

Лабораторная работа № 3

Тема: разработка моделей информационных систем по методологии моделирования данных (IDEF1X)

Цель: изучить методологию моделирования данных (IDEF1X), освоить CASE-средство AllFusion ERWin Data Modeller для построения диаграмм IDEF1X, научиться строить модели информационных систем, используя данную методологию

Краткая теория

Методология моделирования данных IDEF1X, входящая в семейство стандартов IDEF, использует разновидность модели «сущность-связь» и реализована в ряде распространенных CASE-средств (в частности, ERwin Data Modeller).

Сущность в методе IDEF1X является независимой от идентификаторов, или просто независимой, если каждый экземпляр сущности может быть однозначно идентифицирован без определения его отношений с другими сущностями. Сущность называется зависимой от идентификаторов, или просто зависимой, если однозначная идентификация экземпляра сущности зависит от его отношения к другой сущности. Графическое изображение сущностей приведено на рисунке 1. Каждой сущности присваивается уникальное имя и номер, разделяемые косой чертой «/» и помещаемые над блоком.

Каждой сущности присваивается уникальное имя и номер, разделяемые косой чертой «/» и помещаемые над блоком. Имя сущности должно быть именем существительным в единственном числе.

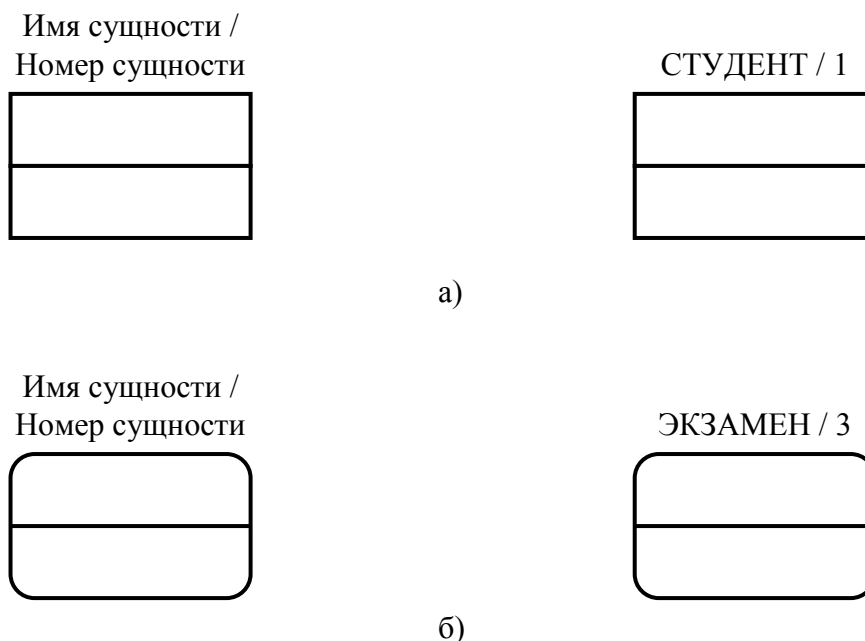


Рисунок 1 – Независимые (а) и зависимые (б) от идентификатора сущности

Связь может дополнительно определяться с помощью указания мощности (количества экземпляров сущности-потомка, которое может существовать для каждого экземпляра сущности-родителя). В IDEF1X могут быть выражены следующие мощности связей:

- каждый экземпляр сущности-родителя может иметь нуль, один или более одного связанного с ним экземпляра сущности-потомка;
- каждый экземпляр сущности-родителя должен иметь не менее одного связанного с ним экземпляра сущности-потомка;

- каждый экземпляр сущности-родителя должен иметь не более одного связанного с ним экземпляра сущности-потомка;
- каждый экземпляр сущности-родителя связан с некоторым фиксированным числом экземпляров сущности-потомка.

Если экземпляр сущности-потомка однозначно определяется своей связью с сущностью-родителем, то связь называется идентифицирующей, в противном случае — неидентифицирующей.

Связь изображается линией, проводимой между сущностью-родителем и сущностью-потомком с точкой на конце линии у сущности-потомка (рисунок 2). Мощность связи может принимать следующие значения: N – нуль, один или более, Z – нуль или один, P – один или более. По умолчанию мощность связи принимается равной N.

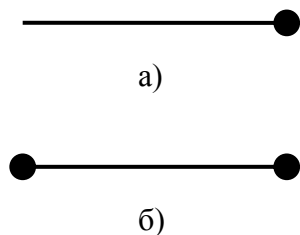


Рисунок 2 – Графическое обозначение связи

а) – мощность «один–ко–многим», б) – мощность «многие–ко–многим»

Идентифицирующая связь между сущностью-родителем и сущностью-потомком изображается сплошной линией (рисунок 3). Сущность-потомок в идентифицирующей связи является зависимой от идентификатора сущностью. Сущность-родитель в идентифицирующей связи может быть как независимой, так и зависимой от идентификатора сущностью (это определяется ее связями с другими сущностями).

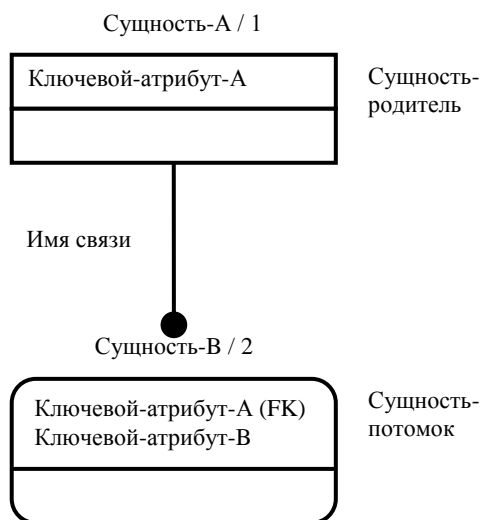


Рисунок 3 – Идентифицирующая связь

Пунктирная линия изображает неидентифицирующую связь (рисунок 4). Сущность-потомок в неидентифицирующей связи будет независимой от идентификатора, если она не является также сущностью-потомком в какой-либо идентифицирующей связи.

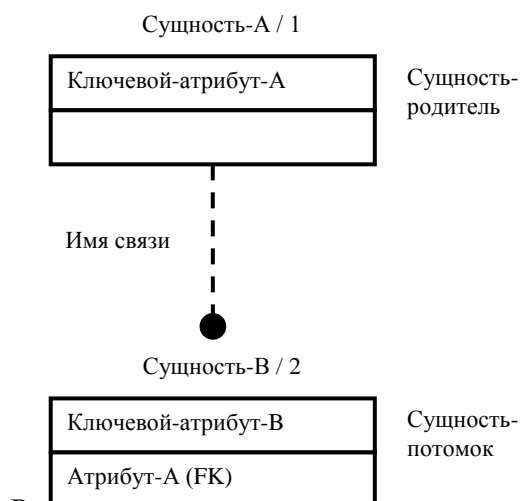


Рисунок 4 – неидентифицирующая связь

Атрибуты изображаются в виде списка имен внутри блока сущности. Атрибуты, определяющие первичный ключ, размещаются наверху списка и отделяются от других атрибутов горизонтальной чертой.

Сущности могут иметь также внешние ключи (Foreign Key), которые могут использоваться в качестве части или целого первичного ключа или неключевого атрибута. Внешний ключ изображается с помощью помещения внутрь блока сущности имен атрибутов, после которых следуют буквы FK в скобках.

Методология IDEF1X подразделяется на уровни, соответствующие проектируемой модели данных системы (рисунок 5). Каждый такой уровень соответствует определенной фазе проекта. Такой подход полезен при создании систем по принципу «сверху вниз».

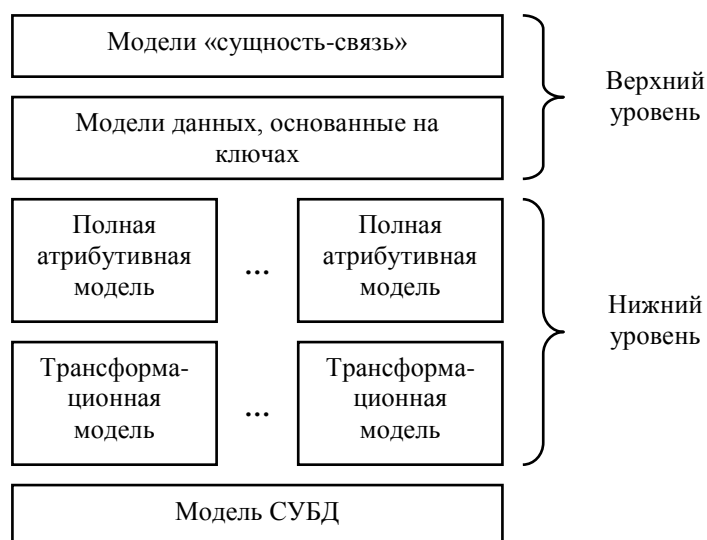


Рисунок 6 – Уровни методологии IDEF1X

Верхний уровень состоит из Entity Relation Diagram (Диаграмма сущность-связь) и Key-Based model (Модель данных, основанная на ключах). Диаграмма сущность-связь определяет сущности и их отношения. Модель данных, основанная на ключах, дает более подробное представление данных. Она включает описание всех сущностей и первичных ключей, которые соответствуют предметной области.

Нижний уровень состоит из Transformation Model (Трансформационная модель) и Fully Attributed (Полная атрибутивная модель). Трансформационная модель содержит всю информацию для реализации проекта, который может быть частью общей информационной системы и описывать предметную область. Трансформационная модель позволяет проектировщикам и администраторам БД представлять, какие объекты БД хранятся в словаре

данных, и проверить, насколько физическая модель данных удовлетворяет требованиям информационной системы. Фактически из трансформационной модели автоматически можно получить модель СУБД, которая является точным отображением системного каталога СУБД.

Диаграмма «сущность-связь» является самым высоким уровнем в модели данных и определяет набор сущностей и атрибутов проектируемой системы. Целью этой диаграммы является формирование общего взгляда на систему для ее дальнейшей детализации. Пример диаграммы «сущность-связь» для информационной системы проката фильмов приведен на рисунке 7. В данной системе присутствует восемь сущностей:

- клиент (CUSTOMER) – содержит сведения о человеке, заказывающем в прокат видеофильмы;
- фильм (MOVIE) – содержит описание фильма;
- копия фильма (MOVIE_COPY) – описывает копию фильма для проката;
- данные о прокате (MOVIE_RENTAL_RECORD) – содержит информацию о строках и условиях проката;
- оплата (PAYMENT) – содержит данные об оплате за прокат видеофильма;
- чек (CHECK) – содержит сведения об оплате банковским чеком;
- электронный платеж (E_PAYMENT) – содержит сведения об электронном платеже;
- кредитная карточка (CREDIT_CARD) – содержит сведения об оплате кредитной карточкой.

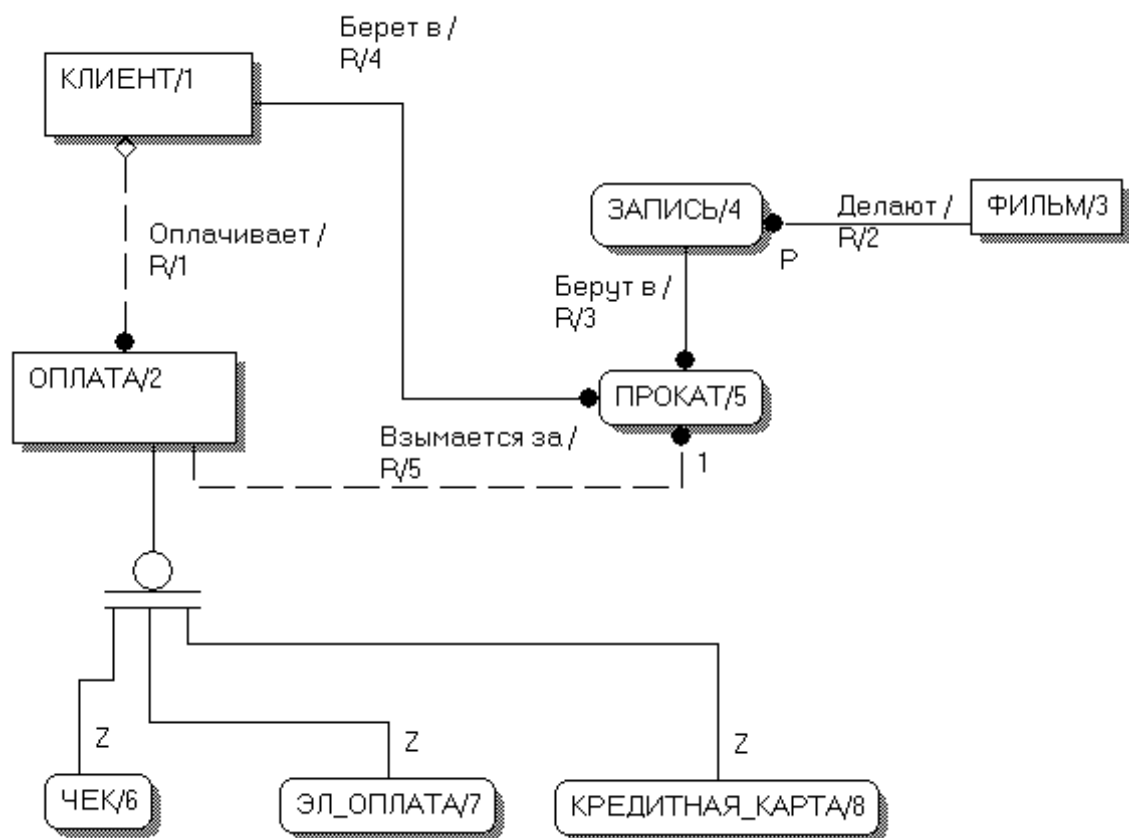


Рисунок 7 – Пример диаграммы «сущность-связь»

Модель данных, основанных на ключах, описывает структуру данных системы, в которую включены все сущности и атрибуты, в том числе ключевые. Целью этой модели является детализация модели сущность-связь, после чего модель данных может начать реализовываться.

Построение данной модели начинается с выбора первичного ключа, предназначенного для уникальной идентификации сущности. Атрибуты сущности образующие первичный ключ располагаются, как было указано ранее, над горизонтальной линией, разделяющей сущность. Также в методологии IDEF1X допускается определять альтернативные ключи – потенциальный

ключ, не ставший первичным. При создании альтернативного ключа на диаграмме рядом с атрибутом записываются символы (АК). Дополнительно, если в сущности присутствует несколько альтернативных ключей, то после символов АК может быть указан номер этого альтернативного ключа.

Атрибуты, участвующие в неуникальных индексах, называются инверсионными входами (Inversion Entries). Инверсионные входы – это атрибут или группа атрибутов, которые не определяют экземпляр уникальным образом, но часто используются для обращения к экземплярам сущности. Инверсионные входы на диаграмме обозначаются символами ИЕ, записанными справа от имени атрибута.

При проведении связи между двумя сущностями в дочерней сущности автоматически образуются внешние ключи (foreign key). Связь образует ссылку на атрибуты первичного ключа в дочерней сущности, и эти атрибуты образуют внешний ключ в дочерней сущности. Атрибуты внешнего ключа обозначаются символами (FK) после своего имени.

Полная атрибутивная модель включает в себя все сущности, атрибуты и является наиболее детальным представлением структуры данных. Полная атрибутивная модель представляет данные в третьей нормальной форме.

Пример полной атрибутивной модели, для рассмотренной ранее модели «сущность-связь» (рисунок 7), приведен на рисунке 8.

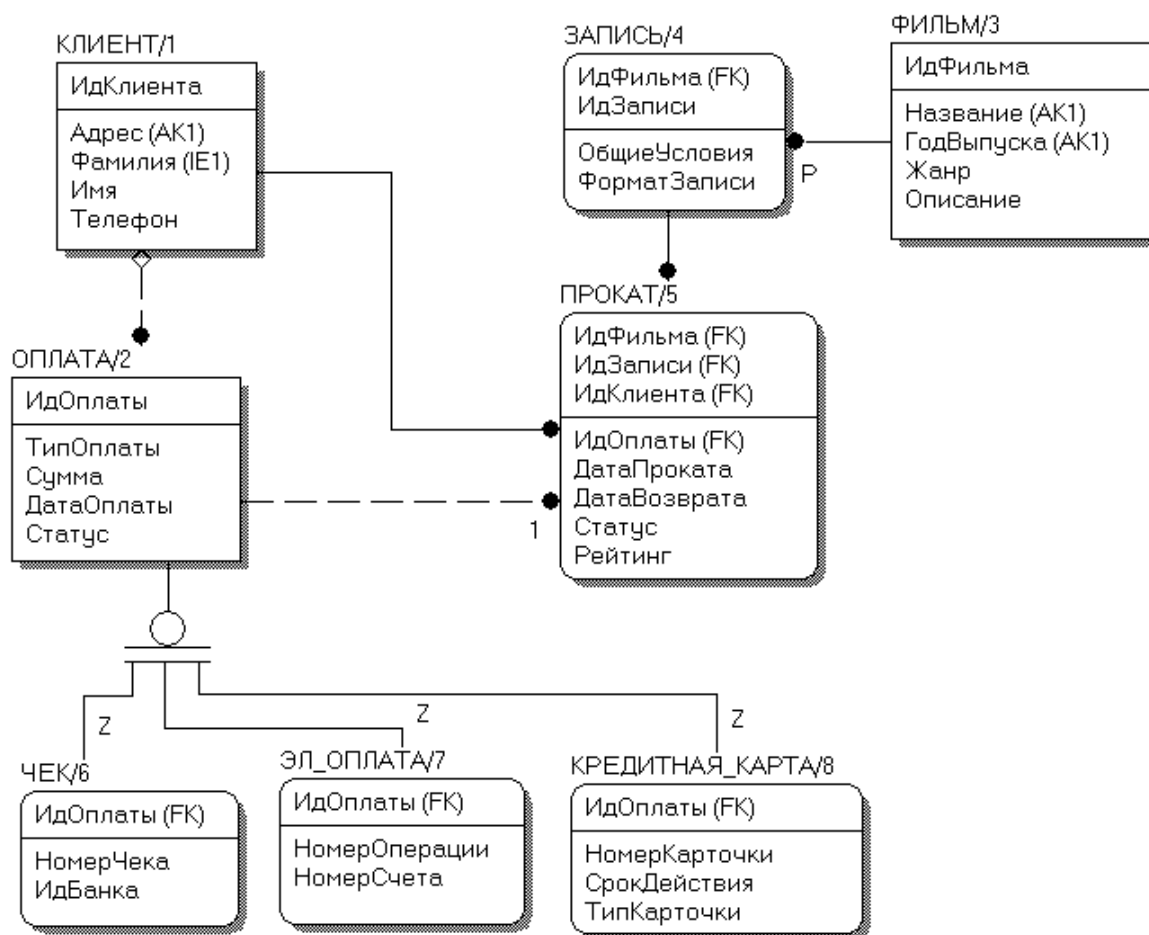


Рисунок 8 – Пример полной атрибутивной модели

Целью трансформационной модели является предоставление информации администратору БД для создания эффективной структуры хранения, включающей в себя записи, формирующие БД. Трансформационная модель должна помочь разработчикам выбрать структуру хранения данных и реализовать систему доступа к ним.

Перед началом проектирования БД необходимо убедиться в обеспечении следующих требований:

- физическая модель данных должна соответствовать требованиям, предъявляемым к проектируемой системе;
- выбор определенной физической модели должен быть аргументирован;
- должны быть определены возможности наращивания существующей структуры хранения, а также выявлены ее ограничения.

Модель СУБД напрямую транслируется из трансформационной модели, являясь отображением системного каталога. ERWin напрямую поддерживает эту модель через функцию генерации схемы БД. При составлении схемы БД в качестве индексов могут использоваться как ключевой атрибут, так и остальные поля БД. Пример модели СУБД приведен на рисунке 9.

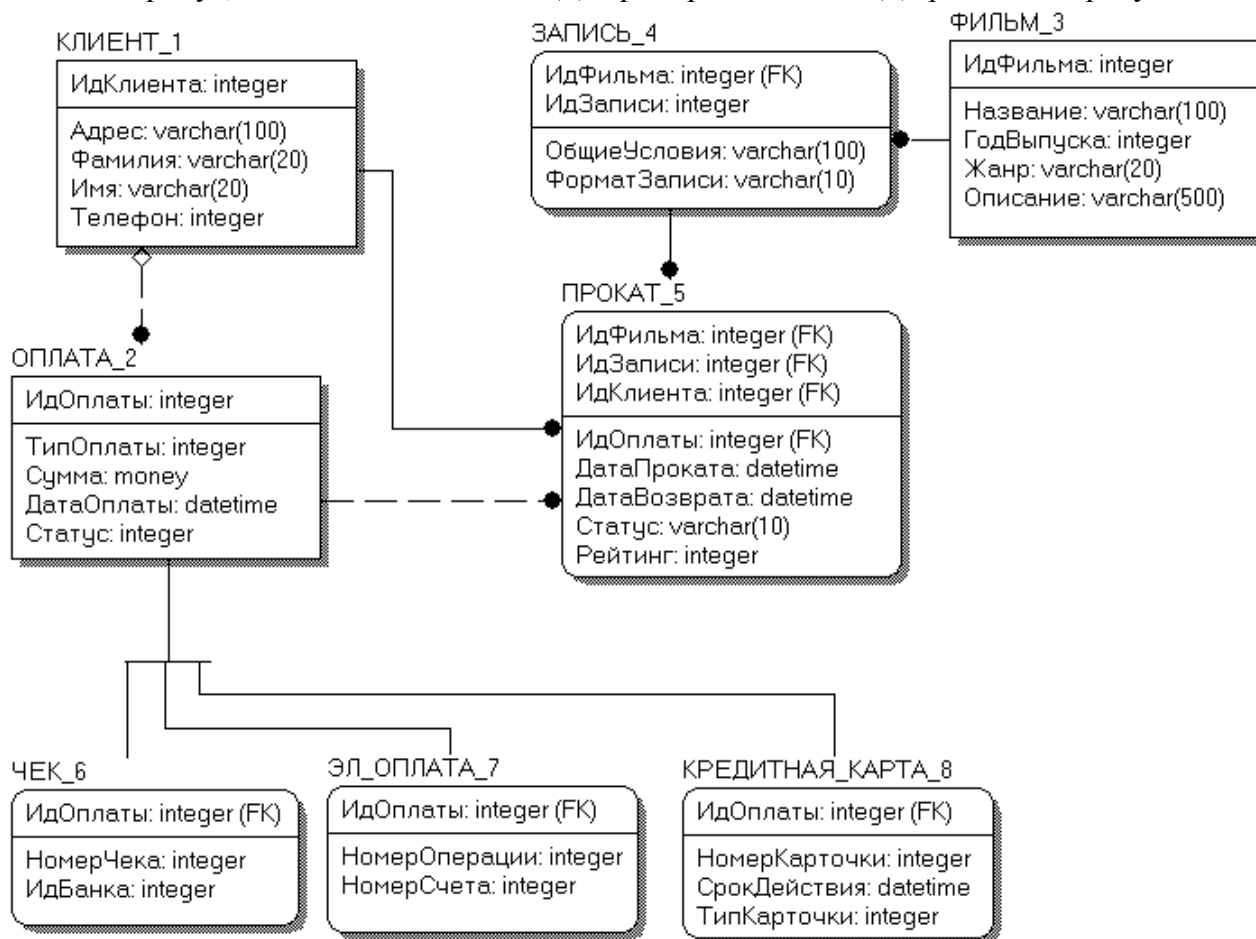


Рисунок 9 – Пример модели СУБД

Ход работы

1. Создать проект в CASE-средстве ERWin Data Modeller, включающий логическую и физическую модели (в качестве СУБД указать MS Access).
2. Выявить сущности предметной области проектируемой информационной системы.
3. Определить связи между выявленными сущностями предметной области.
4. Построить диаграмму сущность-связь согласно методологии IDEF1X.
5. Определить атрибуты сущностей и их типы.
6. Построить полную атрибутивную модель предметной области проектируемой информационной системы.
7. Произвести нормализацию полученной модели.
8. Построить физическую модель предметной области проектируемой информационной системы.
9. Определить и указать типы атрибутов согласно выбранной СУБД.
10. Трансформировать физическую модель в базу данных выбранной СУБД.

Контрольные вопросы

1. Для каких целей используется методология IDEF1X?
2. Что такое сущность и как она обозначается согласно методологии IDEF1X?
3. В чем заключается отличие зависимых и независимых сущностей в методологии IDEF1X?
4. Что такое связь и как она обозначается согласно методологии IDEF1X?
5. Какие виды связей могут использоваться в методологии IDEF1X?
6. В чем заключается отличие идентифицирующей и не идентифицирующей связей в методологии IDEF1X?
7. Опишите иерархию моделей IDEF1X.
8. Для чего предназначена и как осуществляется построение модели сущность-связь?
9. Для чего предназначена и как осуществляется построение модели, основанной на ключах?
10. Для чего предназначена и как осуществляется построение полной атрибутивной модели?
11. Для чего предназначена и как осуществляется построение трансформационной модели?
12. Для чего предназначена и как осуществляется построение модели СУБД?