**CHECK constraint** is used to **limit the range of the values**, that can be entered for a column.  
  
  
Let's say, we have an integer AGE column, in a table. The AGE in general cannot be less than ZERO and at the same time cannot be greater than 150. But, since AGE is an integer column it can accept negative values and values much greater than 150.  
  
  
So, to limit the values, that can be added, we can use CHECK constraint. In SQL Server, CHECK constraint can be created graphically, or using a query.   
  
   
  
   
  
**The following check constraint, limits the age between ZERO and 150.**  
ALTER TABLE tblPerson  
ADD CONSTRAINT CK\_tblPerson\_Age CHECK (Age > 0 AND Age < 150)  
  
  
**The general formula for adding check constraint in SQL Server:**  
ALTER TABLE { TABLE\_NAME }  
ADD CONSTRAINT { CONSTRAINT\_NAME } CHECK ( BOOLEAN\_EXPRESSION )  
  
  
If the BOOLEAN\_EXPRESSION returns true, then the CHECK constraint allows the value, otherwise it doesn't. Since, AGE is a nullable column, it's possible to pass null for this column, when inserting a row. When you pass NULL for the AGE column, the boolean expression evaluates to **UNKNOWN**, and allows the value.  
  
  
**To drop the CHECK constraint:**  
ALTER TABLE tblPerson  
DROP CONSTRAINT CK\_tblPerson\_Age