In this video we will discuss, **how to find blocking queries in sql server**.   
  
   
  
Blocking occurs if there are open transactions. Let us understand this with an example.   
  
**Execute the following 2 sql statements**  
Begin Tran  
Update TableA set Name='Mark Transaction 1' where Id = 1   
  
Now from a different window, execute any of the following commands. Notice that all the queries are blocked.

Select Count(\*) from TableA

Delete from TableA where Id = 1

Truncate table TableA

Drop table TableA

This is because there is an open transaction. Once the open transaction completes, you will be able to execute the above queries.   
  
So the obvious next question is - **How to identify all the active transactions**.    
  
One way to do this is by using DBCC OpenTran. DBCC OpenTran will display only the oldest active transaction. It is not going to show you all the open transactions.  
DBCC OpenTran   
  
The following link has the SQL script that you can use to identify all the active transactions.   
<http://www.sqlskills.com/blogs/paul/script-open-transactions-with-text-and-plans>   
  
The beauty about this script is that it has a lot more useful information about the open transactions  
Session Id  
Login Name  
Database Name  
Transaction Begin Time  
The actual query that is executed   
  
You can now use this information and ask the respective developer to either commit or rollback the transactions that they have left open unintentionally.   
  
For some reason if the person who initiated the transaction is not available, you also have the option to KILL the associated process. However, this may have unintended consequences, so use it with extreme caution.   
  
There are 2 ways to kill the process are described below  
  
**Killing the process using SQL Server Activity Monitor :**   
1. Right Click on the Server Name in Object explorer and select **"Activity Monitor"**  
2. In the **"Activity Monitor"** window expand Processes section  
3. Right click on the associated **"Session ID"** and select **"Kill Process"** from the context menu   
  
**Killing the process using SQL command :**  
KILL Process\_ID   
  
**What happens when you kill a session**  
All the work that the transaction has done will be rolled back. The database must be put back in the state it was in, before the transaction started.