

Haskell

✎ 修改

Haskell 的 Typeclass 怎么理解？✎ 修改

中文对应哪个关键词？然后怎么理解这段英文介绍，我感觉不好懂（Java 不懂的情况下）。A typeclass is a sort of interf...显示全部 ▾

关注问题

✎ 写回答

👤 邀请回答

💬 添加评论

🚩 分享

★ 邀请回答

🚩 举报

...

关注者153被浏览10,362

他们也关注了该问题



查看全部 15 个回答

Felis sapiens ⭐

函数式编程、编程语言、编程 话题的优秀回答者

彭飞、题叶、开源哥、刘鑫等 35 人赞同了该回答

补充一下其他答案里好像漏掉的一点：type class其实更严谨的说法叫constructor class。。以下答案中“type class”均指代constructor class。

以最简单的show函数为例，show :: Show a => a -> String，只要某个值type是Show类的instance，就可以调用show函数。另外一个例子是fmap :: Functor f => (a -> b) -> f a -> f b。注意到了什么？f是Functor类的instance，但是f并不是一个类型；f a和f b才是类型。换句话说，能够成为某个类的instance的东西，并不一定是type，还有可能是type constructor。。

如果把type看成另一种value的话，那么type constructor其实不过就是type function而已。。然后，比如a -> b就可以表述为(->) a b，而这个(->)就是一个type function，而type function也支持自动的currying。

怎样区分一个东西是type和type function？你需要一个叫做kind的概念。。规定Int，Bool之类的类型的kind为*，然后像Functor/Monad等类的成员的kind为* -> *，而(->)的kind为* -> * -> *。。然后kind为*的constructor才能够描述值类型，而其他kind的东西都可以看成type function。。然后，定义一个type class的时候，他的instance可以是一个具体的type（也就是说kind为*），也可以是其他kind的constructor。。

然后在实现编译器的时候，type inference之前还有一步简单的kind inference。。确保不会出现type X = Int Int之类的kind错误。

更强的拓展，dependent type什么的，还没学就不提了。。

ref:
[A system of constructor classes: overloading and implicit higher-order polymorphism](#)
[haskell.org/onlinerepor...](#)

编辑于 2015-02-03

▲ 赞同 35 ▾

💬 6 条评论


🚩 分享

★ 收藏

❤ 感谢

...

更多回答

vczh

专业造轮子，拉黑抢前排。 [gaclib.net](#)

26 人赞同了该回答

如果从工程的角度来看，Haskell的type class解决了以下几类问题

- 1、类型推导如何兼容重载函数
- 2、定义OO语言所缺失的concept /*mapping*/（如C#的IComparable<T>和Haskell的Sort，事实上任何时候C#的IComparable<T>都不应该是一个接口，因为每个类型只会实现一次，所以他应



关于作者

Felis sapiens

⭐ 函数式编程、编程语言、编程 话题的优秀回答者

👤 电影旅行敲代码、Antokha Yuuki、暮无井见铃也关注了她

回答	文章	关注者
624	40	14,871
已关注	发私信	

被收藏 9 次

Programming Languages 3,642 人关注
彭飞 创建

长知识
刘典 创建

Haskell
afraid 创建

FP
Dainslef 创建



该是concept，对于任何T应该是静态的，然而OO的语言就是对这些东西有着微妙的敌意）。C++的模板的一些奇葩的功能替代了concept，然而一旦程序写错了，错误信息特别糟糕，所以他们还正在试图发明一次concept。

3、算法的扩展。之前谁的一篇文章讲到了类型的两种扩展。第一种扩展是针对函数扩展，你有一些固定的类型，你可以不断的写新的函数来在运行的时候动态重载他们，就像Visitor模式干的一样。第二种是针对类型的扩展，这个就是大家熟悉的虚函数，你先定义好一系列的函数，于是你就可以无限的扩展新的类型来实现这些函数。Haskell干的就倾向于第一种，没有办法做第二种，因为他的data声明是封闭的，而type class的instance是开放的。其中固定的类型指的就是data的各种构造

[展开阅读全文](#)

赞同 26 2 条评论 分享 收藏 感谢 ...

 **祖与占** 
函数式编程 话题的优秀回答者
6 人赞同了该回答

赞同 6 2 条评论 分享 收藏 感谢 ...

[查看全部 15 个回答](#)

函数式语言 0 人关注 
[ykcopevol](#) 创建

相关问题

- [如何理解Haskell中的Arrow及其用途?](#) 2 个回答
- [为什么业界很少使用 Haskell?](#) 5 个回答
- [Haskell中如何代价最小的操作（插入、删除）一棵树?](#) 4 个回答
- [有没有比较实用、成熟的 Haskell 应用?](#) 8 个回答
- [学过Haskell是一种怎样的体验?](#) 22 个回答

相关推荐



淼懂物理学：理解世界的极简指南
共 31 节课 [▶ 试听](#)



数学妙啊！妙！
张英锋 等
289,062 人读过 [阅读](#)

知乎

超级会员

[戳此进入>>](#)

3000+ 场好课随意听
1000+ 本好书任意读

广告

刘看山 · 知乎指南 · 知乎协议 · 隐私政策

应用 · 工作 · 申请开通知乎机构号

侵权举报 · 网上有害信息举报专区

违法和不良信息举报：010-82716601

儿童色情信息举报专区

电信与服务业务经营许可证

网络文化经营许可证

联系我们 © 2018 知乎