

The logo for Oracle Academy is centered on a light gray background. It features the word "ORACLE" in a bold, orange, sans-serif font. Below it, the word "Academy" is written in a smaller, black, sans-serif font. The entire logo is framed by a thin black border, with dark gray horizontal bars at the top and bottom.

ORACLE
Academy

Database Foundations

3-4

Терминология моделирования данных и сопоставление

ORACLE
Academy



© 2020, Корпорация Oracle и аффилированные с ней лица. Все права защищены.

Дорожная карта

Дополнительные действия со связями

Отслеживание изменений данных

Нормализация и бизнес-правила

Терминология моделирования данных и сопоставление



Часть 3

ORACLE
Academy

DFo 3-4
Терминология моделирования данных и сопоставление

© 2020, Корпорация Oracle и аффилированные с ней лица. Все права защищены.

3

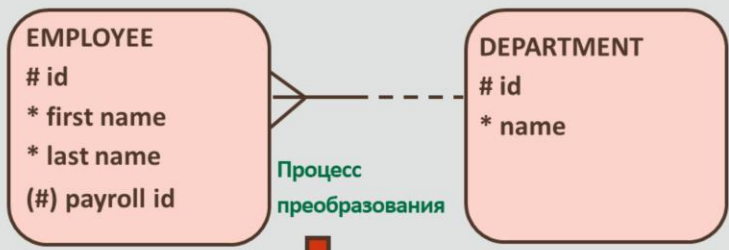
Цели

- Цели этого урока:
 - Применение сопоставления терминологии между логическими и физическими моделями
 - Изучение и применение правил именования Oracle для таблиц и столбцов, используемых в физических моделях
 - Применение правил сопоставления связей для правильного преобразования отношений



Преобразование логической модели в физическую: пример

Логическая модель (ERD)



EMPLOYEES (EPE)

Key Type	Optionality	Column name
pk	*	id
uk	*	payroll_id
	*	last_name
	*	first_name
fk	*	department_id

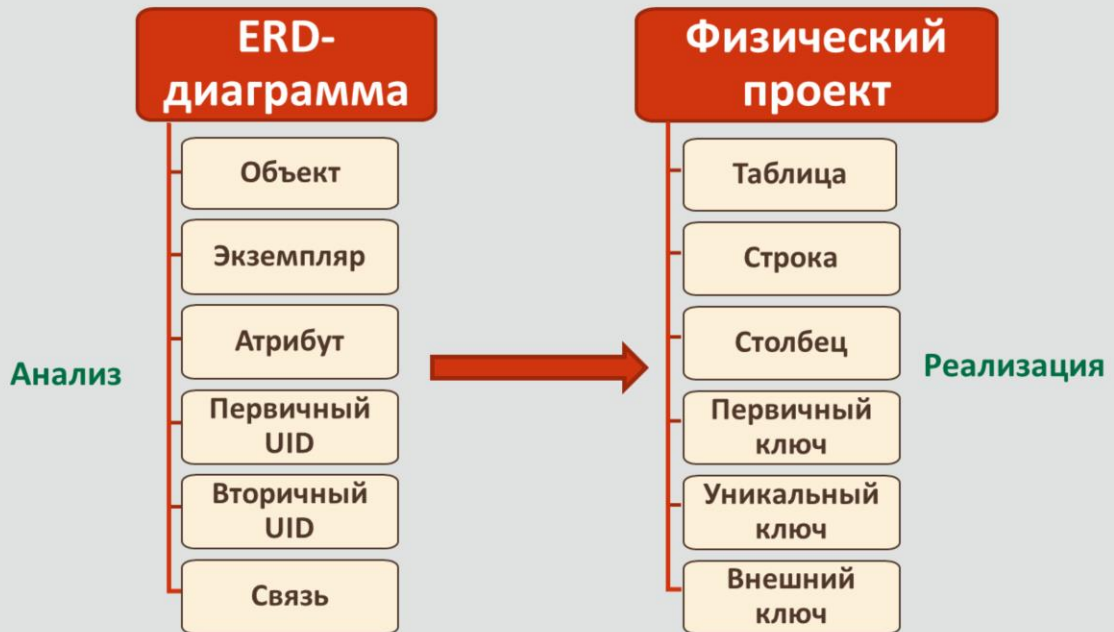
Реализация физической модели: Реляционная база данных

DEPARTMENTS (DPT)

Key Type	Optionality	Column name
pk	*	id
	*	department_name



Сопоставление терминологии



Объект и соответствующая таблица

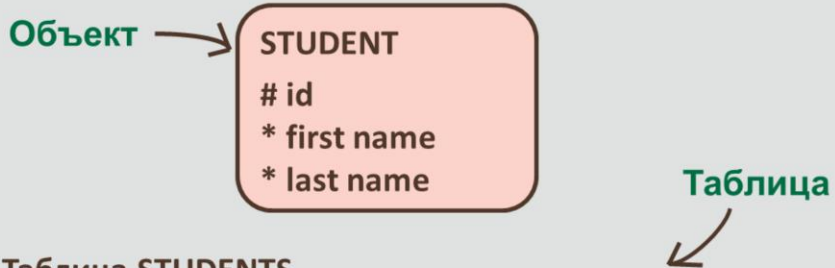
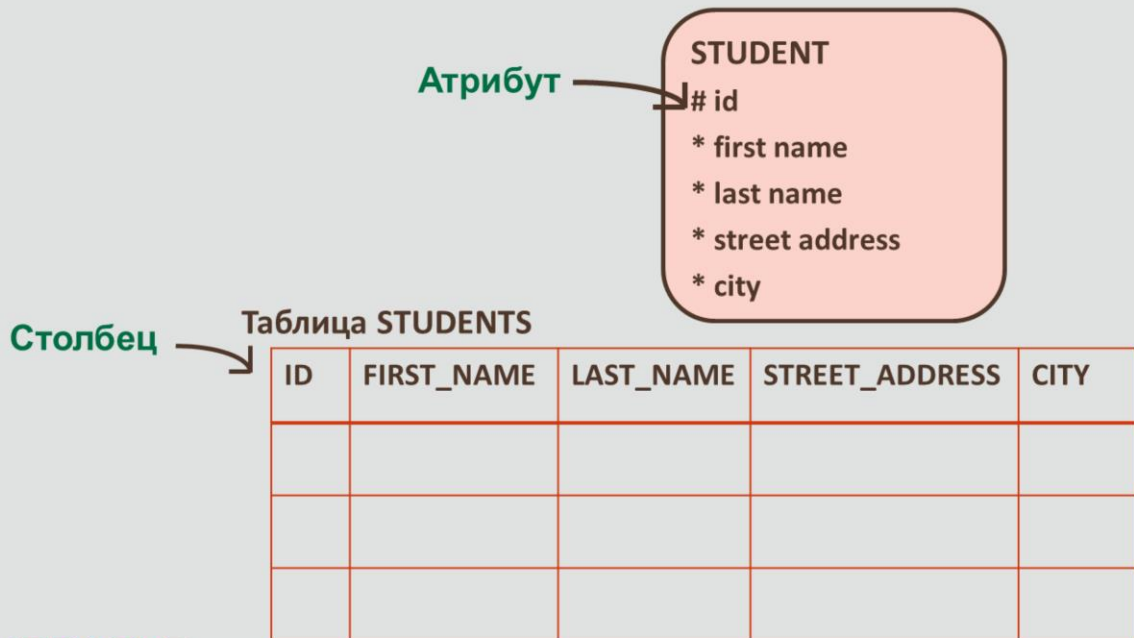


Таблица STUDENTS

ID	FIRST_NAME	LAST_NAME

Атрибуты и соответствующие столбцы



ORACLE
Academy

DFo 3-4
Терминология моделирования данных
и сопоставление

© 2020, Корпорация Oracle и аффилированные с ней лица. Все права защищены.

8

Атрибуты классифицируются следующим образом:

- **Ненулевой (обязательный):** обозначается звездочкой (*) рядом с атрибутом
- **Необязательный (может иметь значение null):** обозначается символом «o» рядом с атрибутом

Экземпляр и соответствующая строка

Объект

STUDENT

Экземпляр

J Smith

ID	FIRST_NAME	LAST_NAME	STREET_ADDRESS	CITY
101	Sam	Linkin	99B, Chuah Street	LA
102	Neena	Markin	44A, Church Street	NZ
103	Rick	Austina	1 st Cross, Palm Street	SA
104	J	Smith	Alpha Street	CA

→ **Строка**

Обозначения в диаграмме таблицы

- Диаграмма таблицы — это дополнительная документация, которая часто используется для более подробного объяснения ключей и столбцов в физической базе данных

Таблица STUDENTS

Key Type	Optionality	Column Name
pk	*	id
	*	first_name
	*	last_name
	*	street_address
	*	city

- Столбец «Key Type» (Тип ключа) должен содержать значения: pk для первичного ключа, uk для уникального ключа, fk для внешнего ключа. Эта ячейка остается пустой, если столбец не входит в состав ключа.
- Столбец «Optionality» (Обязательность) должен содержать звездочку (*), если этот столбец обязательный, или букву «o» в нижнем регистре, если он необязательный. Это аналогично диаграмме ERD.
- Третий столбец предназначен для имени столбца.

Правила именования таблиц

- Имя таблицы — это множественное число от имени объекта
 - Пример: STUDENT становится STUDENTS
- Имена столбцов аналогичны именам атрибутов, за тем исключением, что специальные символы и пробелы заменяются на подчеркивания

```
STUDENT
# id
* first name
* last name
* street address
* city
```



STUDENTS		
Key Type	Optionality	Column name
pk	*	id
	*	first_name
	*	last_name
	*	street_address
	*	city

Правила именования столбцов

- Имена столбцов аналогичны именам атрибутов, за тем исключением, что специальные символы и пробелы заменяются на подчеркивания
- В именах столбцов часто используется больше сокращений, чем в именах атрибутов.
 - Пример: First name становится first_name или fname.

STUDENT
id
* first name
* last name
* street address
* city



STUDENTS		
Key Type	Optionality	Column name
pk	*	id
	*	first_name
	*	last_name
	*	street_address
	*	city

Краткие имена таблиц

- Уникальное краткое имя для каждой таблицы полезно при именовании столбцов внешних ключей.



PRIVATE_HOMES (PHE)		
Key Type	Optionality	Column name
pk	*	id
	*	address
	o	comments

Краткие имена таблиц

- Создайте краткие имена на основе следующего:
 - Имена объектов, содержащие более одного слова
 - Имена объектов, содержащие одно слово, состоящее более чем из одного слога
 - Имена объектов, содержащие один слог, состоящий более чем из одного символа



PRIVATE_HOMES (PHE)		
Key Type	Optionality	Column name
pk	*	id
	*	address
	o	comments

Для имен объектов, содержащих более одного слова, нужно взять:

- первую букву первого слова;
- первую букву второго слова;
- последнюю букву последнего слова.

Пример: PRIVATE HOME получает краткое имя PHE.

Для имен объектов, содержащих одно слово более чем из одного слога, нужно взять:

- первую букву первого слога;
- первую букву второго слога;
- последнюю букву последнего слога.

Пример: EMPLOYEE получает краткое имя EPE, а CLIENT — CET.

Для имен объектов, содержащих один слог более чем из одного символа, нужно взять:

- первую букву;
- вторую букву;
- последнюю букву.

Пример: FLIGHT получает краткое имя FLT.

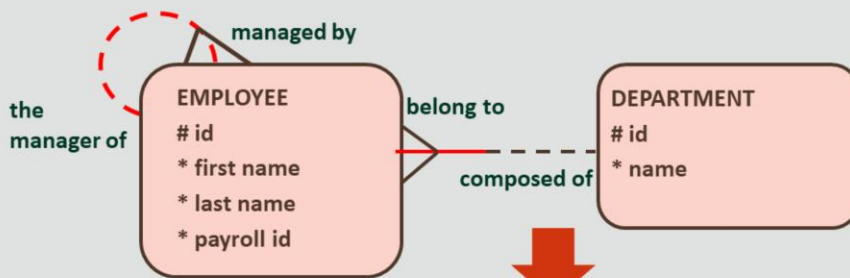
Ограничения именования в Oracle

- Имена таблиц и столбцов:
 - Должны начинаться с буквы
 - Могут содержать до 30 символов
 - Не могут содержать пробелы или специальные символы, например «!» (но «\$», «#» и «_» разрешены)
 - Не могут быть «зарезервированными словами» в БД Oracle или SQL
- Имена таблиц должны быть уникальными в пределах одной учетной записи пользователя в базе данных Oracle
- Имена столбцов в таблице должны быть уникальными

Сопоставление связей

- Связи сопоставляются между первичными ключами и внешними ключами, чтобы одна таблица могла ссылаться на другую
- Связь создает в таблице один или несколько столбцов внешнего ключа на стороне связи «многие»
- Для именованного столбца внешнего ключа используется краткое имя таблицы
- В примере на следующих страницах столбец внешнего ключа в таблице EMPLOYEES называется `dpt_id` для связи DEPARTMENT и `mgr_id` для рекурсивной связи с самим собой

Иллюстрация сопоставления связей



EMPLOYEES (EPE)

Key Type	Optionality	Column name
pk	*	id
	*	first_name
	*	last_name
uk	*	payroll_id
fk1	*	dpt_id
fk2	o	mgr_id

DEPARTMENTS (DPT)

Key Type	Optionality	Column name
pk	*	id
uk	*	name

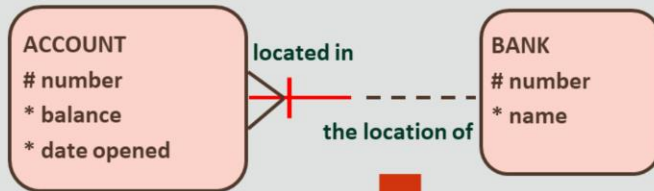
внешний ключ ссылается на

внешний ключ ссылается на

Сопоставление ассоциативных (barred) связей

- Ассоциативная связь сопоставлена со столбцом внешнего ключа на стороне «многие», как любая другая связь M:1
- В этом случае столбец внешнего ключа играет двойную роль, так как он также входит в состав первичного ключа
- В этом примере `bak_number` — столбец внешнего ключа в `ACCOUNTS`, который относится к первичному ключу `BANKS`
- Он также входит в состав первичного ключа `ACCOUNTS`

Сопоставление ассоциативных (barred) связей



ACCOUNTS (ACT)

Key Type	Optionality	Column name
pk	*	number
	*	balance
	*	date_opened
pk,fk	*	bak_nbr

BANKS (BAK)

Key Type	Optionality	Column name
pk	*	number
	*	name

внешний ключ ссылается на

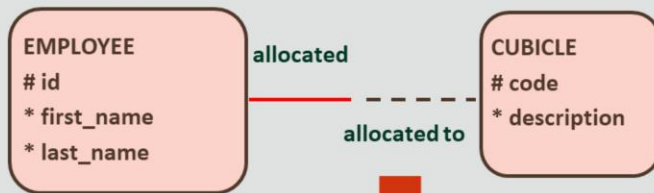
Сопоставление связей «многие ко многим»

- Связь M:M решается объектом пересечения, который сопоставляется с таблицей пересечения
- Эта таблица пересечения будет содержать столбцы внешнего ключа, связанные с исходными таблицами
- В этом примере REVIEWS содержит все сочетания, которые существуют между объектами CRITIC и MOVIE

Сопоставление связей «один к одному»

- При преобразовании связи 1:1 создается внешний ключ и уникальный ключ
- Все столбцы этого внешнего ключа также входят в состав уникального ключа
- Если связь обязательная с одной стороны, в соответствующей таблице создается внешний ключ
- В примере `cbe_code` — столбец внешнего ключа для EMPLOYEES, который относится к первичному ключу CUBICLES
- `Cbe_code` также уникален в таблице EMPLOYEES

Сопоставление связей «один к одному»



EMPLOYEES (EPE)

Key Type	Optionality	Column name
pk	*	id
	*	first_name
	*	last_name
fk,uk	*	cbe_code

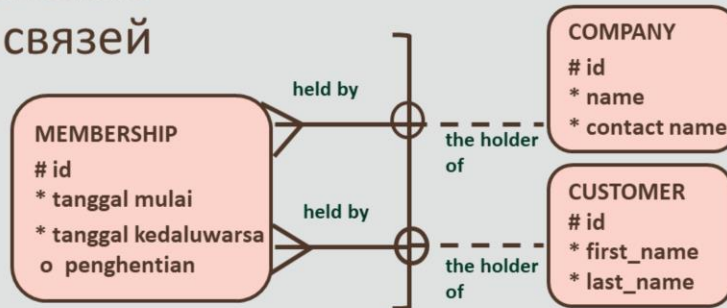
CUBICLES (CBE)

Key Type	Optionality	Column name
pk	*	code
	*	description

Сопоставление дуговых связей

- Объект, имеющий дугу, будет сопоставлен с таблицей, содержащей внешние ключи из таблиц на конце связей «один»
- Обратите внимание: даже хотя связи в дуге обязательные на стороне «многие», итоговые внешние ключи должны быть необязательными (так как один из них всегда будет пустым)
Это может обеспечить ограничение проверки значений, хранящихся в базе данных

Сопоставление дуговых связей



MEMBERSHIPS (MBP)

Key Type	Optionality	Column name
pk	*	id
	*	start_date
	*	expiration_date
	o	termination
fk1	o	cpe_id
fk2	o	cms_id

COMPANIES (CPE)

Key Type	Optionality	Column name
pk	*	id
	*	name
	*	contact_name

CUSTOMERS (CMS)

Key Type	Optionality	Column name
pk	*	id
	*	first_name
	*	last_name

ORACLE
Academy

DFo 3-4
Терминология моделирования данных
и сопоставление

© 2020, Корпорация Oracle и аффилированные с ней лица. Все права защищены.

25

Так как дуга представляет исключаящую связь, требуется дополнительный код, который принудительно обеспечит только для одного из внешних ключей наличие значения для каждой строки в таблице.

Это может обеспечить ограничение проверки значений, хранящихся в базе данных.

В примере код для ограничения проверки будет выглядеть так:

- CHECK (cpe_id is not null AND cms_id is null)
- OR (cpe_id is null AND cms_id is not null)

Если связи полностью необязательные, можно добавить следующее: OR (cpe_id is null AND cms_id is null)

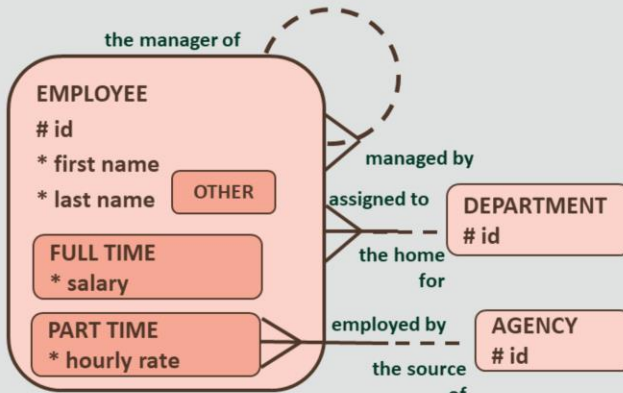
Сопоставление супертипа/подтипов

- Объекты-супертипы и объекты-подтипы можно сопоставлять несколькими способами
 - Реализация в одной таблице: создается одна таблица, независимо от количества подтипов. Используется, когда большинство атрибутов и связей общие и поэтому находятся на уровне супертипа
 - Реализация в двух таблицах: для каждого подтипа создается по таблице (поэтому может быть больше двух таблиц). Используется, когда у подтипов мало общего, мало общих атрибутов и связей

Реализация в одной таблице

- Одна таблица содержит по одному столбцу для каждого атрибута супертипа, а также исходный необязательный характер атрибута
- Также таблица содержит столбец для каждого атрибута, принадлежащего подтипу, но все столбцы становятся необязательными
- Кроме того, необходимо создать обязательный столбец, который будет выступать столбцом-дискриминатором для различения разных типов подтипов объекта

Реализация в одной таблице



DEPARTMENTS (DPT)

Key Type	Optionality	Column name
pk	*	id

AGENCIES (AGY)

Key Type	Optionality	Column name
pk	*	id

EMPLOYEES (EPE)

Key Type	Optionality	Column name
pk	*	id
	*	first_name
	*	last_name
	o	salary
	o	hourly_rate
fk1	*	dpt_id
fk2	o	agy_id
	*	epe_type
fk3	o	mgr_id

ORACLE
Academy

DFo 3-4
Терминология моделирования данных
и сопоставление

© 2020, Корпорация Oracle и аффилированные с ней лица. Все права защищены.

28

Идентификаторы: уникальные идентификаторы преобразуются в первичные и уникальные ключи.

Связи: связи на уровне супертипа преобразуются как обычно. Связи на уровне подтипа реализуются как необязательные столбцы внешнего ключа.

Ограничения целостности: необходимо ограничение проверки для обеспечения того, что для каждого конкретного подтипа все столбцы, которые получаются из обязательных атрибутов, будут не нулевыми.

В логической модели жалование обязательное для сотрудников с полной занятостью, а почасовая ставка обязательная для сотрудников с частичной занятостью.

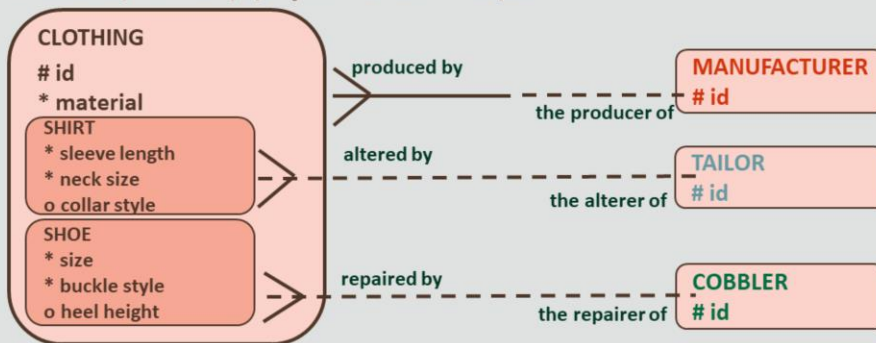
- Когда супертип EMPLOYEE реализуется в физической модели в одной таблице, эти атрибуты становятся необязательными.

- Ограничение проверки необходимо для применения бизнес-правил, смоделированных в ERD.

Реализация в двух таблицах

- Одна таблица предназначена для одного подтипа первого уровня
- Каждая таблица содержит по одному столбцу для каждого атрибута супертипа, а также его исходный необязательный характер
- Каждая таблица также содержит по одному столбцу для каждого атрибута подтипа и его исходный необязательный характер

Реализация в двух таблицах



SHIRTS (SHT)

Key Type	Optionality	Column name
pk	*	id
	*	material
	*	sleeve_length
	*	neck_size
	o	collar_style
fk1	o	tlr_id
fk2	*	mnr_id

SHOES (SHE)

Key Type	Optionality	Column name
pk	*	id
	*	material
	*	size
	*	buckle_style
	o	heel_height
fk1	o	clr_id
fk2	*	mnr_id

ОТНОСИТСЯ К
tailors

ОТНОСИТСЯ К
manufacturers

ОТНОСИТСЯ К
cobblers

ORACLE
Academy

DFo 3-4
Терминология моделирования данных
и сопоставление

© 2020, Корпорация Oracle и аффилированные с ней лица. Все права защищены.

30

Идентификаторы: первичный UID на уровне супертипа создает первичный ключ для каждой таблицы. Вторичные UID супертипа становятся уникальными ключами в каждой таблице.

Связи:

- Все таблицы получают внешний ключ для связи на уровне супертипа с сохранением исходного необязательного характера (manufacturers).
- Для связей на уровне подтипа внешний ключ реализуется в таблице, с которой он сопоставлен, с сохранением исходного необязательного характера (tailors, cobblers).

Упражнение по проекту

- DFo_3_4_Project

- База данных магазина Oracle Baseball League применяет правила сопоставления связей для преобразования связей



Сводка

- В этом уроке вы узнали следующее:
 - Применение сопоставления терминологии между логическими и физическими моделями
 - Изучение и применение правил именования Oracle для таблиц и столбцов, используемых в физических моделях
 - Применение правил сопоставления связей для правильного преобразования отношений



The logo for Oracle Academy is centered on a light gray background. It features the word "ORACLE" in a bold, orange, sans-serif font. Below it, the word "Academy" is written in a smaller, black, sans-serif font. The entire logo is framed by a thin black border, with dark gray horizontal bars at the top and bottom of the frame.

ORACLE
Academy