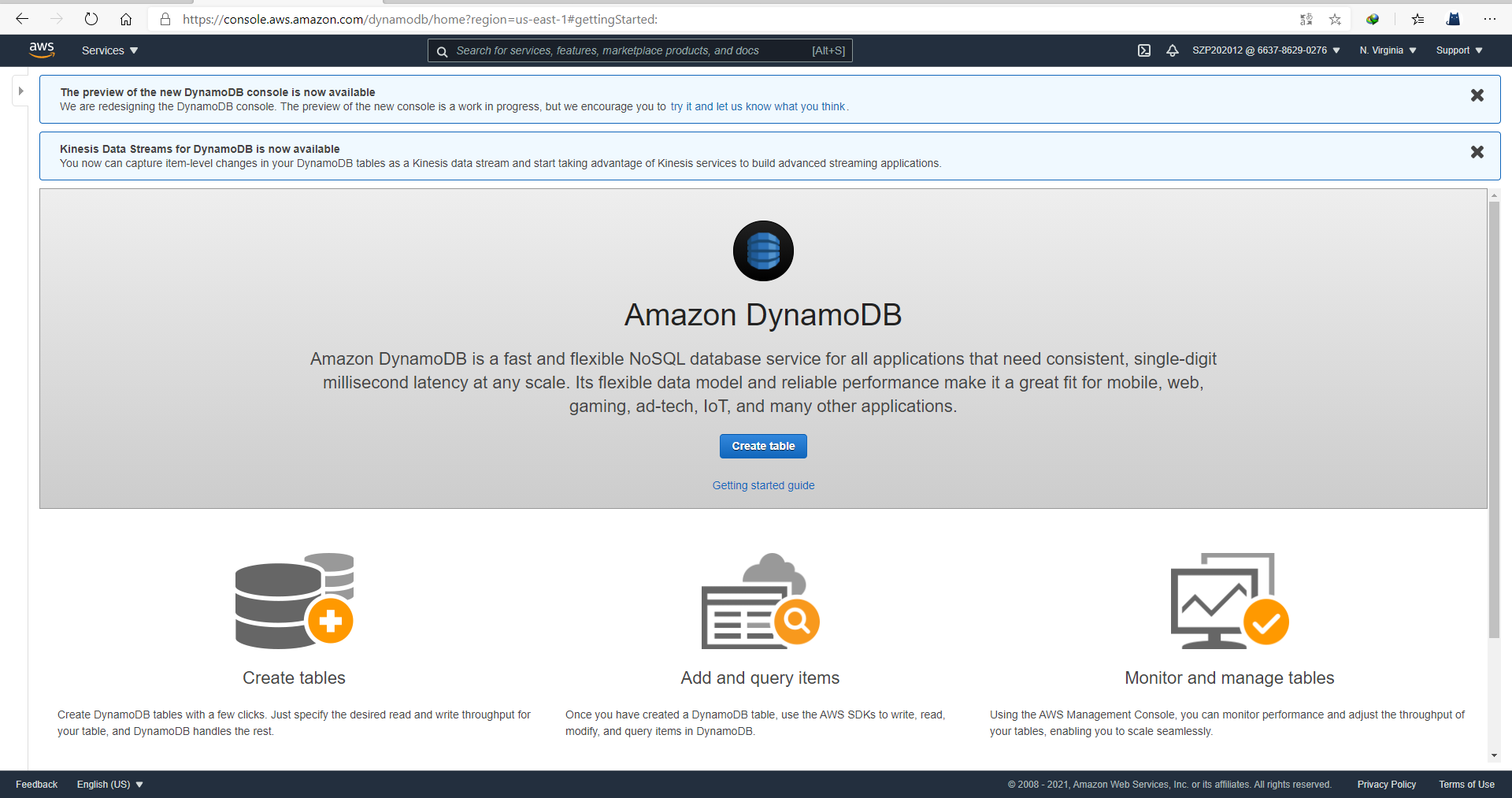
DynamoDB

什么是DynamoDB

Amazon DynamoDB 是一种完全托管的 NoSQL 数据库服务，提供快速而可预测的性能，同时还能够无缝扩展。DynamoDB 让您减轻操作和扩展分布式数据库的管理负担，这样您就不必担心硬件预置、设置和配置、复制、软件修补或集群扩展。\



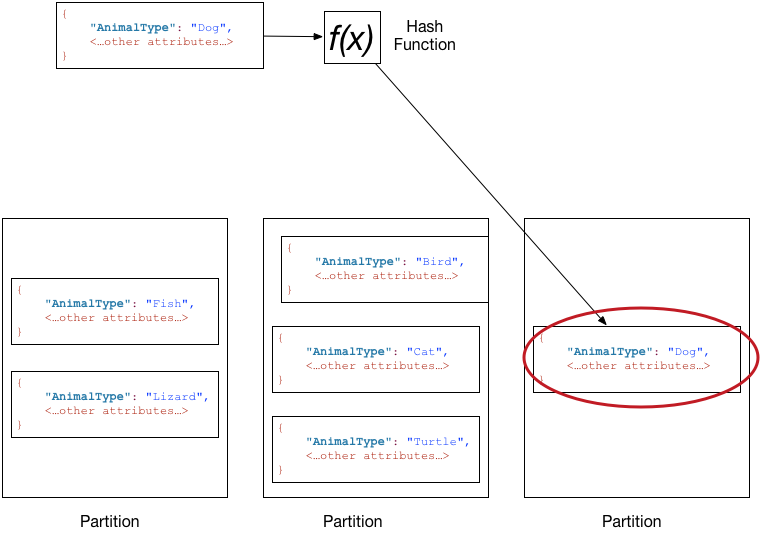
分区键

如果表具有简单主键 (只有分区键)，DynamoDB 将根据其分区键值存储和检索各个项目。

DynamoDB 使用分区键的值作为内部散列函数的输入值，从而将项目写入表中。散列函数的输出值决定了项目将要存储在哪个分区。

要从表中读取某个项目，您必须为该项目指定分区键值。DynamoDB 使用此值作为其哈希函数的输入值，从而生成可从中找到该项目的分区。

下图显示了名为 Pets 的表，该表跨多个分区。表的主键为 AnimalType（仅显示此键属性）。DynamoDB 使用其哈希函数根据字符串 Dog 的哈希值确定存储新项目的位置。请注意，项目并非按排序顺序存储的。每个项目的位置由其分区键的哈希值决定。



排序键

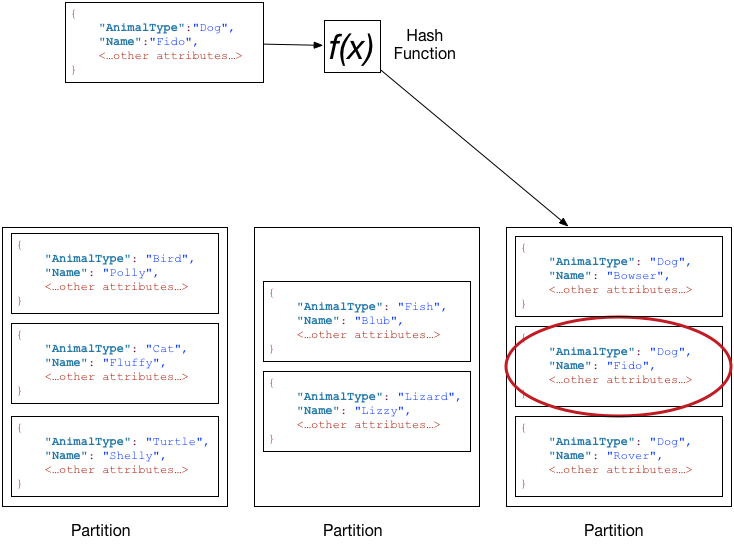
如果表具有复合主键（分区键和排序键），DynamoDB 将计算分区键的哈希值。但是，它按排序键值有序地将具有相同分区键值的项目存储在互相紧邻的物理位置。

为将某个项目写入表中，DynamoDB 会计算分区键的散列值以确定该项目的存储分区。在该分区中，可能有几个具有相同分区键值的项目。因此，DynamoDB 会按排序键的升序将该项目存储在具有相同分区键的其他项目中。

要读取表中的某个项目，您必须为该项目指定分区键值和排序键值。DynamoDB 会计算分区键的哈希值，从而生成可从中找到该项目的分区。

如果您想要的项目具有相同的分区键值，则可以通过单一操作 (Query) 读取表中的多个项目。DynamoDB 将返回具有该分区键值的所有项目。或者，您也可以对排序键应用某个条件，以便它仅返回特定值范围内的项目。

假设 Pets 表具有由 AnimalType（分区键）和 Name（排序键）组成的复合主键。下图显示了 DynamoDB 写入项目的过程，分区键值为 Dog、排序键值为 Fido。



索引

索引使您能够访问替代查询模式，并可以加快查询速度。

Amazon DynamoDB 通过指定主键值来提供对表中项目的快速访问。但是，很多应用程序可能适合有一个或多个二级（或替代）键，以便通过主键以外的属性对数据进行高效访问。要解决此问题，您可以对表创建一个或多个二级索引，然后对这些索引发出 Query 或 Scan 请求。

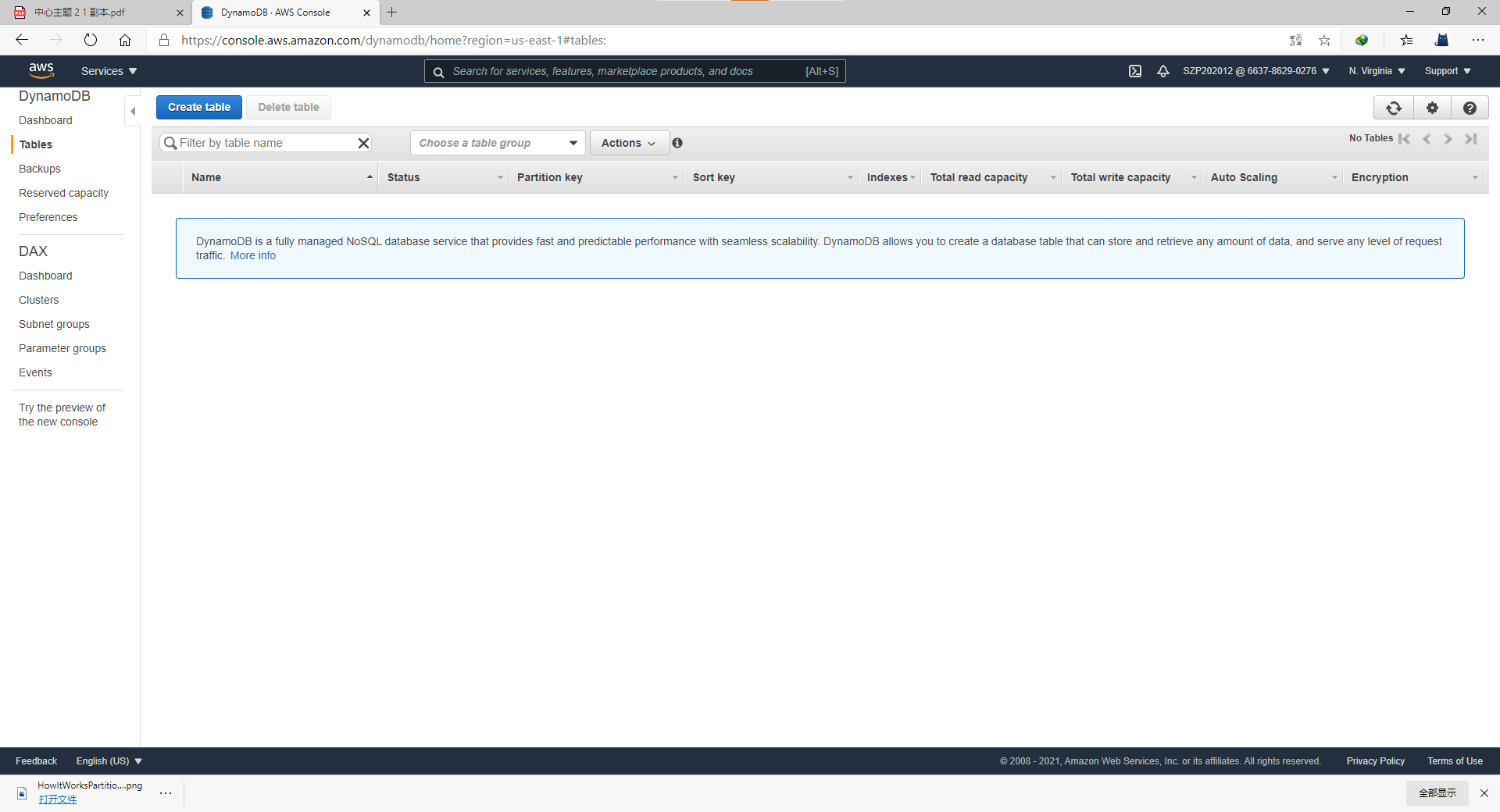
二级索引 是一种数据结构，它包含表中属性的子集以及一个支持 Query 操作的替代键。您可以使用 Query 从索引中检索数据，其方式与对表使用 Query 大致相同。一个表可以有多个二级索引，这样，应用程序可以访问许多不同的查询模式。

无论您使用的是关系数据库还是 DynamoDB，在创建索引时都应谨慎。只要对表进行写入，就必须更新表的所有索引。在具有大型表的写入密集型环境中，这会占用大量系统资源。在只读环境或主读环境中，这算不上一个问题。但是，您应确保索引实际上由应用程序使用，而不只是占用空间。

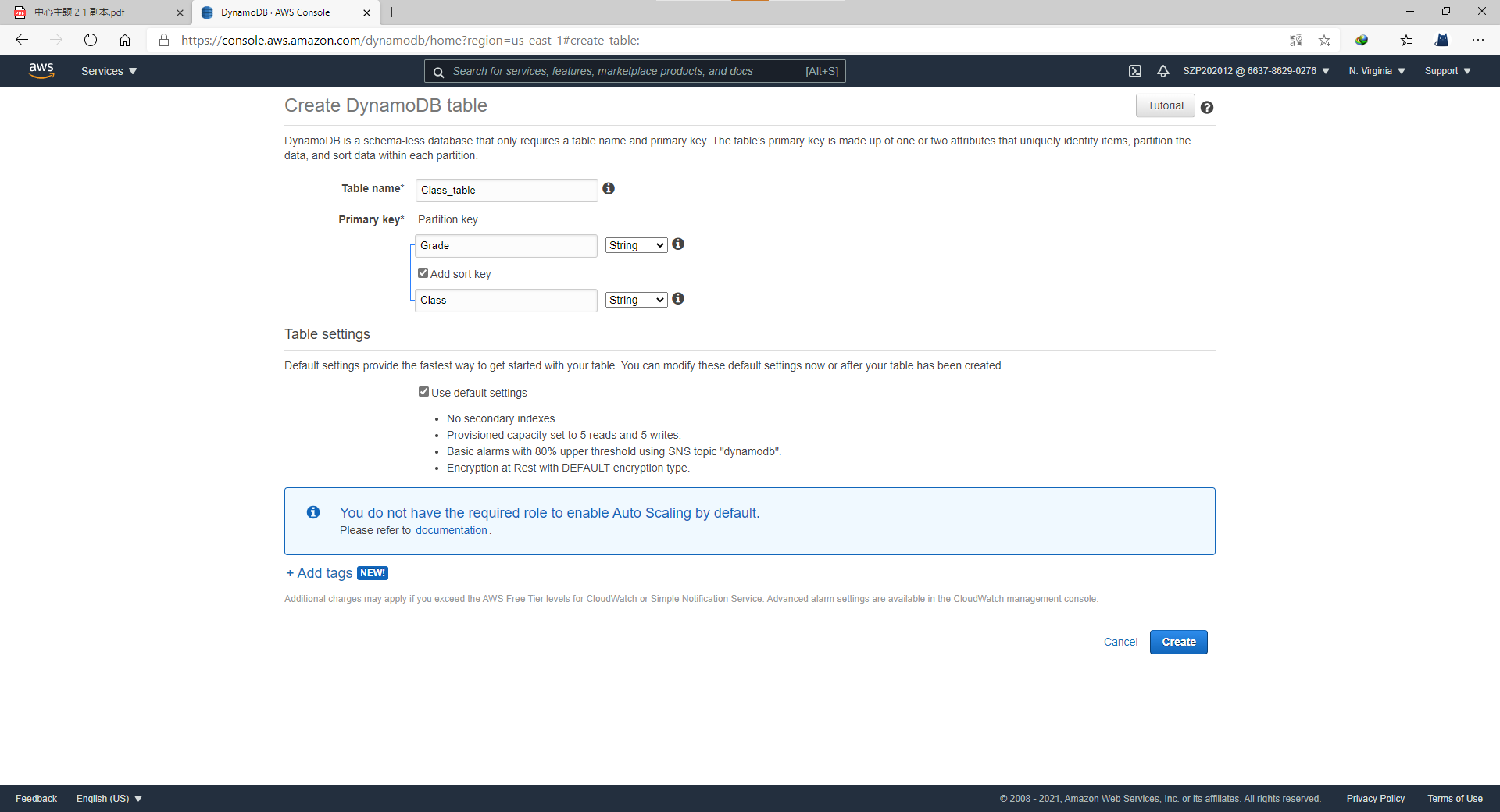


实验：创建复合组建的表并创建全局二级索引

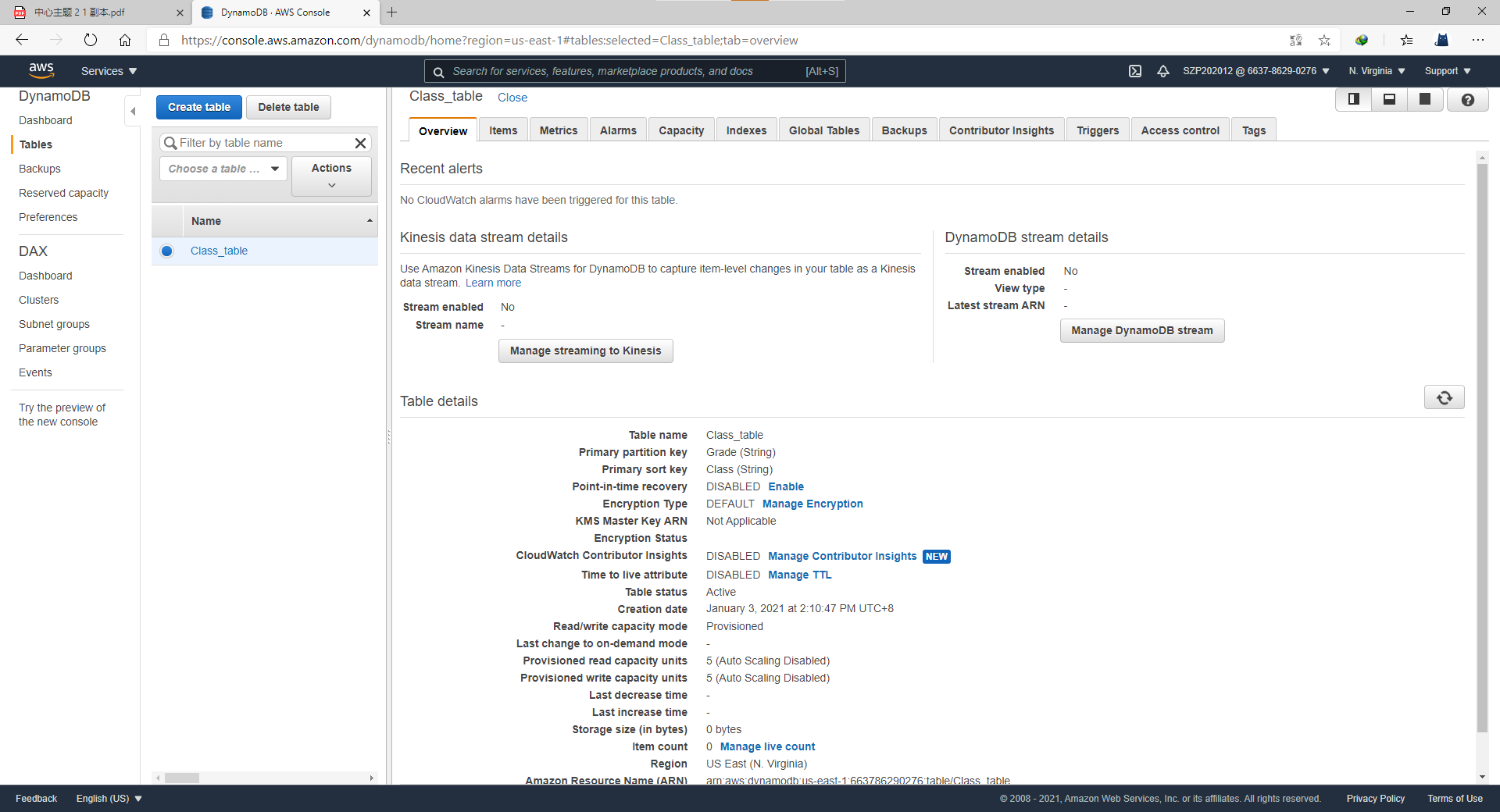
1. 点击主界面Create table



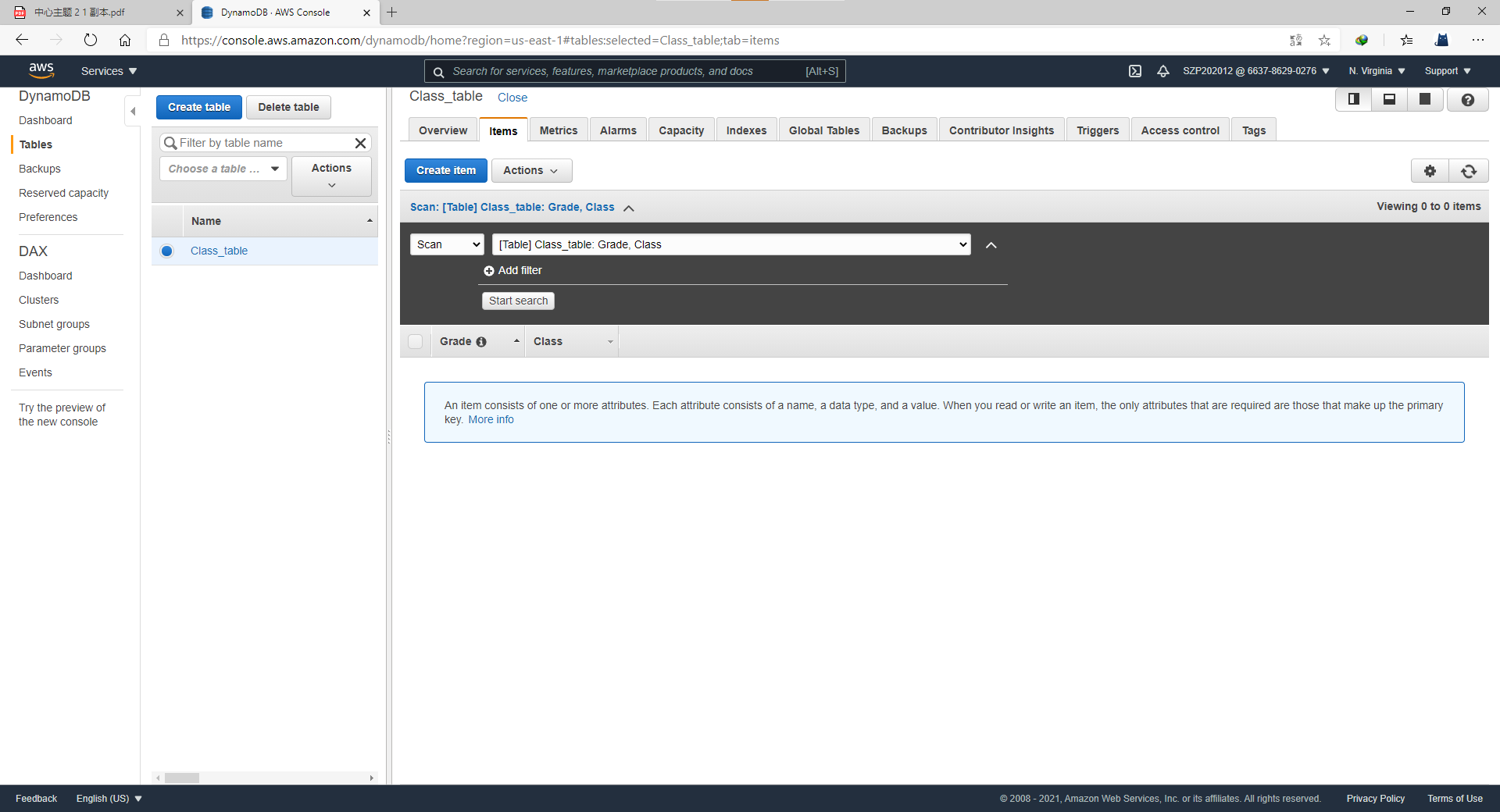
1. 创建表，勾选Add sort key，点击Create



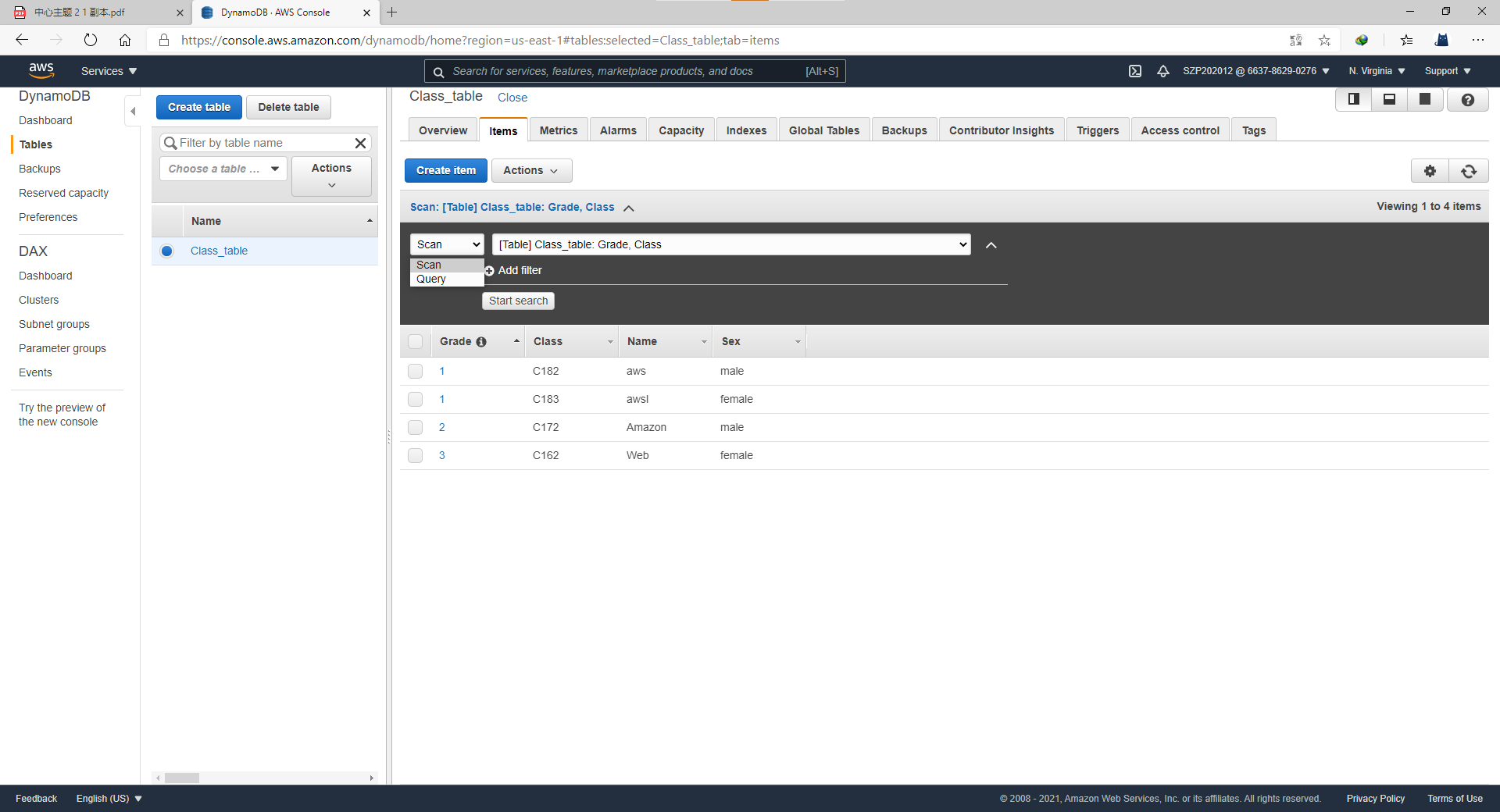
1. 一张表就创建完了



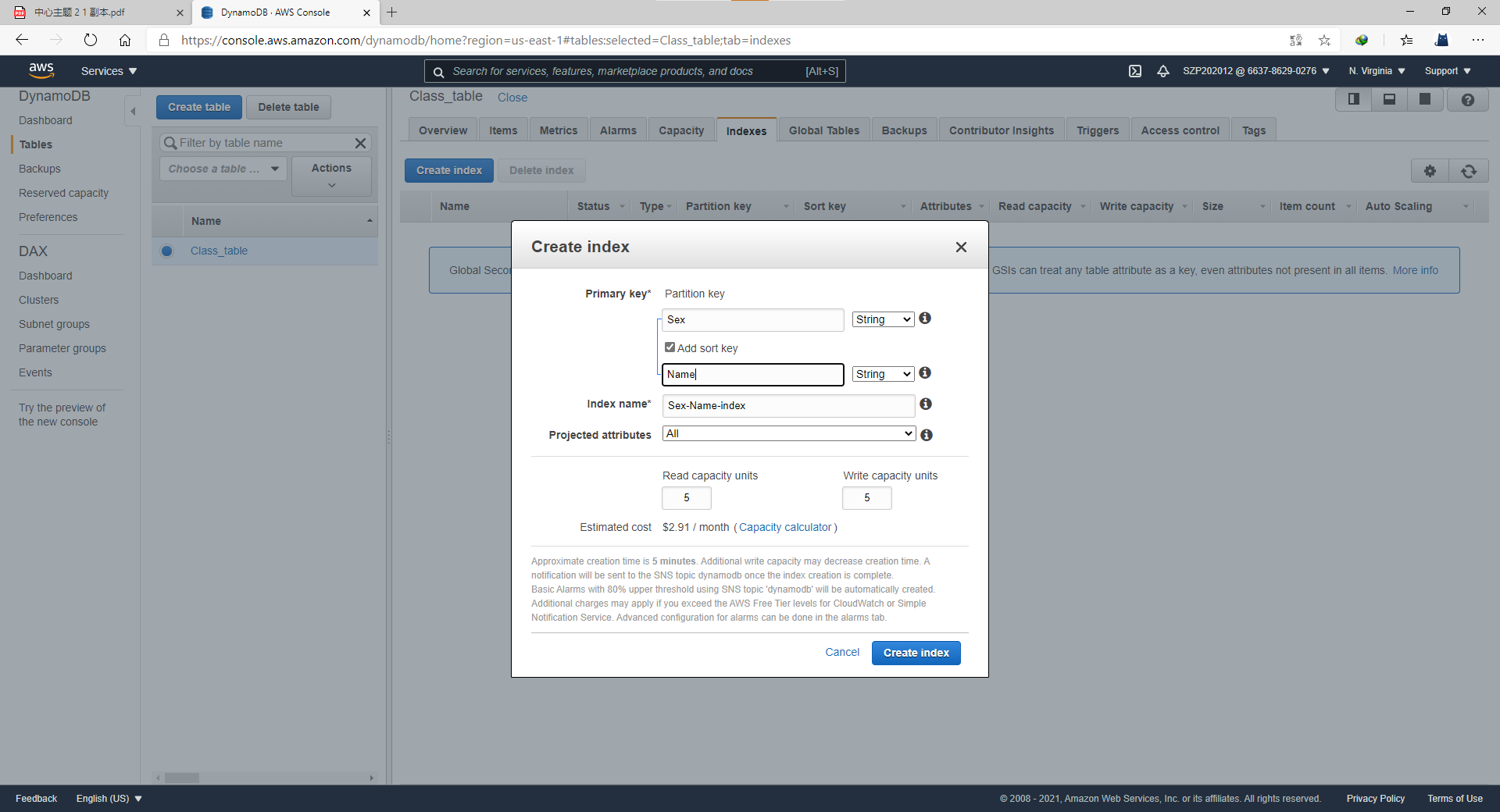
1. 我们试着往其中添加项目，点击Item->Create item



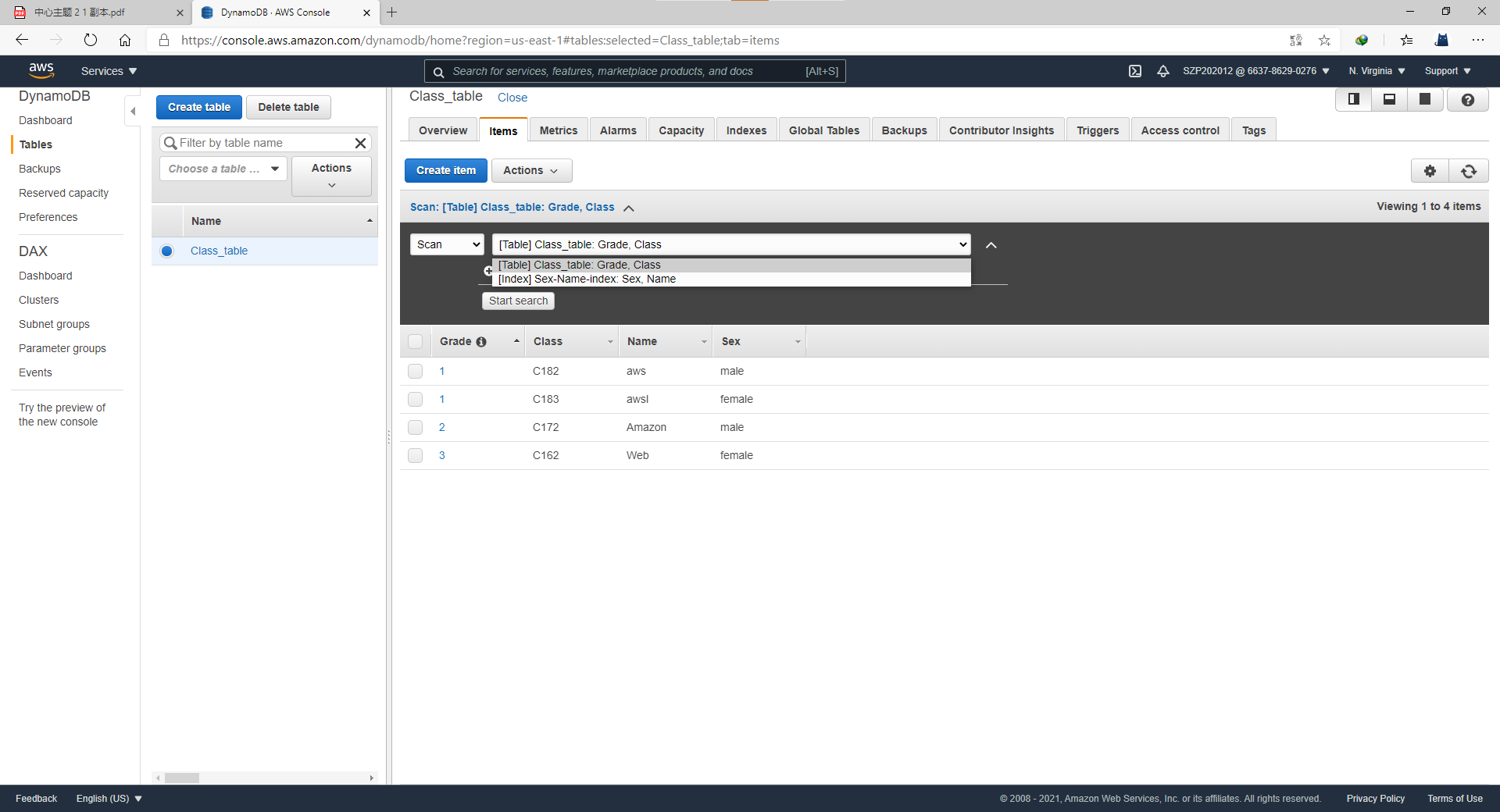
尝试着填写项目吧，这时候发现“查询”有两种方式，这两种方式有何区别呢？



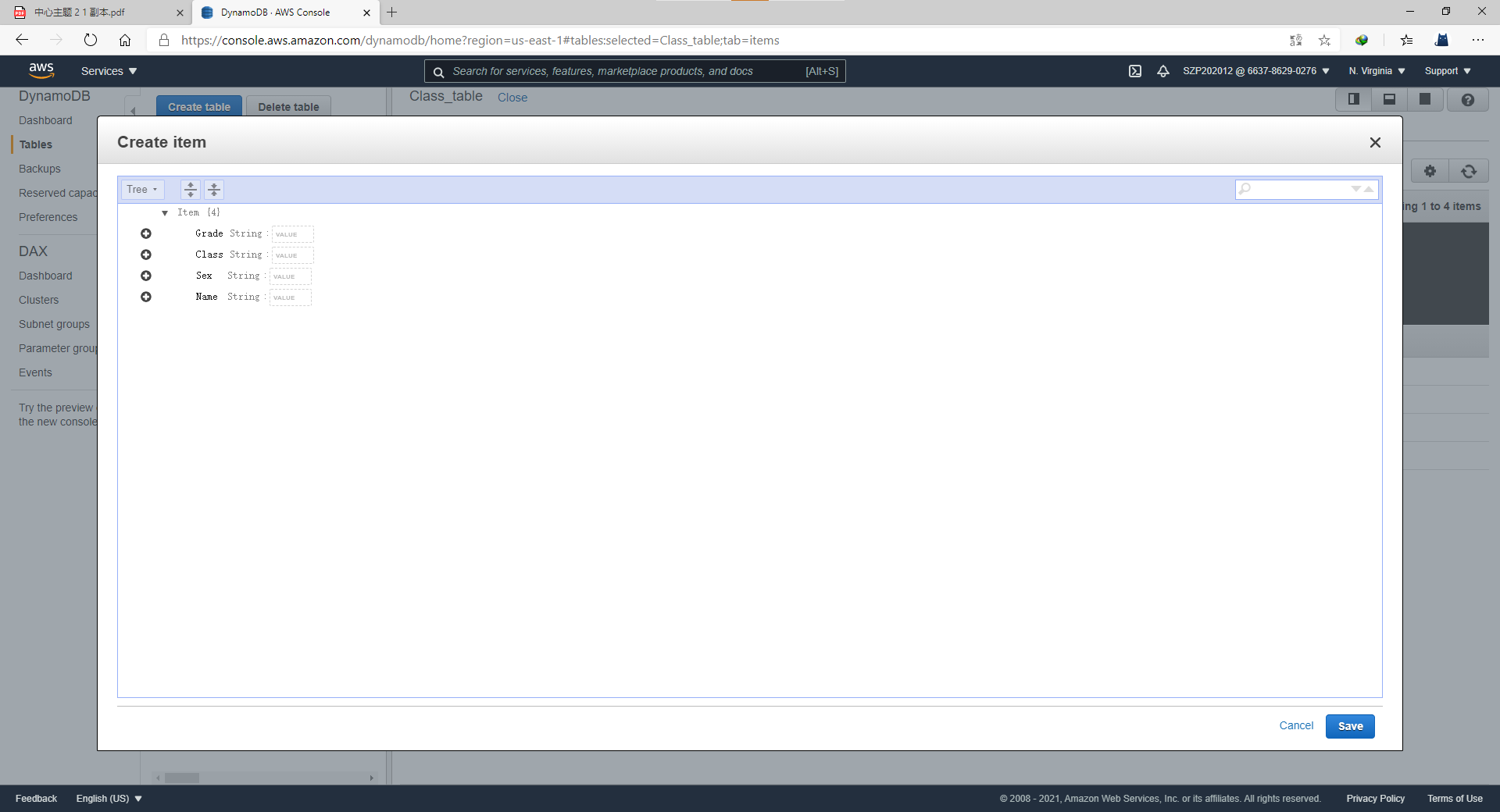
1. 创建索引，点击Indexes->Create index，这次我们拿性别分区，名字排序



1. 等待其创建成功，回到Items发现可以选择索引进行查询了（如果没显示请刷新）



此时发现四个项目都变成了必填项

****

恭喜！实验完成