***15.1. Перепроектирование БД используя Oracle SQL Developer Data Modeler 3.1***

В главе использованы материалы [[apex\_lab1]](http://enisey.name/umk/upr21/ch15s07.html#apex_lab1).

Требования:

Иметь установленное приложение Oracle SQL Developer Data Modeler 3.1

Иметь доступ к Oracle Database 11g или 10g

***15.1.1. Установка  Oracle SQL Developer Data Modeler***

1. Устанавливаем JDK: см. [[MIND]](http://enisey.name/umk/upr21/ch15s07.html#MIND), 3.2-3.3.

Для Linux

2. Устанавливаем  Oracle SQL Developer Data Modeler: см. [[MIND]](http://enisey.name/umk/upr21/ch15s07.html#MIND), 3.4-3.5.

datamodeler-3.1.1-703.zip

3. Устанавливаем  Oracle SQL Developer: см. [[MIND]](http://enisey.name/umk/upr21/ch15s07.html#MIND), 3.6-3.7.

sqldeveloper-3.1.07.42.zip

Если выдает ошибку “cannot access nls data files or invalid environment specified”  добавляем “AddVMOption -Duser.region=US” во все файлы “sqldeveloper.conf“

4. Устанавливаем  Oracle 10g XE: см. [[MIND]](http://enisey.name/umk/upr21/ch15s07.html#MIND), 3.8.

***15.1.2. Импорт таблиц из HR схемы***

В этом разделе создадим соединение с HR схемой.  Используя это соединение импортируем таблицы EMPLOYEES и DEPARTMENTS  в SQL Developer Data Modeler.

1. Запустите Oracle SQL Developer Data Modeler: выберите Пуск > Программы > DataModeler (или откройте каталог С:\datamodeler, выберите datamodeler.exe, М2).

2. Импортируем таблицы из словаря данных, выберите File > Import > Data Dictionary.

3. Окно Data Dictionary Import Wizard открыто. Создадим соединение с БД, выберите Add.

4. Окно New / Update Database Connection открыто. Введите следующие параметры соединения  c локальной СУБД или серверной СУБД и выберите Test.

|  |  |
| --- | --- |
| Локальная СУБД | Серверная СУБД |
| Connection Name: hr  User Name: hr  Password: hr  Hostname: localhost  SID: xe | Connection Name: hr252  User Name: hr  Password: hr  Hostname: 192.168.100.252  SID: xe |

5. Статус должен быть Success. Выберите Connect.

6. В окне Data Dictionary Import Wizard выберите hr соединение  и нажмите Next.

7. Импортируйте из HR схемы: выберите HR схема и нажмите Next.

8. Выберите таблицы DEPARTMENTS и EMPLOYEES и нажмите  Next.

9. Нажмите Finish для генерация модели.

10. Просмотрите log файл создания и нажмите Close.

11. EMPLOYEES и DEPARTMENTS таблицы удачно импортированы в реляционную модель SQL Developer Data Modeler.

**Рисунок 15.1. Таблицы EMPLOYEES и DEPARTMENTS импортированы в реляционную модель SQL Developer Data Modeler**

|  |
| --- |
| Таблицы EMPLOYEES и DEPARTMENTS импортированы в реляционную модель SQL Developer Data Modeler |

***15.1.3. Просмотр физической модели***

В этом разделе просмотрим физичекую модель автоматически созданную при импорте  объектов из словаря данных.

13. В Nagivation Browser раскройте Relational Models > Relational\_1.

14. Вы видите физическую модель. Раскройте Physical Models > Oracle Database 10g.

15. Раскройте Tables > HR.DEPARTMENTS > Columns. Просмотрите список. Обратите внимание на другие объекты представленные в физической модели.

**Рисунок 15.2. Физическая модельв SQL Developer Data Modeler**

|  |
| --- |
| Физическая модельв SQL Developer Data Modeler |

***15.1.4. Создание таблицы***

В этом разделе создадим таблицу PROJECTS.  В таблице  PROJECTS четыре столбца: PROJECT\_ID, PROJECT\_NAME, PROJECT\_START\_DATE, и PROJECT\_END\_DATE.

16. Выберите New Table в toolbar.

17. Выберите незанятое пространство на диаграмме.

18.  Окно Table Properties открыто. Введите PROJECTS для Name и выберите Columns в левом навигаторе.

19. Добавим столбец, выберите Create Column.

20. Введите PROJECT\_ID для Name. Выберите Logical для Datatype и NUMERIC для Type.

21. Введите 6 для Precision. Определим PROJECT\_ID столбец как первичный ключ. Выберите PK и нажмите Create Column.

22. Введите PROJECT\_NAME для Name. Выберите Logical для  Datatype иVARCHAR для Type.

23. Введите 100 для Size и нажмите Create Column.

24. Введите PROJECT\_START\_DATE для Name. Выберите  Logical для Datatype и Date для Type. Выберите Create Column.

25. Введите PROJECT\_END\_DATE для Name. выберите Logical для Datatype и Date для Type. Нажмите  OK  для создания таблицы.

26. Таблица удачно создана.

**Рисунок 15.3. Создание таблицы в SQL Developer Data Modeler**

|  |
| --- |
| Создание таблицы в SQL Developer Data Modeler |

***15.1.5. Соединение новой таблицы с выходной таблицей***

В этом разделе создадим внешний ключ между таблицами PROJECTS и DEPARTMENTS.

27. Выберите New FK Relation в toolbar.

28. Создадим внешний ключ в таблице PROJECTS, выберите таблицу DEPARTMENTS  и выберите таблицу PROJECTS.

29. Окно Foreign Key Properties открыто. Выберите Associated Columns в левом навигаторе.

30. DEPARTMENTS\_DEPARTMENT\_ID выберите для Child Column и нажмите OK.

31. Выберите Arrow  в toolbar. FK Relation создана. Буква 'F' напротив DEPARTMENTS\_DEPARTMENT\_ID в таблице PROJECTS определяет этот столбец как внешний ключ.

32. Сделаем FK необязательным, выберите  линию соединения двух таблиц, МП, выберите Properties.

33. Отключите Mandatory и нажмите OK.

34. Выберите OK.

35. Нажмите  Yes.

36. FK теперь необязателен. Это означает, что можно иметь проект без отдела.

**Рисунок 15.4. Соединение таблиц в SQL Developer Data Modeler**

|  |
| --- |
| Соединение таблиц в SQL Developer Data Modeler |

***15.1.6. Модификация выходной таблицы***

В этом разделе добавим столбец COST\_CENTER в таблицу DEPARTMENTS.

37. На диаграмме выберите таблицу DEPARTMENTS. М2.

38. Выберите Columns в левом навигаторе.

39. Создадим новый столбец, выберите Create Column.

40. Введите COST\_CENTER для Name, выберите Logical для  Datatype, VARCHAR для  Type и введите 25 для Size. Нажмите OK для создания столбца.

41. Столбец COST\_CENTER создан в таблице DEPARTMENTS.

**Рисунок 15.5. Создание столбца в SQL Developer Data Modeler**

|  |
| --- |
| Создание столбца в SQL Developer Data Modeler |

***15.1.7. Создание логической модели***

В этом разделе перепроектируем реляционную модель созданием логической модели.

42. Выберите Engineer to Logical Model в toolbar.

43. Окно Engineer to Logical Model открыто. Раскройте Tables. Просмотрите сущности которые будут созданы в логической модели. Нажмите Engineer.

44. Логическая модель создана. Просмотрите сущности и атрибуты. Диаграмма логической модели называется Entity Relationship Diagram (ERD).

**Рисунок 15.6. Создание логической модели в SQL Developer Data Modeler**

|  |
| --- |
| Создание логической модели в SQL Developer Data Modeler |

***15.1.8. Модификация логической модели***

В этом разделе модифицируем логическую модель.

45. Добавим атрибуты в сущность PROJECTS. Выберите сущность  PROJECTS, М2.

46. Выберите Attributes в левом навигаторе.

47. Выберите  '+' для добавления атрибута.

48. Введите STATUS для Name, выберите Logical для Datatype, выберите VARCHAR для Type, и введите 30 для Size. Выберите  атрибут DEPARTMENT\_DEPARTMENT\_ID из списка.

49. Этот атрибут показывает результат связи между сущностями DEPARTMENTS и PROJECTS. Заметьте, что мы не можем изменить имя этого атрибута. Для этого необходимо изменить параметры настройки. Нажмите OK.

50. Заметьте, что атрибут STATUS добавленный в сущность PROJECTS  для связи с DEPARTMENT не показывается на диаграмме. Это происходит так как текущая нотация Barker не показывает такие атрибуты.

51. Изменим значение по умолчанию для атрибутов связи. Выберите Tools > Preferences...

52. Раскройте Data Modeler > Model и выберите Logical. Отключите Keep as the name of the Originating attribute.

53. Нажмите OK.

54. Выберите сущность PROJECTS, М2.

55. Выберите Attributes.

56. Выберите атрибут DEPARTMENT\_DEPARTMENT\_ID из списка.

57. Заметьте, что теперь можно изменять имя. Измените имя на PROJECT\_DEPT и нажмите OK.

58. Изменим нотацию на Bachman. Выберите свободное пространство на диаграмме, МП,  выберите Bachman Notation.

59. Нотация изменена. Теперь мы видим атрибуты связи (с буквой  'F') как и первичные атрибуты. Символ  '\*' перед именем атрибута означает обязательность атрибута.

60. Добавим сущность в логическую модель. Выберите New Entity icon в toolbar.

61. Выберите свободное пространство на диаграмме.

62. Введите TASKS для Name и выберите Attributes.

63. Выберите  '+' для добавления атрибута.

64. Введите TASK\_ID для Name, выберите Logical для Datatype, выберите NUMERIC для Type, введите 6 для Precision и выберите Primary UID.  Выберите '+'.

65. Введите TASK\_NAME для Name, выберите Logical для Datatype, выберите VARCHAR для Type и введите 255 для Size. Выберите '+'.

66. Введите COST для Name, выберите Logical для Datatype, выберите NUMERIC для Type и введите 6 для Precision. Выберите '+'.

67. Введите BUDGET для Name, выберите Logical для Datatype, выберите NUMERIC для Type и введите 6 для Precision и нажмите OK для создания сущности и атрибутов.

68. Создадим две связи между EMPLOYEES и TASKS, и между the PROJECTS и TASKS сущностями. Выберите New 1:N Relation в  toolbar.

69. Выберите сущность EMPLOYEES и выберите сущность TASKS.

70. Нажмите OK для создания связи.

71. Создадим вторую связь. Выберите сущность PROJECTS и выберите сущность TASKS.

72. Нажмите OK для создани ясвязи.

73. Изменим имя связи в сущности TASKS. Выберите сущность TASKS, М2.

74. Выберите Attributes свойства.

75. выберите атрибут EMPLOYEES\_EMPLOYEE\_ID из списка.

76. Измените имя на ASSIGNED\_TO и выберите атрибут PROJECTS\_PROJECT\_ID из списка.

77. Изменим имя на PROJECT\_ID и нажмите OK.

78. Изменение логической модели закончено.

**Рисунок 15.7. Создание логической модели в SQL Developer Data Modeler**

|  |
| --- |
| Создание логической модели в SQL Developer Data Modeler |

***15.1.9. Синхронизация логической модели с реляционной моделью***

В этом разделе синхронизируем логическую модель с реляционной моделью.

79. Выберите Engineer to Relational Model.

80. Окно Engineer to Relational Model открыто. Раскройте Entities > PROJECTS > Attributes. Обратите внимание на различия. Нажмите Engineer.

81. Обратите внимание на новый столбец STATUS, измененное имя внешнего ключа PROJECT\_DEPT, и новую таблицу TASKS.

**Рисунок 15.8. Синхронизация логической модели с реляционной моделью в SQL Developer Data Modeler**

|  |
| --- |
| Синхронизация логической модели с реляционной моделью в SQL Developer Data Modeler |

***15.1.10. Генерация DDL обновления БД***

Теперь модели готовы и можно синхронизировать словарь данных с моделью и сгенерировать DDL-скрипт.

82. Выберите Synchronize Data Dictionary with Model в toolbar.

http://enisey.name/umk/upr21/img/apex9.png83. Окно Compare Models открыто. Узлы где есть изменения отмечены символом( ). Раскройте Tables узел и увидите новые таблицы PROJECTS и TASKS. Обратите внимание что таблица DEPARTMENTS изменена.

84. Раскройте DEPARTMENTS > Columns. Столбец COST\_CENTER добавлен. Нажмите DDL Preview.

85. Сгенерированный код  выведен. Просмотрите команды  CREATE и ALTER. Заметьте, что таблицы PROJECTS и TASKS создаются и дополняются первичными ключами.

86. Ниже можно увидить новый столбец внешнего ключа. Нажмите Save.

87. Можно запускать этот скрипт в SQL Developer для обновления БД.

88. Сохраним файл. Нажмите Save.

89. Выберите Close для закрытия окна Compare Models.

90. Сохраним модель. Выберите File > Save.

91. Сохраним файл. Выберите Save.

92. Закройте SQL Developer Data Modeler. Выберите File > Exit.

**Рисунок 15.9. Выход из SQL Developer Data Modeler**

|  |
| --- |
| Выход из SQL Developer Data Modeler |

***15.1.11. Итого***

А разделе изучено:

Импорт таблиц из словаря данных

Просмтр физической модели

Создание новой таблицы и соединение с другой таблицей

Создание нового столбца

Перепроектирование для создания логической модели

Модификация логической модели

Синхронизация логической модели с реляционной

Генерация DDL для обновления БД

***15.2. Тестирование и отладка процедур с помощью SQL Developer 3.1***

В главе использованы материалы [[apex\_lab2]](http://enisey.name/umk/upr21/ch15s07.html#apex_lab2).

Требования:

1. Иметь установленное приложение Oracle SQL Developer 3.1
2. Иметь доступ к Oracle Database 11g или 10g
3. Grant HR user DEBUG CONNECT SESSION and DEBUG ANY PROCEDURE privileges.
4. Загрузить архив files.zip

***15.2.1. Создание соединения БД***

В этом разделе создадим соединение с HR схемой в SQL Developer.

1. Запустите Oracle SQL Developer: выберите Пуск | Программы | sqldeveloper (или откройте каталог С:\sqldeveloper, выберите sqldeveloper.exe, М2)..

2. Первое после запуска SQL Developer появляется окно Tip of the Day. Можно отключить его появление.

3. Выберите вкладкуConnections, МП и выберите New Connection.

4. Окно New / Update Database Connection открыто. Введите следующие параметры соединения  c локальной СУБД или серверной СУБД и выберите Test.

|  |  |
| --- | --- |
| Локальная СУБД | Серверная СУБД |
| Connection Name: hr  User Name: hr  Password: hr  Hostname: localhost  SID: xe | Connection Name: hr252  User Name: hr  Password: hr  Hostname: 192.168.100.252  SID: xe |

5. Статус должен быть Success. Выберите Connect.

6. Раскройте hr соединение. Обратите внимание на объекты разных типов. Раскройте Tables.

**Рисунок 15.10. Объекты в SQL Developer**

|  |
| --- |
| Объекты в SQL Developer |

|  |  |  |
| --- | --- | --- |
| [Пред.](http://enisey.name/umk/upr21/ch15.html) | [Уровень выше](http://enisey.name/umk/upr21/ch15.html) | [След.](http://enisey.name/umk/upr21/ch15s02s02.html) |
| Глава 15. Лабораторная работа «Разработка web-приложения в среде Oracle Application Express» | [Начало](http://enisey.name/umk/upr21/index.html) | [ToC](http://enisey.name/umk/upr21/bk01-toc.html) | 15.2.2. Просмотр объектов в HR схеме |

## *15.3. Настройка APEX*

### *15.3.1. Вступление о APEX*

APEX является бесплатным продуктом, интегрированным с СУБД Oracle Database.

Изначально APEX предназначался для создания HTML-интерфейса к базе данных. В настоящее время выпущена 4-я версия продукта, который стал полноценной средой проектирования и разработки web-приложений любой сложности с интегрированной БД. На базе APEX и бесплатной редакции Oracle Database eXpress Edition (XE) можно создавать сайты и порталы, которые не требуют затрат на лицензирование.

Характерной особенностью этой среды разработки является то, что для работы с ней не требуется высокой квалификации в web-программировании и HTML-верстке. APEX представляет собой конструктор готовых блоков сайта. Фактически, минимально подготовленный пользователь может создавать рабочие сайты со встроенными средствами аутентификации и безопасности, современным дизайном и интерфейсом. С другой стороны, это гибкий инструмент, и квалифицированный разработчик может создавать страницы и сайты любого дизайна и структуры.

Немаловажным является и то, что работоспособность этого сайта будет поддерживаться мощной и надежной базой данных Oracle Database. Сайты и порталы, разработанные на APEX, способны обслуживать сотни пользователей, т. е. отвечают требованиям, предъявляемым по масштабируемости к Интернет-приложениям [14].

В состав APEX входят следующие четыре основных компонента.

1. Application Builder – собственно среда разработки web-страниц и бизнес-правил.
2. SQL Workshop – среда управления объектами базы данных (индексы, таблицы, представления и т. п.). Включает мастер создания SQL запросов для пользователей, которые не обладают знаниями в языке SQL.
3. Utilities – импорт и экспорт данных, генерация SQL-скриптов на изменение структуры базы данных, отчеты и восстановление удаленных объектов.
4. Administration – управление пользователями, настройками, правами доступа и просмотр отчетов.

APEX включает в себя следующие возможности:

1. cреда разработки имеет простой и эффективный web-интерфейс, т. е. для начала разработки не требуется специализированных сред, разработка может вестись с любого компьютера с web-браузером;
2. помощники миграции из настольных баз данных и электронных таблиц;
3. встроенный мастер генерации отчетов в формате pdf;
4. инструменты для интеграции и web-сервисами;
5. большое количество шаблонов пользовательского интерфейса;
6. интуитивно-понятное управление рабочим пространством;
7. управление объектами по принципу drag & drop;
8. графический помощник создания SQL-запро сов;
9. защищенность данных сессии после авторизации пользователя;
10. встроенный редактор PL/SQL;
11. мастер создания диаграмм и отчетов на сайте;
12. поддержка более 20 языков, включая русский.

APEX является кросс-платформенной системой, т. е. он успешно работает как на операционной системе Windows, так и на Linux, Solaris, HP-UX, MAC OS и других.

Одним из простейших примеров применения APEX на предприятии является переход от настольных баз данных и электронных таблиц (например, MS Access, MS Excel) к web-представлению этих баз и документов. Это бывает очень полезным, когда необходимо обеспечить одновременный доступ для редактирования одного и того же документа, особенно когда пользователи находятся в территориально удаленных офисах. В APEX встроен инструмент конвертации из таблиц Excel в таблицы APEX. После конвертации эти таблицы становятся доступны на корпоративном Интранет- или Интернет-сайте. Пользователь получает доступ к такой таблице после того, как вводит имя и пароль на сайте. Таким образом, можно организовать совместную работу над документом без пересылки его по электронной почте и т. п.

Разработка в APEX может вестись на нескольких языках: PHP, Java, PL/SQL. При разработке на PL/SQL, внутреннем языке базы данных Oracle Database, можно обойтись без промежуточного звена в виде web-сервера Apache (Oracle HTTP Server), HTML-код будет выдавать непосредственно СУБД.

***15.4. Построение приложения с использованием Oracle Application Express: часть I***

В главе использованы материалы [[apex\_lab3]](http://enisey.name/umk/upr21/ch15s07.html#apex_lab3).

Требования:

Иметь доступ к Oracle Database 11g или 10g

Иметь установленный Application Express Release 4.1.1 в Oracle Database 11g.

Создать рабочее пространство HR основанное на HR схеме

Загрузить файл files.zip

***15.4.1. Использование Data Workshop для загрузки данных в таблицу***

В этом разделе загрузим данные в таблицу Projects.

1. Запустите APEX и войдите в рабочее пространство hr: см. 3.6.

2. Выберите SQL Workshop > Data Workshop.

4. В области Data Load выберите Text Data.

5. Загрузите данные в таблицу PROJECTS: выберите Existing table для Load To и нажмите Next.

6. Выберите PROJECTS  в Table Name и нажмите Next.

7. Выберите Browse.

8. Выберите файл projects.txt и нажмите Open.

9. Файл разделен табуляцией, поэтому измените Separator на \t нажмите Next.

10. Информация о сопоставлении столбцов отображена. Нажмите  Load Data.

11. Данные загружены удачно.

**Рисунок 15.18. Данные загружены в APEX**

|  |
| --- |
| Данные загружены в APEX |

***15.5. Построение приложения с использованием Oracle Application Express: часть II***

В главе использованы материалы [[apex\_lab4]](http://enisey.name/umk/upr21/ch15s07.html#apex_lab4).

Требования:

Иметь доступ к Oracle Database 11g или 10g

Иметь установленный Application Express 4.1.1.

Создать рабочее пространство HR основанное на HR схеме

Загрузить файл files.zip

***15.5.1. Создание и переключение между разными темами***

В этом разделе создадим другую тему и переключимся на нее сделав ее текущей темой.

1. Запустите приложение созданное в предыдущей работе 4. Если она пропущена или вы запутались при ее выполнении, то импортируйте приложение из файла f100.sql.

2. Просмотрите стиль ваших вкладок. Изменим тему. Выберите  кнопку Application в developer toolbar.

3. Выберите Shared Components.

4. В области User Interface выберите Themes.

5. Выберите Create.

6. Примите значения по умолчанию и нажмите Next.

7. Выберите Theme 22 и нажмите Next.

8. Выберите Create.

9. Тема создана  в приложении. Переключитесь в новую тему, выберите Switch Theme.

10. Примите значения по умолчанию 22. Bluejay и нажмите Next.

11. Примите значения по умолчанию и нажмите Next.

12. Выберите Switch Theme.

13. Запускаем приложение с новой темой. Выберите Run Page 1 в правом верхнем углу окна.

14. Заметьте что стиль новый. Выберите кнопку Application в  Developer Toolbar.

**Рисунок 15.25. Темы в APEX**

|  |
| --- |
| Темы в APEX |

***15.6. Контрольная работа оРЕX***

Создайте web-приложение для просмотра, обновления, поиска информации таблиц БД  в соответствии с вариантом задания.

***15.6.1. Настройка среды***

Работа выполняется в среде Oracle 10g XE и APEX 4.  В качестве среды можно использовать:

1. локальную СУБД Oracle установленную на рабочем компьютере или
2. сервер СУБД Oracle установленную на сервере 192.168.100.252 или
3. облачный сервис apex.oracle.com

**Рисунок 15.32. Вход в рабочее пространство в APEX**

