20210313-周报总结

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **小论文进展：** | **🞎阅读文献** | **🞎做实验** | **🞎撰写中** | **🗹其它** |
| **一、本周工作**   1. Python语法最后部分，模块、类的继承、装饰器看完了。在leetcode上刷刷题，继续熟悉语法锻炼思维。 2. 配置了anaconda的源，在pycharm中配置了pytorch的包。 3. 学习pytorch框架，开始看《动手学深度学习》，到第二章基础知识。 | | | | |
| 1. **下周计划** 2. Python继续leetcode刷题熟悉语法。 3. 暂时放弃爬虫的学习。 4. 《动手学深度学习》第三章。 | | | | |
| 1. **本周工作摘要**  内嵌函数和闭包 Global 关键字可以在局部中修改全局变量    内嵌函数 函数内部可以嵌套函数    **闭包**        关键字nonlocal 声明变量不是局部变量，就可以获取外部值。    Leetcode刷题      Python中的链表操作和c类似  无法直接通过列表的方式访问  python是动态语言，可以直接把对象赋值给新的变量。  用对象充当指针使用。 匿名函数 Lambda 关键字      **Bif+lambda**  Fliter() 过滤  map()     递归   斐波拉契数列    汉诺塔   字典 数学中映射逻辑    **Dict()**创建字典 工厂函数        **Fromkeys** 《动手学深度学习》 **第一章**  略  **第二章**  在PyTorch中，torch.Tensor是存储和变换数据的主要工具。如果你之前用过NumPy，你会发现Tensor和NumPy的多维数组非常类似。然而，Tensor提供GPU计算和自动求梯度等更多功能，这些使Tensor更加适合深度学习。  **"tensor"这个单词一般可译作“张量”，张量可以看作是一个多维数组。标量可以看作是0维张量，向量可以看作1维张量，矩阵可以看作是二维张量。**  import torch  x = torch.empty(5, 3)  print(x)  tensor([[ 0.0000e+00, 1.5846e+29, 0.0000e+00],  [ 1.5846e+29, 5.6052e-45, 0.0000e+00],  [ 0.0000e+00, 0.0000e+00, 0.0000e+00],  [ 0.0000e+00, 0.0000e+00, 0.0000e+00],  [ 0.0000e+00, 1.5846e+29, -2.4336e+02]])  我们可以通过shape或者size()来获取Tensor的形状:  print(x.size())  print(x.shape)Copy to clipboardErrorCopied  输出：  torch.Size([5, 3])  torch.Size([5, 3]) | | | | |