程式語言

2014-06-11

1.Go是基於Inferno作業系統所開發的。Go語言於2009年11月正式宣布推出，成為開放原始碼項目，並在Linux及Mac OS X平台上進行了實現，後追加Windows系統下的實現。Go是Google開發的一種編譯型、並髮型程式語言，並具有垃圾回收功能。2.



C一定比C++性能差？當然不！破除迷信，從我做起！

2017-03-05

幾乎所有的程式設計師在初學編程之時，都被灌輸過「託管語言（Java、C#）性能比非託管語言（C、C++）差好多」的迷信教條。



java9新特性-21-java的動態編譯器

2018-01-07

1. 官方Feature243: Java-Level JVM Compiler Interface295: Ahead-of-Time Compilation2.

神級程式設計師教你（沒有編程經驗）的小白也是能快速進入IT行業的！

2017-11-15

此篇是針對沒有計算機基礎和沒有編程基礎的小夥伴整理出的資料乾貨！如果你是大牛的話也可以看下，幫忙指正錯誤哦！小編推薦大家

加一下這個群：330637182 這個群里好幾千人了！大家遇到啥問題都會在裡面交流！而且免費分享零基礎入門資料 web開發 爬蟲資料一整套！



掌握多少門程式語言才能成為優秀程式設計師？

2018-08-01

甚至架構師、技術總監等，顯然只精通一種程式語言是不夠的，還應該在常見領域學會幾門程式語言，正如我們要成為高級人才不僅要會

中文還要會英文，甚至還要會德文、俄文一樣。



1.python第一步

2017-12-03

pythonpython的優點：python哲學就是簡單優雅，儘量寫容易看明白的代碼，儘量寫少的代碼。適合開發的應用：網絡應用：包括網站，後台應用其次是許多日常的小工具：如系統管理員需要的腳本等，另外

就是把其他語言開發的程序包裝起來方便使用。缺點：1.

程式語言的發展趨勢及未來方向（5）：元編程

2017-04-12

這是Anders Hejlsberg(不用介紹這是誰了吧)在比利時TechDays 2010所做的開場演講。由於最近我在博客上關於語言的討論比較多，出於應景，也打算將Anders的演講完整地聽寫出來。在上一部分中，Anders談及了他眼中程式語言的另一個發展趨勢：動態性。



從C到C++，他們有什麼區別？能做什麼？與其他語言又有什麼不同？

2017-09-12

1、從C到C++計算機誕生初期，用機器語言或彙編語言編寫程序；第一種高級語言FORTRAN誕生於1954年；BASIC語言(1964)是由FORTRAN語言的簡化而成的是為初學者設計的小型高級語言；C語言是1972年由美國貝爾實驗室的D.M.Ritchie 研製成功的。



為什麼說Crystal是2018年最有前景的程式語言

2018-07-04

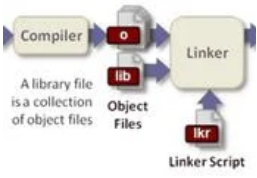
在多年前很多開發人員集思廣議「我們如何使用Ruby類似的語法達到C的執行效率，通用性，以及像C/C++一樣可以編譯成任何平台的本地二進位文件且具有高級語言的特性，比如豐富的標準庫，fiber支持等等」，若干年之後的今天這個夢想終於在Csystal程式語言上得到實現。



對比篇：如何理解JAVA和C語言的跨平台？

2018-10-18

一、C/C++語言的編譯和執行過程1、編譯過程包括預處理、編譯、彙編和連結四個階段：1）預處理：偽指令轉換和特殊符號處理，生成最終需要編譯部分的代碼文件；2）編譯：通過作業系統中安裝的編譯器。



簡述預處理，編譯，彙編，連結都是什麼

2017-12-27

編譯器就是將"一種語言"翻譯為"另一種語言"的程序。* 預處理：預處理中會展開以#起始的行，試圖解釋為預處理指令，其中ISO C/C++要求支持的包括#if/#ifdef/#ifndef/#else/#elif/#endif、#define、#include、#line、#



為什麼Python這麼慢？

2018-08-14

大數據文摘出品編譯：jojo、Hope、張秋玥、CoolBoyPython語言近年來人氣爆棚。它廣泛應用於網絡開發運營，數據科學，網絡開發，以及網絡安全問題中。

什麼是C#語言以及語言結構

2016-08-28

C#是微軟公司發布的一種面向對象的、運行於.NET Framework之上的高級程序設計語言。並定於在微軟職業開發者論壇(PDC)上登台亮相。C#是微軟公司研究員Anders Hejlsberg的最新成果。



程序猿逆天改命之Python大法(1)：Python介紹

2018-09-22

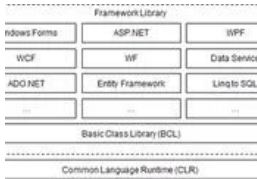
一：Python的出生與應用python的創始人為吉多·范羅蘇姆。1989年的聖誕節期間，吉多·范羅蘇姆為了在阿姆斯特丹打發時間，決心開發一個新的腳本解釋程序。



扣丁學堂教你Python的基礎必備（一）

2018-08-06

2008年發布了Python3.0，但由於此版本並不完全兼容之前的廣泛使用的2.4版本，為解決這個問題。



.NET Core跨平台的奧秘：歷史的枷鎖

2017-11-06

微軟推出的第一個版本的.NET Framework是一個面向Windows桌面和伺服器的基礎框架，在此之後，為此微軟根據設備自身的需求對.NET Framework進行裁剪，不斷推出了針對具體設備類型的.NET Framework版本以實現針對移動、平板和嵌入式設備提供支持。



第2天 拓展：JavaScript 與 Java 的關係

2018-01-02

對於這個問題，有這樣一種說法：Java和JavaScript是雷鋒與雷峰塔的關係。後來選擇JavaScript作為正式名字，其中緣由大概是想讓人有出身名門的感覺。