**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет прикладной математики и информатики**

**Кафедра МСС**

**Понтелей Виталий Витальевич**

Отчет по лабораторной работе №2-3

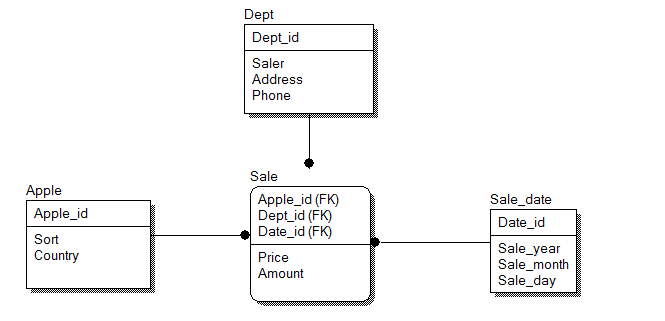
Студента 3 курса 12 группы

**Преподаватель**

*Довнар С.Е.*

Минск 2013

1. Строение базы данных



Сущность Apple содержит поля: Apple\_id (PK), обозначающее идентификационный номер яблока, Sort -- cорт яблока, Country -- страна экспортер.

Сущность Dept содержит поля: Dept\_id (PK) -- идентификационный номер места продажи, Saler -- ФИО продавца, Address -- адрес точки продажи, Phone -- телефон продавца.

Сущность Sale\_date содержит поля: Date\_id (PK) -- идентификационный номер даты, Sale\_year -- год продажи, Sale\_month -- месяц продажи, Sale\_day -- день продажи.

Сущность Sale содержит поля: Apple\_id (FK) -- идентификационный номер яблока, Dept\_id (FK) -- идентификационный номер места продажи, Date\_id (FK) -- идентификационный номер даты, Price -- цена проданного яблока, Amount -- количество проданных яблок.

1. Описание XML отчета

<?xml version="1.0" encoding="utf-8"?>

<Applebd database="database/Apple.db3">

<DimensionByColumn>Dept</DimensionByColumn>

<DimensionByRow>Sale\_date</DimensionByRow>

<FixedDimension>

<Apple>

<Apple\_id>1</Apple\_id>

</Apple>

</FixedDimension>

<Selection>

<Dept>

<Dept\_id>1 2</Dept\_id>

</Dept>

<Sale\_date>

<Date\_id>3 2</Date\_id>

</Sale\_date>

</Selection>

</Applebd>

Первый тег Applebd содержит атрибут database, который указывает путь к базе данных. Тег DimensionByColumn содержит имя таблицы, которая будет отображаться по горизонтали. Тег DimensionByRow содержит имя таблицы, которая будет отображаться по вертикали. Тег FixedDimension содержит зафиксированную таблицу, и идентификатор, по которому происходит фиксация. Тег Selection содержит таблицы, по которым происходит выборка, по указанным идентификаторам.

1. Описание XML-файла метаданных

Тег Dimensions содержит измерения кубика, приведем пример одного из таких измерений:

<Apple>

<DimensionInfo>

<Id>1</Id>

<Name>Яблоки</Name>

</DimensionInfo>

<Fields>

<Apple\_id>

<Attributes>

<Id>1</Id>

<PK>true</PK>

<Pname>Идентификатор яблока</Pname>

<Type>int</Type>

<Visible>false</Visible>

</Attributes>

<DBFields>

<Name>Apple\_id</Name>

<Type>int</Type>

<Attribute>false</Attribute>

</DBFields>

</Apple\_id>

<Sort>

<Attributes>

<Id>2</Id>

<PK>false</PK>

<Pname>Сорт яблока</Pname>

<Type>text</Type>

<Visible>true</Visible>

</Attributes>

<DBFields>

<Name>Sort</Name>

<Type>text</Type>

<Attribute>true</Attribute>

</DBFields>

</Sort>

<Country>

<Attributes>

<Id>3</Id>

<PK>false</PK>

<Pname>Страна экспортер</Pname>

<Type>text</Type>

<Visible>false</Visible>

</Attributes>

<DBFields>

<Name>Country</Name>

<Type>text</Type>

<Attribute>true</Attribute>

</DBFields>

</Country>

<Parent\_key>

<Attributes>

<Id>4</Id>

<PK>false</PK>

<Pname>Ключ родителя</Pname>

<Type>int</Type>

<Visible>false</Visible>

</Attributes>

<DBFields>

<Name>Parent\_key</Name>

<Type>int</Type>

<Attribute>false</Attribute>

</DBFields>

</Parent\_key>

</Fields>

<TableInfo>

<NameDB>Apple</NameDB>

<PK>Apple\_id</PK>

<Parent>Parent\_key</Parent>

</TableInfo>

</Apple>

В данном примере представлена таблица Apple. В DimensionInfo, представлены, уникальный идентификатор для данной таблицы и выводимое имя таблицы. В теге Fields приведены поля данной таблицы. В теге Attributes приведены по порядку, уникальный идентификатор поля, является ли поле первичным ключом, имя отображаемое для пользователя, тип, и является ли оно отображаемым. В теге DBFields включены по порядку: имя поля в реальной базе данных, тип в реальной базе данных, и является ли оно атрибутом. После тега Fields идет последний тег TableInfo, который включает: имя таблицы в реальной базе данных, первичный ключ и родительский ключ.

После тега Dimensions, следует тег DataTable, который содержит в себе всю информацию о таблице фактов, приведем пример:

<DataTable name="Sale">

<Fields>

<Apple\_id>

<Name>Apple\_id</Name>

<Type>int</Type>

<Link>Apple</Link>

</Apple\_id>

<Dept\_id>

<Name>Dept\_id</Name>

<Type>int</Type>

<Link>Dept</Link>

</Dept\_id>

<Date\_id>

<Name>Date\_id</Name>

<Type>int</Type>

<Link>Sale\_date</Link>

</Date\_id>

<Price>

<Name>Price</Name>

<Type>double</Type>

<Link></Link>

</Price>

<Amount>

<Name>Amount</Name>

<Type>int</Type>

<Link></Link>

</Amount>

</Fields>

</DataTable>

В самом теге присутствует атрибут, показывающий имя таблицы фактов. В теге Fields, содержатся поля таблицы фактов. В свою очередь они содержат теги с информацией о: названии поля в реальной базе данных, типе в реальной базе данных, и связи с таблицей.

1. Описание программы утилиты создающей XML-отчет

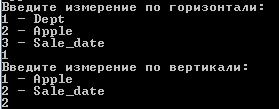
При запуске программы утилиты (CreateXML), первым делом она запрашивает имя выходного xml-файла:



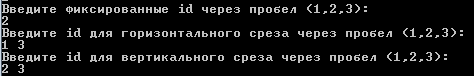
После этого вам придется ввести путь к базе данных:



Затем вам предлагается выбрать измерения по горизонтали и по вертикали:



Далее требуется ввести фиксированный идентификатор, идентификатор для горизонтального и вертикального среза:



После этого в каталоге, который вы указали, создастся xml-отчет.

1. Описание программы делающей срез по xml-отчету:

Программа содержит следующие классы:

1. Класс Field, который описывает какое-либо поле таблицы, содержит:

- nameOfTable – имя таблицы

- id – уникальный идентификатор поля

- PK – показывает, является ли поле первичным ключом

- name – имя поля

- pname – имя поля для вывода

- type – тип поля

- visible – показывает, является ли поле отображаемым

- nameinDB – имя поля в реальной базе данных

- attribute – показывает, является ли поле аттрибутом

2) Класс Fact, описывает поле таблицы фактов:

- name – имя поля

- nameinDB – имя поля в реальной базе данных

- type – тип в реальной базе данных

- link – связь с таблицей

3) Класс Table, описывает таблицу в базе данных. Содержит поля:

- table – имя таблицы

- tableBD – имя таблицы в базе данных

- pk – первичный ключ

- parK – родительский ключ

- fields – список полей, содержащихся в таблице (список Field)

Конструктор данного класса принимает ref XmlTextReader reader, который должен указывать на начало описания таблицы в метаданных.

4) Класс Datatable, описывает таблицу фактов, с двумя полями:

- nameDT – имя таблицы в базе данных

- facts – список полей в данной таблице (список Fact)

Конструктор этого класса также принимает ref XmlTextReader reader, однако в отличии от предыдущего необязательно, чтобы он указывал на DataTable, достаточно, чтобы данный тег можно было достичь с этой точки.

5) Класс Section, описывает xml-файл отчет и содержит:

- path – путь к базе данных

- dimByColumn – срез по горизонтали

- dimByRow – срез по вертикали

- fixedDim – фиксированная таблица

- fixedField – фиксированное поле

- fixedId – список фиксированных идентификаторов

- selectField – список выборочных полей

- selectId – списки выборочных идентификаторов

- selectDim – список выборочных измерений

Конструктор принимает путь к xml-отчету и парсит его с помощью следующего кода:

XmlTextReader reader = new XmlTextReader(xmlFile);

while (reader.Read())

{

if (reader.NodeType == XmlNodeType.Element)

{

if (reader.Name == "Applebd")

{

reader.MoveToNextAttribute();

path = reader.Value;

}

else if (reader.Name == "DimensionByColumn")

{

reader.Read();

dimByColumn = reader.Value;

}

else if (reader.Name == "DimensionByRow")

{

reader.Read();

dimByRow = reader.Value;

}

else if (reader.Name == "FixedDimension")

{

skip(ref reader);

fixedDim = reader.Name;

skip(ref reader);

fixedField = reader.Name;

reader.Read();

string [] split = reader.Value.Split(new Char[] { ' ' });

foreach (string s in split)

fixedId.Add(Convert.ToInt32(s));

}

else if (reader.Name == "Selection")

{

reader.Read();

int i = 0;

for (; i<2; i++)

{

skip(ref reader);

selectDim.Add(reader.Name);

skip(ref reader);

selectField.Add(reader.Name);

reader.Read();

string[] split = reader.Value.Split(new Char[] { ' ' });

selectId.Add(new List<int>());

foreach (string s in split)

selectId[i].Add(Convert.ToInt32(s));

}

}

}

}

Во всех предыдущих классах принцип чтения xml-файла такой же.

6) Класс SelectQueries, класс содержащий запросы для вывода информации:

- queries – запросы для вывода

- idCol – именование столбцов

- idRow – именование рядов

Конструктор принимает класс Datatable, Table[] и Section.

7) Класс Programm создает все нужные классы для вывода результата в консоль, при том если нужные данные не найдены, в таблицу выводится 0. Пример вывода:

