# 使用 Order By 对 DocumentDB 数据进行排序

Microsoft Azure DocumentDB 支持使用 JSON 文档的 SQL 对文档进行查询。可以使用 SQL 查询语句中的 ORDER BY 子句对查询结果进行排序。

阅读本文之后，你将能够回答以下问题：

* 如何使用 Order By 进行查询？
* 如何为 Order By 配置索引策略？
* 即将推出的内容有哪些？

还提供了[示例](#samples)和[常见问题](#faq)。

有关 SQL 查询的完整参考，请参阅 [DocumentDB 查询教程](/documentation/articles/documentdb-sql-query)。

## 如何使用 Order By 进行查询

正如在 ANSI-SQL 中，现在可在查询 DocumentDB 时将可选 Order By 子句包括在 SQL 语句中。该子句可以包括可选的 ASC/DESC 参数以指定检索结果必须遵守的顺序。

### 使用 SQL 进行排序

例如，下面的查询以标题降序形式检索前 10 本书籍。

SELECT TOP 10 \*   
FROM Books   
ORDER BY Books.Title DESC

### 配合使用排序（通过 SQL）与筛选

可以在文档内使用任何嵌套属性（如 Books.ShippingDetails.Weight）进行排序，还可以在 WHERE 子句和 Order By 组合中指定其他筛选器，例如：

SELECT \*   
FROM Books   
WHERE Books.SalePrice > 4000  
ORDER BY Books.ShippingDetails.Weight

### 使用 .NET 的 LINQ 提供程序进行排序

使用 .NET SDK 版本 1.2.0 和更高版本，你还可在 LINQ 查询中使用 OrderBy() 或 OrderByDescending() 子句，例如：

foreach (Book book in client.CreateDocumentQuery<Book>(UriFactory.CreateDocumentCollectionUri("db", "books"))  
 .OrderBy(b => b.PublishTimestamp)  
 .Take(100))  
{  
 // Iterate through books  
}

DocumentDB 支持每个查询使用单个数值、字符串或布尔值属性进行排序，即将推出其他查询类型。请参阅[即将推出的内容](#Whats_coming_next)以了解详细信息。

## 为 Order By 配置索引策略

回想一下，DocumentDB 支持两种索引（哈希和范围），可以为特定的路径/属性、数据类型（字符串/数字）设置该索引并且是以不同的精度值（最大精度值或固定精度值）。由于 DocumentDB 使用哈希索引做为默认索引，因此必须使用带有数字、字符串（或两者）范围的自定义索引策略创建新集合，以便能够使用 Order By。

[AZURE.NOTE] 在 2015 年 7 月 7 日推出了字符串范围索引，以及 REST API 版本 2015-06-03。若要为 Order By 创建针对字符串的策略，必须使用 .NET SDK 的 SDK 版本 1.2.0，或者 Python、Node.js 或 Java SDK 的版本 1.1.0。

在 REST API 版本 2015-06-03 之前，默认的集合索引策略为字符串哈希和数字哈希。这都已更改为字符串哈希和数字范围。

有关详细信息，请参阅 [DocumentDB 索引策略](/documentation/articles/documentdb-indexing-policies)。

### 针对所有属性的 Order By 的索引

下面显示如何针对集合中 JSON 文档内出现的所有数字或字符串属性使用“所有范围”索引为 Order By 创建集合。此处我们将字符串值默认索引类型改写为范围，并且采用最大精度 (-1)。

DocumentCollection books = new DocumentCollection();  
books.Id = "books";  
books.IndexingPolicy = new IndexingPolicy(new RangeIndex(DataType.String) { Precision = -1 });  
  
await client.CreateDocumentCollectionAsync(UriFactory.CreateDatabaseUri("db"), books);

[AZURE.NOTE] 请注意，Order By 只能返回使用 RangeIndex 作为索引的数据类型（字符串和数字）的结果。例如，如果你的默认索引策略仅有数字的 RangeIndex，那么针对含字符串值的路径使用 Order By 将不返回任何文档。

如果已为集合定义了分区键，请注意仅在针对单个分区键筛选的查询中支持 Order By。

### 针对单个属性的 Order By 的索引

下面显示如何只是针对标题属性（为字符串）使用索引为 Order By 创建集合。这里有两个路径，一个用于具有范围索引的标题属性 (“/Title/?”)，另一个用于具有默认索引方案（即字符串哈希和数字范围）的其他每个属性。

booksCollection.IndexingPolicy.IncludedPaths.Add(  
 new IncludedPath {   
 Path = "/Title/?",   
 Indexes = new Collection<Index> {   
 new RangeIndex(DataType.String) { Precision = -1 } }   
 });  
  
await client.CreateDocumentCollectionAsync(UriFactory.CreateDatabaseUri("db"), booksCollection);

## 示例

查看 [Github 示例项目](https://github.com/Azure/azure-documentdb-dotnet/tree/master/samples/code-samples/Queries)，了解如何使用 Order By，包括创建索引策略和使用 Order By 进行分页。这些示例是开放源代码的，并且我们鼓励你提交可让其他 DocumentDB 开发人员获益的相关拉取请求。请参考 [Contribution guidelines（贡献准则）](https://github.com/Azure/azure-documentdb-net/blob/master/Contributing.md)以获如何做出贡献的指导。

## 常见问题

**Order By 查询的预期请求单位 (RU) 使用情况如何？**

由于 Order By 利用 DocumentDB 索引进行查找，Order By 查询使用的请求单位数将类似于没有 Order By 的等效查询。正如对 DocumentDB 的任何其他操作，请求单位的数量取决于文档的大小/形状以及查询的复杂性。

**预期的 Order By 的索引开销如何？**

索引存储开销将与属性的数目成比例。在最坏的情况下，索引开销将是数据的 100%。范围/Order By 索引和默认哈希索引之间的吞吐量（请求单位）开销没有什么区别。

**如何使用 Order By 查询 DocumentDB 中现有的数据？**

为了使用 Order By 对查询结果进行排序，必须修改集合的索引策略从而针对用于排序的属性使用范围索引类型。请参阅[修改索引策略](documentdb-indexing-policies.md#modifying-the-indexing-policy-of-a-collection)。

**Order By 当前的限制是什么？**

当以最大精度 (-1) 使用范围作为索引时，可以指定 Order By 仅针对属性（数值或字符串）。

你不能执行以下操作：

* 通过内部字符串属性使用 Order By，如 ID、\_rid 以及 \_self（即将推出）。
* 通过派生自文档内联接结果的属性使用 Order By（即将推出）。
* 使用多个属性进行 Order By（即将推出）。
* 通过数据库、集合、用户、权限或附件上查询使用 Order By（即将推出）。
* 通过计算的属性使用 Order By，如表达式或 UDF/内置函数的结果。

## 后续步骤

分叉 [Github 示例项目](https://github.com/Azure/azure-documentdb-dotnet/tree/master/samples/code-samples/Queries)并开始对数据进行排序！

## 参考

* [DocumentDB 查询参考](/documentation/articles/documentdb-sql-query)
* [DocumentDB 索引策略参考](/documentation/articles/documentdb-indexing-policies)
* [DocumentDB SQL 参考](https://msdn.microsoft.com/library/azure/dn782250.aspx)
* [DocumentDB Order By 示例](https://github.com/Azure/azure-documentdb-dotnet/tree/master/samples/code-samples/Queries)