

NOLOCK FAQ

Last Modified on 04/22/2024 12:02 pm CDT

In Campus. 2331, Infinite Campus removed all NOLOCK (and its equivalent, READUNCOMMITTED) query hints and READ UNCOMMITTED settings from the Infinite Campus code base.

To mitigate the risk of query result inconsistencies, please remove any NOLOCK (and READUNCOMMITTED) query hints from your own custom SQL code **when you take Release Pack .2331 (July 2023) or greater**.

Failure to remove NOLOCK from your queries will not break them, but it can lead to performance issues and data inconsistencies if multiple queries operating on the same data utilize different mechanisms to ensure data integrity.

Please see below for answers to the most frequently asked questions.

- [What are NOLOCK Hints and Why Are They Used?](#)
- [What's an Example of a NOLOCK Hint?](#)
- [Why is Infinite Campus Removing NOLOCK Hints?](#)
- [Am I Impacted?](#)
- [I'm Impacted. What Do I Have to Do and by When?](#)
- [I'm Impacted, But My Custom SQL Code is Maintained by a 3rd party. What Should I Do?](#)

What are NOLOCK Hints and Why Are They Used?

NOLOCK hints, which are equivalent to READUNCOMMITTED hints, are query hints in SQL Server that instruct the query optimizer to read data from tables without acquiring shared locks. When SQL Server's transaction isolation level is set to READ COMMITTED, certain queries can hold a shared lock long enough to block other transactions which can eventually take down the server. Adding NOLOCK hints to these queries effectively mitigates this issue.

What's an Example of a NOLOCK Hint?

Here is an example of a SQL statement that includes a NOLOCK hint:

```
INNER JOIN dbo.[Identity] i WITH (NOLOCK) ON i.identityID = p.currentIdentityID
```

Here is how the statement would be modified to remove the NOLOCK hint:

```
INNER JOIN dbo.[Identity] i ON i.identityID = p.currentIdentityID
```

Why is Infinite Campus Removing NOLOCK Hints?

Infinite Campus has switched from READ COMMITTED to READ COMMITTED SNAPSHOT transaction isolation level.

- READ COMMITTED SNAPSHOT uses row versioning and snapshot isolation rather than shared locks, so using NOLOCK hints is no longer necessary.
- Continued use of NOLOCK hints can decrease performance, as it can cause additional reads and increase contention for system resources.
- Data and query inconsistencies can arise if multiple queries operating on the same data utilize different mechanisms to ensure data integrity.

Am I Impacted?

If you have any custom SQL code that you maintain, whether you've written it yourself or whether it has been provided by Infinite Campus or a 3rd party, you may be impacted.

Custom SQL code is commonly used in the following areas but can be used much more broadly:

- Task Scheduler
- DEU (Data Extract Utility)
- Direct ODBC access
- Custom Reports (SSRS, etc.)
- Custom JSP files

I'm Impacted. What Do I Have to Do and by When?

Infinite Campus recommends removing all NOLOCK (and READUNCOMMITTED) query hints from your custom SQL code when you upgrade to **Release Pack .2331 (July 2023) or greater**. In addition, any queries that set the isolation level to READ UNCOMMITTED should also be removed.

I'm Impacted, But My Custom SQL Code is Maintained by a 3rd party. What Should I Do?

If you have hired a 3rd party to customize your Infinite Campus application or write queries against the Campus Database, please contact the 3rd party and forward this information to them.