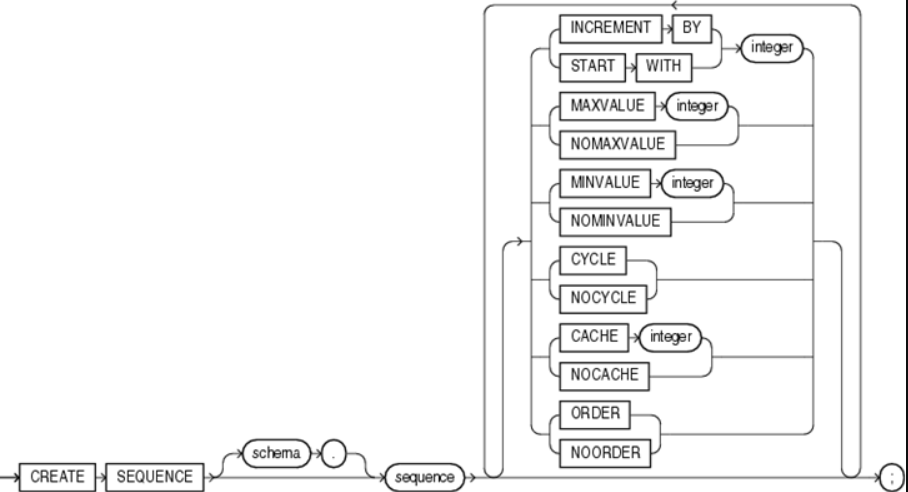
1. Что такое последовательность?

Объект БД, предназначенный для генерации числовой последовательности

1. Перечислите основные параметры последовательности.



* + **start** **with —** позволяет создателю последовательности указать первое генерируемое ею значение. После создания последовательность генерирует указанное в start with значение при первой ссылке на ее виртуальный столбец **NEXTVAL**
  + **increment by n** — определяет приращение последовательности при каждой ссылке на виртуальный столбец **NEXVAL**. Если значение не указано явно, по умолчанию устанавливается 1. Для возрастающих последовательностей устанавливается положительное n, для убывающих, или последовательностей с обратным отсчетом - отрицательное
  + **minvalue** — определяет минимальное значение, создаваемое последовательностью. Если оно не указано, Oracle применяет значение по умолчанию **NOMINVALUE**
  + **nominvalue** — указывает, что минимальное значение равно 1, если последовательность возрастает, или -10, если последовательность убывает

26

* + **maxvalue** — определяет максимальное значение, создаваемое последовательностью. Если оно не указано, Oracle применяет значение по умолчанию **NOMAXVALUE**
  + **nomaxvalue** — указывает, что максимальное значение равно 10, если последовательность возрастает, или -1, если последовательность убывает. По умолчанию принимается **NOMAXVALUE**
  + **cycle** — позволяет последовательности повторно использовать созданные значения при достижении **MAXVALUE** или **MINVALUE**. Т.е. последовательность будет продолжать генерировать значения после достижения своего максимума или минимума. Возрастающая последовательность после достижения своего максимума генерирует свой минимум. Убывающая последовательность после достижения своего минимума генерирует свой максимум. Если циклический режим нежелателен или не установлен явным образом, Oracle применяет значение по умолчанию – **NOCYCLE**. Указывать **CYCLE** вместе с **NOMAXVALUE** или **NOMINVALUE** нельзя. Если нужна циклическая последовательность, необходимо указать **MAXVALUE** для возрастающей последовательности или **MINVALUE** – для убывающей
  + **nocycle** — указывает, что последовательность не может продолжать генерировать значения после достижения своего максимума или минимума
  + **cache** **n** — указывает, сколько значений последовательности **ORACLE** распределяет заранее и поддерживает в памяти для быстрого доступа. Минимальное значение этого параметра равно 2. Для циклических последовательностей это значение должно быть меньше, чем количество значений в цикле. Если кэширование нежелательно или не установлено явным образом, **Oracle** применяет значение по умолчанию – 20 значений.
  + **order** — гарантирует, что номера последовательности генерируются в порядке запросов. Эта опция может использоваться, к примеру, когда номера последовательности предстают в качестве отметок времени. Гарантирование порядка обычно не существенно для тех последовательностей, которые используются для генерации первичных ключей. Если упорядочение нежелательно или не установлено явным образом, Oracle применяет значение по умолчанию **NOORDER**
  + **noorder** — не гарантирует, что номера последовательности генерируются в порядке запросов

1. Какие привилегии необходимы для создания и удаления последовательности?

CREATE SEQUENCE

1. Что такое кластер?

Таблицы, с которыми часто работают совместно, можно физически хранить совместно. Для этого создается кластер, который будет их содержать. Строки из отдельных таблиц сохраняются в одних и тех же блоках, поэтому объединяющие запросы выполняются быстрее.

**Кластер** – объект БД, который хранит значения общих столбцов нескольких таблиц.

1. Что означает параметр **hash**?

**Хэш-кластеры** используют функции хэширования кластерного ключа строки для определения физической локализации места, где строку следует хранить.

1. Какие привилегии необходимы для создания и удаления кластера?

CREATE CLUSTER

1. Что такое синоним?

**Синоним –** способ обращаться к объекту базы данных без указания обязательной полной идентификации объекта (хост – экземпляр – владелец – объект).

1. Чем отличается публичный синоним от частного синонима?

**Частный синоним** принадлежит пользователю, который его создал.

**Публичный синоним** используется совместно всеми пользователями базы данных

1. Что такое материализованное представление?

**Материализо́ванное представле́ние —** физический объект базы данных, содержащий результат выполнения запроса.

**Материализованные представления** позволяют многократно ускорить выполнение запросов, обращающихся к большому количеству (сотням тысяч или миллионам) записей, позволяя за секунды (и даже доли секунд) выполнять запросы к терабайтам данных. Это достигается за счет прозрачного использования заранее вычисленных итоговых данных и результатов соединений таблиц. Предварительно вычисленные итоговые данные обычно имеют очень небольшой объем по сравнению с исходными данными.

1. Чем отличается материализованное представление от обыкновенного представления?

*Материализованные представления* основаны на дисках и периодически обновляются на основе определения запроса.

*Представления* являются только виртуальными и запускают определение запроса каждый раз, когда к ним обращаются.