Write a generic Python3 programme/script, that doesn't use BioPython that IS NOT just a (series of) Linux/bash scripts called from Python, that will allow the user to define a dataset that is of interest to them, which will then be processed to produce the required outputs.【编写一个通用的Python3程序/脚本，它不使用BioPython，它不仅仅是从Python中调用的（一系列）Linux\/bash脚本，它将允许用户定义他们感兴趣的数据集，然后对其进行处理以产生所需的输出。】

I have suggested a "test set" for you that has a fairly small number of sequences (pyruvate dehydrogenase from ascomycete fungi (taxonID 4890)); please use this "test set" for the examples that you show in the help manual.

To summarise, your code will need to be generic: 【我为您建议了一个“测试集”，它有相当少的序列（子囊菌属真菌的丙酮酸脱氢酶（分类号4890））；请将此“测试集”用于帮助手册中显示的示例。

总之，您的代码需要通用：】

1. the user of your code will specify the protein family, and the taxonomic group, and then your code will need to obtain the relevant protein sequence data, and perform all subsequent analyses and outputs in the user's space on the server. Do NOT set it up so it will only work in your workspace! As all files/databases/outputs will be made in the users homespace on the server, the user's allowable starting sequence set probably shouldn't have more than 1,000 sequences (note, this is just a guideline, not a hard limit!) How useful the programme will be might depend a bit on how many species are represented in the dataset chosen by the user (i.e. are the sequences all from one species, or are there many different species?), so it would probably make sense to tell the user, and give them the option to continue or not continue with the current dataset? There are almost certainly other checks that could/should be done at this stage before continuing to the main processing stages【代码的用户将指定蛋白质家族和分类组，然后您的代码将需要获得相关的蛋白质序列数据，并在服务器上的用户空间中执行所有后续分析和输出。不要设置它，这样它只能在您的工作空间中工作！由于所有文件\/数据库\/输出都将在服务器上的用户主空间中进行，用户允许的起始序列集可能不应超过1000个序列（注意，这只是一个指导原则，而不是一个硬限制！）程序的有用程度可能取决于用户选择的数据集中表示了多少物种（即序列都来自一个物种，还是有很多不同的物种？），因此，告诉用户并让他们选择继续或不继续当前数据集可能是有意义的？在继续进行主要处理阶段之前，几乎可以肯定在这个阶段可以/应该进行其他检查】
2. to determine, and plot, the level of conservation between the protein sequences. Here, we are wanting to establish the degree of similarity within the sequence set chosen. The output should go to screen and be saved as a file output. Think carefully about how many sequences you might want to use for the conservation analysis and/or whether to limit the number of sequences used for the conservation analysis and plotting to some number? If you think they should, working out which ones to keep could be done in several different ways, some a lot easier than others: the method of selection choice is up to you, should you go down this route. 【以确定和绘制蛋白质序列之间的保守性水平。在这里，我们希望确定所选序列集的相似程度。输出应进入屏幕并保存为文件输出。仔细考虑您可能要使用多少序列进行保守性分析和/或是否将用于保守性研究和绘图的序列数量限制在某个数量？如果你认为他们应该这样做，那么可以用几种不同的方法来确定要保留哪些，有些方法比其他方法容易得多：选择的方法取决于你，如果你走这条路的话。】

3. to scan protein sequence(s) of interest with motifs from the PROSITE database, to determine whether any known motifs (domains) are associated with this subset of sequences: were there any, and if so, what were their names? Please note that the relevant databases that EMBOSS might need to do this are already installed. 【用PROSITE数据库中的基序扫描感兴趣的蛋白质序列，以确定是否有任何已知的基序（结构域）与该序列子集相关：是否有，如果有，它们的名称是什么？请注意，EMBOSS可能需要的相关数据库已经安装。】

4. a "wildcard" option for you: to do any other appropriate EMBOSS (or other) analysis that you think might add relevant biological information to the outputs; this does not mean just running some other EMBOSS programme with default parameters with no further processing of the outputs! I am leaving it up to you to choose what might constitute "relevant biolgical information" that your programme will provide to the user: you are training to be a bioinformatician, so you should be able to decide, and justify, your choice! 【EMBOSS（或其他）分析，您认为这些分析可能会将相关生物信息添加到输出中；这并不意味着只使用默认参数运行其他EMBOSS程序，而不进一步处理输出！我将由您选择您的程序将向用户提供的“相关生物信息”：您正在接受生物信息学家的培训，因此您应该能够决定并证明您的选择！】

Some of the programs you might (or might not) need:

esearch , efetch , and others for searching and retrieving from any of the NCBI databases from the edirect package

clustalo for clustering sequences etc

makeblastdb , blastn , blastx , blastp for doing BLAST analyses etc

plotcon and many others from EMBOSS ; use -help -verbose to get more info, or check out the link given above【您可能需要（或可能不需要）的一些程序：esearch、efetch和其他程序，用于从edirect包clustalo中的任何NCBI数据库中搜索和检索聚类序列等，makeblastdb、blastn、blastx、blastp用于进行BLAST分析等，plotcon和EMBOSS中的许多其他程序；使用-help-verbose获取更多信息，或查看上面给出的链接】

When putting the programme together, you should consider the following:

lots of comment lines in the code

ALWAYS test the code with lots of print statements, you can comment them out later

ALWAYS test the code with something that you know will work the user may be clever, but they can't read your mind: be explicit in telling them what they can and can't do, preferably at the time they are doing it!

have lots of "error traps" in your code (e.g. when the wrong thing is input, or there is no output, or ...)【当把程序放在一起时，你应该考虑以下几点：代码中的许多注释行总是用大量的打印语句来测试代码，你可以稍后将它们注释掉。总是用你知道会起作用的东西来测试代码，最好是在他们这样做的时候！在你的代码中有很多“错误陷阱”（例如，当输入了错误的东西，或者没有输出，或者…）】

Write a single "help manual" for your programme, which has two sections;

1. a section aimed at an "ordinary user": it contains basic instructions for running the program, and interpreting the output; it is aimed more at biologists who may not know much about Python3 coding, they just want the outputs! The user section of manual doesn't need to be huge. Remember that this is for a non-programmer to read, so you don't have to go into details of what variables you're using, etc.. It just has to describe briefly how the programme works, how to use it, and, perhaps, how NOT to use it!? You shouldn't need more than 5 pages, and probably fewer would be fine. Please don't reproduce all your code in this: I can see that when I clone from GitHub. You should use the outcomes from the test set (pyruvate dehydrogenase from ascomycete fungi (taxonID 4890)) as example inputs and outputs in this manual.
2. a section aimed at a competent Python3 code-writer. This is usually called a "maintenance manual" , which explains/shows how the different parts of the programme fit together. Feel free to use words like variable, function, iteration, dictionary, error trap etc. Once again, please dont reproduce all your code in this. Flow diagrams work very well here. You shouldn't need more than 5 pages, and probably fewer would be fine.【为您的计划编写一份“帮助手册”，其中有两个部分；1.针对“普通用户”的部分：它包含运行程序和解释输出的基本指令；它更多地针对那些可能不太了解Python3编码的生物学家，他们只是想要输出！手册的用户部分不需要很大。请记住，这是供非程序员阅读的，因此您不必详细说明您正在使用的变量等。它只需简单描述程序的工作原理、如何使用它，以及如何不使用它！？你不应该需要超过5页，可能更少就可以了。请不要在这里复制您的所有代码：当我从GitHub克隆时，我可以看到这一点。您应该使用测试集的结果（来自子囊真菌的丙酮酸脱氢酶（分类号4890））作为本手册中的输入和输出示例。2.针对合格的Python3代码编写者的部分。这通常被称为“维护手册”，它解释/显示了程序的不同部分如何配合在一起。请随意使用变量、函数、迭代、字典、错误陷阱等词。再次，请不要在这里重复您的所有代码。流程图在这里工作得很好。你不应该需要超过5页，可能更少就可以了。】

Marks will be awarded for : 1. The user "experience" Utility: does your programme present a seemingly useful tool for biologists? Usability: how easy is your programme to use? Interface: does your programme present a consistent interface to the user? 2. The help manual Readability of documentation: are your manual sections easy to read and understand? General description: is it easy for the user (of any level of competence) to understand what the programe does? 3. The code Does it use Python3 in preference to bash/awk? Does it use many of the Python3 features we have learnt? Correct: does the programme produce the correct output when run? Robust to error: does the programme check for valid input and if not good, fail gracefully with a useful error message? Well-structured: is code divided into logical functions/units, where that is possible? Well-documented: do the comments help a programmer to understand/maintain your code? Concise, well formatted: is your code laid out in a clear, consistent way?【分数将授予：1.用户“体验”实用性：您的程序是否为生物学家提供了一个看似有用的工具？可用性：您的程序使用起来有多容易？界面：您的程序是否向用户提供了一致的界面？2.帮助手册文档的可读性：您的手册部分是否易于阅读和理解？一般描述：（任何能力水平的）用户是否容易理解程序的功能？3.代码是否使用Python3而不是bash\/awk？它是否使用了我们学到的许多Python3功能？正确：程序运行时是否产生正确的输出？抗错误：程序是否检查输入是否有效，如果不好，是否会以有用的错误消息优雅地失败？结构良好：代码是否被划分为逻辑函数/单元？有据可查：注释是否有助于程序员理解/维护您的代码？简洁、格式良好：您的代码是否以清晰、一致的方式布局？】