**[drop、truncate和delete的区别](https://www.cnblogs.com/zhizhao/p/7825469.html)**

（1）DELETE语句执行删除的过程是每次从表中删除一行，并且同时将该行的删除操作作为事务记录在日志中保存以便进行进行回滚操作。

   TRUNCATE TABLE 则一次性地从表中删除所有的数据并不把单独的删除操作记录记入日志保存，删除行是不能恢复的。并且在删除的过程中不会激活与表有关的删除触发器。执行速度快。

（2）表和索引所占空间。

   当表被TRUNCATE 后，这个表和索引所占用的空间会恢复到初始大小，

   DELETE操作不会减少表或索引所占用的空间。

   drop语句将表所占用的空间全释放掉。

（3）一般而言，drop > truncate > delete

（4）应用范围。

    TRUNCATE 只能对TABLE；         DELETE可以是table和view

（5）TRUNCATE 和DELETE只删除数据， DROP则删除整个表（结构和数据）。

（6）truncate与不带where的delete ：只删除数据，而不删除表的结构（定义）drop语句将删除表的结构被依赖的约束（constrain),触发器（trigger)索引（index);依赖于该表的存储过程/函数将被保留，但其状态会变为：invalid。

（7）delete语句为DML（data maintain Language),这个操作会被放到 rollback segment中,事务提交后才生效。如果有相应的 tigger,执行的时候将被触发。

（8）truncate、drop是DLL（data define language),操作立即生效，原数据不放到 rollback segment中，不能回滚

（9）在没有备份情况下，谨慎使用 drop 与 truncate。要删除部分数据行采用delete且注意结合where来约束影响范围。回滚段要足够大。要删除表用drop;若想保留表而将表中数据删除，如果于事务无关，用truncate即可实现。如果和事务有关，或老师想触发trigger,还是用delete。

（10） Truncate table 表名 速度快,而且效率高,因为:   
truncate table 在功能上与不带 WHERE 子句的 DELETE 语句相同：二者均删除表中的全部行。但 TRUNCATE TABLE 比 DELETE 速度快，且使用的系统和事务日志资源少。DELETE 语句每次删除一行，并在事务日志中为所删除的每行记录一项。TRUNCATE TABLE 通过释放存储表数据所用的数据页来删除数据，并且只在事务日志中记录页的释放。

（11） TRUNCATE TABLE 删除表中的所有行，但表结构及其列、约束、索引等保持不变。新行标识所用的计数值重置为该列的种子。如果想保留标识计数值，请改用 DELETE。如果要删除表定义及其数据，请使用 DROP TABLE 语句。

（12） 对于由 FOREIGN KEY 约束引用的表，不能使用 TRUNCATE TABLE，而应使用不带 WHERE 子句的 DELETE 语句。由于 TRUNCATE TABLE 不记录在日志中，所以它不能激活触发器。

一、delete

1、delete是DML，执行delete操作时，每次从表中删除一行，并且同时将该行的的删除操作记录在redo和undo表空间中以便进行回滚（rollback）和重做操作，但要注意表空间要足够大，需要手动提交（commit）操作才能生效，可以通过rollback撤消操作。

2、delete可根据条件删除表中满足条件的数据，如果不指定where子句，那么删除表中所有记录。

3、delete语句不影响表所占用的extent，高水线(high watermark)保持原位置不变。

二、truncate

1、truncate是DDL，会隐式提交，所以，不能回滚，不会触发触发器。

2、truncate会删除表中所有记录，并且将重新设置高水线和所有的索引，缺省情况下将空间释放到minextents个extent，除非使用reuse storage，。不会记录日志，所以执行速度很快，但不能通过rollback撤消操作（如果一不小心把一个表truncate掉，也是可以恢复的，只是不能通过rollback来恢复）。

3、对于外键（foreignkey ）约束引用的表，不能使用 truncate table，而应使用不带 where 子句的 delete 语句。

4、truncatetable不能用于参与了索引视图的表。

三、drop

1、drop是DDL，会隐式提交，所以，不能回滚，不会触发触发器。

2、drop语句删除表结构及所有数据，并将表所占用的空间全部释放。

3、drop语句将删除表的结构所依赖的约束，触发器，索引，依赖于该表的存储过程/函数将保留,但是变为invalid状态。

**总结：**

1、在速度上，一般来说，drop> truncate > delete。

2、在使用drop和truncate时一定要注意，虽然可以恢复，但为了减少麻烦，还是要慎重。

3、如果想删除部分数据用delete，注意带上where子句，回滚段要足够大；

   如果想删除表，当然用drop；

   如果想保留表而将所有数据删除，如果和事务无关，用truncate即可；

   如果和事务有关，或者想触发trigger，还是用delete；

   如果是整理表内部的碎片，可以用truncate跟上reuse stroage，再重新导入/插入数据。

4、truncate 和delete都是删除表数据，但是前者是对整个表的操作，不能使用条件，即删除的是整个表数据；后者虽然也可删除整个表数据，但是后者可以添加删除条件，根据用户需求删除对应的数据。