Step 1: Describe your project

The project begins with an exploration of the [New York Times API endpoints (Links to an external site.)](https://developer.nytimes.com/). You might have an idea about what kind of data you want to analyze or what kind of question you want to answer. But if the data you need is not available, then you are only setting yourself up for a very challenging – perhaps impossible – final project. Therefore, your first step is to identify an actionable goal, e.g., a question to answer, for your project by exploring the documentation for the [New York Times API (Links to an external site.)](https://developer.nytimes.com/).

With that question in mind, it’s time to sketch out each of the stages in your ETL pipeline. The main questions for which you need to provide answers are as follows:

步骤1：描述您的专案

该项目从探索《纽约时报》 API端点（链接到外部站点）开始。 您可能对要分析的数据类型或要回答的问题有一个想法。 但是，如果没有您需要的数据，那么您只是在为一个非常具有挑战性的（也许是不可能的）最终项目做好准备。 因此，您的第一步是通过探索《纽约时报》 API（指向外部网站的链接）的文档，为您的项目确定可行的目标，例如要回答的问题。

考虑到这个问题，现在该勾勒出ETL管道中的每个阶段了。 您需要提供答案的主要问题如下：

1. Where is the data I need? Be specific! Which API endpoint? Are the important pieces of data I need to **extract** from the API actually available from the endpoint(s) I have identified?
2. What do I need to do to **transform** the raw data extracted from the API into something usable? You do not need to be writing code yet! In fact, thinking in Python terms at this stage could be counter-productive. Just think in human terms for now: What about your data needs to be transformed? What parts do you need to keep and what can you discard?
   * ETL projects often require multiple transform stages. For example, one transform stage that cleans and prepares the extracted data, and another that uses built-in Python and/or third-party libraries to transform and even generate new data.
3. What will I **load** my data into, and how? Your raw data, once processed in your transform stages, is going to be loaded into a file or database depending on what software you plan to use to present your results. How should you store it? What format and structure should the data be stored in?

1.我需要哪里的数据？请明确点！哪个API端点？我需要从API提取的重要数据是否实际上已从我确定的端点中获得？

2.我需要怎么做才能将从API提取的原始数据转换成可用的东西？您不需要编写代码！实际上，在此阶段以Python术语进行思考可能适得其反。暂时以人为考虑：您的数据需要转换吗？您需要保留什么零件，可以丢弃什么？

o ETL项目通常需要多个转换阶段。例如，一个转换阶段将清理并准备提取的数据，而另一个阶段则使用内置的Python和/或第三方库来转换甚至生成新数据。

3.我将数据加载到什么以及如何加载？在转换阶段中处理完后，原始数据将被加载到文件或数据库中，具体取决于您计划使用哪种软件来显示结果。您应该如何存储它？数据应以什么格式和结构存储？

For the first step in your project, describe the following:

1. What is the goal or question of your project?
2. Where is the data?
3. What will you do to transform the data?
4. How will you store the processed data?

You will submit this informal (but important) explanation through Canvas as the Data Pipeline Project (Final Project) Plan.

对于项目的第一步，请描述以下内容：

1.您的项目的目标或问题是什么？

2.数据在哪里？

3.您将如何转换数据？

4.您将如何存储处理后的数据？

您将通过Canvas作为数据管道项目（最终项目）计划提交此非正式（但很重要）的解释。

Step 2: Write and run your code example

When you are considering how much code you need to write, keep in mind the following: “How much do I need to write?” is never a good question. Moreover, when it comes to code, less code is often better than more code. If you can achieve the same results with half of the code, that’s a good thing! In fact, the same is true when you’re writing prose but that’s outside the scope of this course.

**What matters is not how much code you write, but how much your code does.**

What does you code need to do? Your code needs to perform a Data ETL process. The rest is detail. For an idea of what I’ll be looking for, check out the project checklist that is (or will soon be) posted to Canvas.

步骤2：编写并运行您的代码示例

在考虑需要编写多少代码时，请记住以下几点：“我需要编写多少代码？” 从来都不是一个好问题。 而且，在代码方面，更少的代码通常比更多的代码更好。 如果您用一半的代码就能达到相同的结果，那就太好了！ 实际上，撰写散文时也是如此，但这不在本课程的讨论范围之内。

重要的不是您编写了多少代码，而是您编写了多少代码。

您需要编写什么代码？ 您的代码需要执行数据ETL过程。 剩下的就是细节。 要了解我要寻找的内容，请查看已（或即将发布）到Canvas的项目清单。

**You also need to create a code execution video.** This video is just a backup measure in case we find ourselves in a circumstance where you believe your code successfully runs but I cannot make it run. The video must clearly be you, on your computer, running your code. The best way to do this is to:

1. Record your screen with audio.
2. Say hello at the beginning of the code, and me your name and RUID.
3. Use your computer’s clock or calendar app to confirm that the video is recorded before the deadline.
4. Execute your code and explain whatever is necessary to clearly demonstrate that the code executed successfully.

This video does not have to be any kind of polished presentation. It just needs to be a clear demonstration.

您还需要创建一个代码执行视频。 如果您认为自己的代码成功运行，但我无法运行，则该视频只是一种备用措施。 视频必须清楚地由您在计算机上运行您的代码。 最好的方法是：

1.用音频录制屏幕。

2.在代码开头打个招呼，然后输入您的姓名和RUID。

3.使用计算机的时钟或日历应用确认在截止日期之前已录制视频。

4.执行您的代码并说明为清楚证明代码已成功执行所必需的一切。

该视频不必是任何精美的演示。 这只是一个清晰的示范。

Step 3: Create code-images and write your report

Summarize your project in a short written report. Here are the contents you need to produce:

* A **Summary** of the purpose or goal of your project, the technologies used, and the results. This shouldn’t take more than a paragraph, but you should write as much as you feel is necessary to summarize your project.
* A series of **Code images** [using Carbon (Links to an external site.)](https://carbon.now.sh/) to display the important parts of your code. What are the important parts? That’s for you to decide – it should not be *every* line of code, but what lines do important things in the context of your project?
* Include **Code descriptions** based on each code image.
* Wrap up with **more ideas** about what you would do next if you had to (or chose to) do a follow-up project.

步骤3：建立程式码图片并撰写报告

在简短的书面报告中总结您的项目。 这是您需要制作的内容：

•项目目的或目标，所用技术和结果的摘要。 这不应该超过一个段落，但是您应该写一些您认为总结项目所需的内容。

•使用Carbon的一系列代码图像（链接到外部站点。）以显示代码的重要部分。 重要的部分是什么？ 这是由您决定的-它不应该是每一行代码，但是在您的项目范围内，哪些行在做重要的事情？

•包括基于每个代码图像的代码描述。

•总结更多关于如果必须（或选择）进行后续项目的下一步工作的想法。