**Сущность** - класс, хранящийся в базе данных, таблица.

**Объект** - экземпляр сущности, обычно это строка в таблице.

**Атрибут** - свойства характеризующие сущность, название столбца в таблице.

**Кортеж** - строка в таблице или выборке, набор значений конкретных атрибутов.

**Ключ** - это элемент данных, который используется для идентификации записи, или значение, используемое для идентификации записи в базе данных, называется **ключом**. Это помогает однозначно идентифицировать объект из набора объектов.Ключ позволяет нам идентифицировать набор атрибутов, которые делают его достаточным для отличия сущностей друг от друга.

**Сущность в базе данных** – это любой объект в базе данных, который можно выделить исходя из сути предметной области для которой разрабатывается эта база данных. Разработчик базы данных должен уметь правильно определять сущности.

**Систе́ма управле́ния ба́зами да́нных**, сокр. **СУБД** — совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных.

**Валидация данных**— это процесс проверки данных различных типов по критериям корректности и полезности для конкретного применения.

**Связь** является **идентифицирующей** тогда и только тогда, когда первичный ключ дочерней сущности содержит внешний ключ, идущий от родительской сущности. Если такой вещи нет - связь будет **неидентифицирующей**. Объясняю **на примере** Ответов. Есть две сущности - ВОПРОС и ОТВЕТ. Связь ВОПРОС-ОТВЕТ является идентифицирующей, поскольку сущность ОТВЕТ не может быть однозначно определена, если не задана сущность ВОПРОС (просто ответов без вопроса нет) . Или там, сущности ДОМ и КВАРТИРА. Сущность КВАРТИРА не может быть однозначно определена, если не задана сущность ДОМ. Что касается неидентифицирующих связей. Есть сущность ЗВАНИЕ и сущность ПОЛЬЗОВАТЕЛЬ. Пользователь обладает каким-то званием (в Вашем случае "Ученик"), но сущность Пользователь может быть однозначно определена и без этого звания (например, по e-mail). То есть поле Звание таблицы Пользователь не входит в первичный ключ сущности Пользователь.

**Связь многие-ко-многим** – это связь, при которой множественным записям из одной таблицы (A) могут соответствовать множественные записи из другой (B). Примером такой связи может служить школа, где учителя обучают учащихся. В большинстве школ каждый учитель обучает многих учащихся, а каждый учащийся может обучаться несколькими учителями.

**Существуют 7 нормальных форм:**

Первая нормальная форма (1NF): сохраняемые данные на пересечении строк и столбцов должны представлять скалярное значение, а таблицы не должны содержать повторяющихся строк.

Вторая нормальная форма (2NF): каждый столбец, не являющийся ключом, должен зависеть от первичного ключа.

Третья нормальная форма (3NF): каждый столбец, не являющийся ключом, должен зависеть только от первичного ключа.

Четвертая нормальная форма (4NF): применяется для устранения многозначных зависимостей.

Пятая нормальная форма (5NF): разделяет таблицы на более малые таблицы для устранения избыточности данных.

Шестая нормальная форма (6NF): каждое ограничение в связях между таблицами должно зависеть только от ограничений ключа и ограничений домена.