

PF_XML6 Пакет анализа/генерации обменных XML-файлов индивидуальных сведений Пенсионного фонда РФ 2011-2016гг (Версия 70200)

DocRev 71002

0. Замечания

- в связи с радикальным изменением отчетности ПФР с 2017г пакет более не поддерживается;
- содержимое документа может незначительно отличаться от последней версии пакета.

1. Назначение.

Пакет предназначен для:

- анализа (парсинга) XML-файлов индивидуальных сведений ПФР 2011-2016гг, содержащих исходные, корректирующие и отменяющие формы СЗВ-6-1(2), СЗВ-6-3, СЗВ-6-4, СЗВ-РСВ (РСВ-1 раздел 6) и исходных форм РСВ-1 (разделы 1, 2.1, 2.5, 4);
- генерации XML-файлов пакетов исходных, корректирующих, отменяющих форм СЗВ-6-1(2), СЗВ-6-4, СЗВ-РСВ и исходной расчетной формы РСВ-1 2014 года (разделы 1, 2.1, 2.5, 4).

Парсинг и генерация производятся в/из коллекции PL/SQL, что обеспечивает независимость от структур таблиц Oracle DB. Пакет совместим с Oracle-XE 11g.

2. Ограничения реализации.

- при анализе и генерации проверяются только наличие и элементарный тип данных, форматно-логический контроль не производится;
- дополнительные записи о стаже (период работы) не обрабатываются;
- форма РСВ-1 анализируется и генерируется в объеме разделов 1, 2.1, 2.5.1, 2.5.2, 4;
- формы СЗВ-6-1(2) 2010г с периодом отчетности полугодие не поддерживаются.

Расчет данных формы РСВ-1 производится с помощью процедур пакета PF_CALC.

3. Структуры данных.

Структуры данных определяют сведения об:

- организации-плательщике и отчетном периоде;
- XML-файлах индивидуальных сведений;
- формах индивидуального учета ПФР;
- именах разделов и полей таблицы данных пакета для расчетной формы РСВ-1.

В пакете используется унифицированная структура формы индивидуальных сведений, состоящая из:

- заголовка, содержащего вид формы и тип сведений, данные о корректируемом периоде, идентификацию и характеристики застрахованного лица, суммы страховых взносов;
- набора записей о периодах работы;
- набора записей о доходах физического лица и базах для начисления страховых взносов;
- набора записей о суммах выплат для начисления взносов по дополнительному тарифу;
- набора записей о корректирующих сведениях (раздел 6.6 РСВ-1).

Данные расчетной формы РСВ-1 хранятся во внутренней таблице пакета, адресуемой именем раздела/поля, номером строки, номером колонки:

- имена разделов и полей таблицы определяются константами пакета;

- количество разделов 2.1 соответствует количеству применяемых тарифов, при этом имя раздела дополняется кодом тарифа;
- номера строк разделов 1 и 2.1 соответствуют коду строки;
- номера строк разделов 2.5.1, 2.5.2 и 4 соответствуют №п/п, итоговые строки имеют номер 0;

- при анализе/генерации заполняются/используются следующие колонки:

раздел 1	колонки 3 - 8	
раздел 2.1	колонки 3 - 6	
раздел 2.5.1	колонки 1 - 5	,итог 2 - 4
раздел 2.5.2	колонки 1 - 8	,итог 4 - 7
раздел 4	колонки 1 - 14,итог 6 - 14	

- поля титульного листа адресуются как и разделы - по имени с номерами строки и колонки равными 0.

Обмен данными с таблицей производится посредством интерфейсных функций.

3.1 Сведения об организации-плательщике и отчетном периоде

-- ORG Организация-плательщик и отчетный период

Type org_rc is record

```
( ONAME varchar2(100) -- краткое наименование организации
, OPFN number(12) -- регистрационный номер организации в ПФР
, OINN number(10) -- ИНН организации
, OKPP number(9) -- КПП организации
, OKVED varchar2(16)-- ОКВЭД (только для РСБ-1)
, PYEAR number(4) -- отчетный год
, PINDEX number(1) -- индекс отчетного квартала (1-4)
, SDECK number(5):=0 -- стартовый номер пачек
);
```

3.2 Унифицированная форма ПФР

-- Вид (номер) формы

```
fn_szv1 constant number(1) := 1; -- СЗВ-6-1 (2010-2012гг)
fn_szv2 constant number(1) := 2; -- СЗВ-6-2 (СЗВ-6-1 без льготных периодов)
fn_szv3 constant number(1) := 3; -- СЗВ-6-3 (2011-2012гг не упоминается с 2014г)
fn_szv4 constant number(1) := 4; -- СЗВ-6-4 (2013г)
fn_szvr constant number(1) := 5; -- СЗВ-РСБ (2014г-? РСБ-1 6-й раздел)
--
fn_adv constant number(1) := 8; -- опись пачек форма АДВ
fn_rsv constant number(1) := 9; -- расчетная форма РСБ-1
```

subType money_tp is number(15,2);

subType empty_tp is varchar2(8); -- код категории застрахованного лица

-- Заголовок формы (СЗВ-6-1,2,3,4, СЗВ-РСБ)

Type hdr_rc is record

```
( FID number -- индекс формы
--, EXTFID varchar2(128) -- внешний идентификатор формы
, FN number(1) -- вид (номер) формы
, LNAME varchar2(80) -- фамилия
, FNAME varchar2(80) -- имя
, SNAME varchar2(80) -- отчество
, SNILS number(11) -- страховой номер индивидуального лицевого счета (СНИЛС)
, FTYPE varchar2(6) -- тип сведений: ИСХ, КОР, ОТМ
, CPINDEX number(1) -- индекс (квартал) корректируемого периода (1-4)
, CPYEAR number(4) -- корректируемый год (2011-2014)
, COPFN number(12) -- регистрационный номер организации в ПФР
--, ADDR1 varchar2(256) -- адрес для информирования(отменен с 2013г)
, EMPTY empty_tp -- код категории плательщика: НР, ВЖНР, ООИ, ...
```

```
, CONTYPE varchar2(6) -- тип договора: 'ТРУ'-ТРУДОВОЙ,'ГРА'-ГРАЖДАНСКО-ПРАВОВОЙ
, ISUM money_tp      -- страховая часть взносов или ОПС 2014г начислено
, FSUM money_tp      -- накопительная часть взносов начислено
, CISUM money_tp      -- страховая часть взносов или ОПС 2014г
--                из корректируемой формы (уплачено)
, CFSUM money_tp      -- накопительная часть взносов
--                из корректируемой формы (уплачено)
, XID number(2)      -- индекс xml-файла
);
```

-- Таблица форм

Type hdr_tbl is table of hdr_rc;

-- Запись о периоде работы

Type pd_rc is record

```
( FID number      -- индекс формы
, NN number(2)    -- порядковый номер записи 1-99
, SDATE date      -- дата начала периода
, EDATE date      -- дата окончания
, TERR varchar2(8) -- территориальные условия: МКС, РКС, ...
, TERRC number(4,2) -- доля ставки ТУ
, DD varchar2(20) -- декрет/дети
, SPECS varchar2(128)-- строка прочих особенностей учета,
--                разделенных ':' или null
);
```

SPECS: позиции особенностей учета в строке

1	ОсобыеУсловияТруда/ОснованиеОУТ
2	ОсобыеУсловияТруда/ПозицияСписка
3	ИсчисляемыйСтаж/ОснованиеИС
4	ИсчисляемыйСтаж/ВыработкаВчасах/Часы
5	ИсчисляемыйСтаж/ВыработкаВчасах/Минуты
6	ИсчисляемыйСтаж/ВыработкаКалендарная/ВсеМесяцы
7	ИсчисляемыйСтаж/ВыработкаКалендарная/ВсеДни
8	ВыслугаЛет/ОснованиеВЛ
9	ВыслугаЛет/ВыработкаВчасах/Часы
10	ВыслугаЛет/ВыработкаВчасах/Минуты
11	ВыслугаЛет/ВыработкаКалендарная/ВсеМесяцы
12	ВыслугаЛет/ВыработкаКалендарная/ВсеДни
13	ВыслугаЛет/ДоляСтавки

-- Список и таблица периодов

Type pd_tb is table of pd_rc index by pls_integer;

Type pd_tbl is table of pd_rc;

-- Запись о доходах, облагаемых базах

Type inc_rc is record

```
( FID number      -- индекс формы
, PMON number(2)  -- месяц 1-12 (0 - сумма нарастающим итогом с начала года)
, EMPTYTYPE emptytype_tp -- категория застрахованного лица: НР, ВЖНР, ООИ, ...
, SUMT money_tp   -- сумма выплат
, SUMB money_tp   -- облагаемая база
, SUMBG money_tp  -- ... в т.ч. по ГПД
, SUMO money_tp   -- сумма превышения
);
```

-- Список и таблица доходов

Type inc_tb is table of inc_rc index by pls_integer;

Type inc_tbl is table of inc_rc;

-- Запись о дополнительных выплатах

Type pmt_rc is record

```
( FID number      -- индекс формы
, PMON number(2)  -- месяц 1-12 (0 - сумма нарастающим итогом с начала года)
, ACODE varchar2(10) -- код специальной оценки труда
, SUMA1 money_tp  -- сумма выплат пп1...
, SUMA2 money_tp  -- ... пп2...
);
```

```
-- Список и таблица выплат
Type pmt_tb is table of pmt_rc index by pls_integer;
Type pmt_tbl is table of pmt_rc;
-- Сведения о корректировках [5]
Type cor_rc is record
( FID number      -- индекс формы
, CPINDEX number(1) -- индекс корректируемого квартала
, CPYEAR number(4) -- корректируемый год (null или 0 = итоговая строка)
, DPSUM money_tp   -- сумма доначисленных взносов ОПС
, DISUM money_tp   -- ... страховая часть
, DFSUM money_tp   -- ... накопительная часть
);
Type cor_tb is table of cor_rc index by pls_integer;
Type cor_tbl is table of cor_rc;
-- Унифицированная форма индивидуального учета
Type form_tp is record
( HDR hdr_rc      -- заголовок формы
, PERIODS pd_tb   -- список записей о периодах работы
, INCOMES inc_tb   -- список записей о доходах/базах
, PAYMENTS pmt_tb -- список записей о дополнительных выплатах
, CORRS cor_tb    -- сведения о корректировках
);
```

3.3 XML-файлы

```
-- XML Описание файла
Type xml_rc is record
( XID number(2)      -- индекс файла (последовательный номер: 1-99)
, XFCOUNT number(3):=0 -- количество форм в файле 1-200
-- ключ группировки форм
, FN number(1)       -- вид (номер) формы
, FTYPE varchar2(6)  -- тип сведений: ИСХ, КОР, ОТМ
, CPINDEX number(1)  -- индекс (квартал) корректируемого периода (1-4)
, CPYEAR number(4)   -- корректируемый год (2010-2014)
, EMPTYPE emptytype_tp -- код категории ЗЛ: НР, ВЖНР, ООИ, ...
, CONTYPE varchar2(6) -- тип договора:
--                      'ТРУ'-ТРУДОВОЙ, 'ГРА'-ГРАЖДАНСКО-ПРАВОВОЙ
);
-- XML Таблица файлов
Type xml_tbl is table of xml_rc;
-- XML Информация о файле
Type xml_info_tp is record
( XNAME varchar2(128) -- имя файла
, XPROGRAM varchar2(80) -- программа подготовки данных
, XVERSION varchar2(20) -- версия программы
, ORG org_rc          -- сведения об организации и отчетном периоде
, XFILE xml_rc         -- описание файла
);
```

3.4 Данные формы РСБ-1

```
subType partname_tp is varchar2(32);
-- Имена разделов данных формы РСБ-1
part1_name constant partname_tp := 'PART_1';
-- PART_2.1-3 , 3 - код тарифа
part21_name constant partname_tp:='PART_2.1-';
part251_name constant partname_tp:='PART_2.5.1';
part252_name constant partname_tp:='PART_2.5.2';
part4_name constant partname_tp:='PART_4';
-- Имена полей титульной страницы
```

fld_empcount constant partname_tp:='FLD_EMPCOUNT'; -- количество ЗЛ
fld_empavg constant partname_tp:='FLD_EMPAVG'; -- среднесписочная численность
-- Таблица имен разделов или номеров строк или номеров колонок
-- имя поля таблицы: column_value
Type part_names_tbl is table of partname_tp;

4. Процедуры и функции.

Общие сведения:

- функции пакета могут генерировать перехватываемые исключения:

pf_xml6.period_error	ошибка периода
pf_xml6.org_error	ошибка в данных организации-плательщика
pf_xml6.form_error	прочие ошибки данных формы
pf_xml6.unsupported	файл/данные не поддерживаются

- анализатор XML-файлов индифферентен к позиции вложенных элементов XML;
- генерация файлов производится в соответствии с требованиями ПФР (checkPFR);
- нормальной практикой является освобождение LOB после использования с помощью функции `dbms_lob.free_temporary` и очистка внутренних данных пакета беспараметровой процедурой `init`.

4.1 Инициализация, сведения об организации-плательщике и отчетном периоде

Procedure init;

Procedure init(p_org org_rc);

Очистить и инициализировать структуры данных.

Procedure org_set(p_org org_rc);

Установить сведения об организации-плательщике и отчетном периоде. Обязательно для генерации. Не обязательно для анализа XML-файлов.

Function org_get return org_rc;

Получить сведения о плательщике и отчетном периоде.

4.2 Формы индивидуального учета ПФР

Procedure form_clear(p_form in out nocopy form_tp);

Очистить структуру пользовательской формы.

Procedure form_add(p_form in out nocopy form_tp);

Добавить (заменить, при наличии FID) форму.

Проверяется наличие сведений об организации и отчетном периоде. В случае отсутствия, генерируется перехватываемое исключение `pf_xml6.org_error` или `pf_xml6.period_error`. По году корректируемого периода и наличию льготной составляющей в периодах работы (для СЗВ-6-1,2) определяется вид (номер) формы. Добавляемая форма ассоциируется с файлом. Возвращает FID формы и XID файла в заголовке формы.

Function form_get(p_fid number) return form_tp;

Возвращает форму по ее FID.

Procedure form_delete(p_fid number);

Удалить форму.

Function forms_tbl return hdr_tbl pipelined;

Возвращает таблицу заголовков форм.

Function periods_tbl(p_fid number:=null) return pd_tbl pipelined;

Возвращает таблицу периодов форм(ы).

Function incomes_tbl(p_fid number:=null) return inc_tbl pipelined;

Возвращает таблицу доходов/баз форм(ы).

Function payments_tbl(p_fid number:=null) return pmt_tbl pipelined;

Возвращает таблицу выплат по дополнительному тарифу форм(ы).

Function corrs_tbl(p_fid number:=null) return cor_tbl pipelined;

Возвращает таблицу корректирующих сведений форм(ы).

4.3 XML-файлы

Function files_tbl return xml_tbl pipelined;

Возвращает таблицу файлов.

Function file_get(p_xid number) return xml_rc;

Получить сведения о содержимом файла.

Procedure file_delete(p_xid number);

Удаляет файл и связанные с ним формы из списка.

Function file_info_get(p_xml CLOB) return xml_inf_tp;

Получить сведения о файле.

Function parse(p_xml CLOB) return number;

Анализирует XML-файл ПФР. Возвращает XID файла (1-99).

Добавляет запись сведений о файле. Добавляет формы сведений индивидуального учета в список форм. Если установлены сведения об организации и отчетном периоде - проверяется соответствие регистрационного номера ПФ (ИНН, КПП) организации и отчетного периода. При несовпадении генерируется перехватываемое исключение: pf_xml6.org_error или pf_xml6.period_error.

При анализе РСВ-1 функция возвращает 0. Формируется таблица разделов РСВ. Запись о файле не создается.

Примечания:

- индекс отчетного квартала форм СЗВ-6-3 при анализе устанавливается = 4;

Function generate(p_xid number:=0) return CLOB;

Генерирует XML-файл форм индивидуального учета по его XID (1-99). При опущенном или равном 0 p_xid генерируется расчетная форма РСВ-1 2014г.

Особенности генерации файлов индивидуального учета:

- генерируются файлы форм СЗВ-6-1(2), СЗВ-6-4, СЗВ-РСВ;
- если не указаны (null) суммы уплаченных страховых взносов ПФР (CISUM, CFSUM), подставляется сумма начисленных (ISUM, FSUM) страховых взносов;
- сумма доначисленных взносов определяется разностью начисленных и уплаченных;
- с целью получения диагностики checkPFR по XML-файлу, в случае отсутствия данных о доходах/базах формы СЗВ-РСВ (раздел 6.4) формируется 'пустой' итоговый блок с категорией ЗЛ = 'НР';
- номер файла форм в пакете (DCK) = начальный номер пачки (org.sdeck) + индекс файла (XID).

Function generate return CLOB;

Генерирует XML-файл расчетной формы РСВ-1 2014г по данным таблицы разделов и сведениям об организации и отчетном периоде.

Особенности генерации РСВ-1:

- генерируются титул и разделы 1, 2.1, 2.5.1, 2.5.2, 4;
- при генерации номер файла РСВ-1 в пакете (DCK) = начальный номер пачки (org.sdeck) + максимальный индекс файла форм индивидуального учета (xid) + 1

4.4 Работа с данными формы РСВ-1

Procedure parts_erase;

Очистить структуру данных формы РСБ-1

Function parts_tbl

```
( p_name varchar2:=null          -- имя раздела  
 , p_row number:=null          -- номер строки  
) return part_names_tbl pipelined;
```

Возвращает таблицу (имя поля таблицы: column_value) имен разделов и полей титульного листа данных формы РСБ-1 или номеров строк/колонок при значениях p_name/p_row отличных от null.

Procedure part_value_set

```
(p_value varchar2, p_part varchar2, p_row pls_integer:=0, p_col pls_integer:=0);
```

Установить значение ячейки таблицы данных РСБ-1.

Procedure part_value_add

```
(p_value number, p_part varchar2, p_row pls_integer:=0, p_col pls_integer:=0);
```

Сложить со значением ячейки таблицы данных РСБ-1 и сохранить.

Function part_value_get

```
(p_part varchar2, p_row pls_integer:=0, p_col pls_integer:=0) return varchar2;
```

Получить значение ячейки таблицы данных РСБ-1.

4.5. Вспомогательные функции.

```
Function pfn2num(p_fn varchar2) return number;
```

```
Function num2pfn(p_fn number) return varchar2;
```

Преобразовать символьное представление регистрационного номера плательщика в/из внутреннего представления (000-000-000000).

```
Function snils2num(p_snils varchar2) return number;
```

```
Function num2snils(p_snils number) return varchar2;
```

Преобразовать символьное представление СНИЛС (000-000-000 00 или 000-000-000-00) в/из числа (000-000-000 00).

```
Function fn2name(p_fn number) return varchar2;
```

Преобразовать номер формы в символьный вид: 1 в 'СЗВ-6-1', ..., 9 в 'РСБ-1'

```
Function money2x(p_num number) return varchar2;
```

```
Function x2money(p_num varchar2) return number;
```

Преобразование сумм в/из символьного формата [-]12345678901.99

```
Function date2x(p_date date) return varchar2;
```

```
Function x2date(p_date varchar2) return date;
```

Преобразование дат в/из символьного формата DD.MM.YYYY

```
Function pindex2x(p_index number, p_year number:=null) return number;
```

```
Function x2pindex(p_code number, p_year number:=null) return number;
```

Преобразовать индекс отчетного (корректируемого) квартала (1-4) в/из код периода в зависимости от отчетного года:

до 2014 1,2,3,4

с 2014 3,6,9,0

Если параметр p_year опущен, производится безусловное преобразование 1-4 в/из 3,6,9,0.

```
Function period_specn_get(p_specs varchar2, p_n number) return varchar2;
```

Вернуть n-ю особенность учета из поля SPECS периода деятельности.

```
Function file_name_make(p_xid number) return varchar2;
```

По установленным данным организации (org.pyear, org.opfn, org.sdeck) и p_xid формируется имя XML-файла вида:

PFR-700-Y-2014-ORG-123-456-789012-DCK-34567-DPT-000000-DCK-00000.XML

```
Function blob2clob(p_xmlb BLOB) return CLOB;
```

```
Function clob2blob(p_xmlc CLOB) return BLOB;
```

Преобразовать бинарное (windows-1251) представление XML-файла ПФР к внутреннему символному Oracle и наоборот.

5. Примеры использования

5.1 Пример анализа XML-файла ПФР.

В примере бинарное содержимое файла передается в анонимный блок параметром XMLBLOB.

```
declare
  l_org pf_xml6.org_rc;           --организация и отчетный период
  l_incs pf_xml6.inc_tb; --список доходов и баз для начисления взносов ПФР
  l_xclob CLOB;
  l_xid number;
begin
  pf_xml6.init;                  --инициализация
  l_xclob:=pf_xml6.blob2clob(:XMLBLOB);
  --получить имя файла и распечатать
  dbms_output.put_line('Файл: '||pf_xml6.file_info_get(l_cxml).xname);
  --получить сведения об организации и отчетном периоде
  l_org:=pf_xml6.file_info_get(l_clob).org;
  dbms_output.put_line('Организация: '||l_org.oname
    ||' PerN ПФР: '||pf_xml6.num2pfn(l_org.pfn));
  dbms_output.put_line('Отчетный год/квартал: '||l_org.pyear
    ||'/'||l_org.pindex);
  l_xid:=pf_xml6.parse(l_clob);--анализ файла
  --поиск формы по фамилии застрахованного лица
  for ft in (select * from table(pf_xml6.forms_tbl) where lname='ИВАНОВ' )
  loop
    dbms_output('ФИО: '||ft.lname||' '||ft.fname||' '||ft.sname
      ||' СНИЛС: '||pf_xml6.num2snils(ft.snils));
  --получить сведения о доходах/базах по id формы
  l_incs:=pf_xml6.form_get(ft.fid).incomes;
  if l_incs.count>0 then
    for i in l_incs.first..l_incs.last
    loop
      continue when not l_incs.exists(i);
  --вывести базу для начисления взносов по месяцам периода
      dbms_output.put_line(case when l_incs(i).pmon is null
        then 'Итого с начала года:'
        else 'Месяц:'||l_incs(i).pmon end
        ||' База:'||l_incs.sumb);
    end loop;
  end if;
end loop;
--освободить память
dbms_lob.freetemporary(l_clob);
init;
end;
```

5.2 Пример генерации XML-файла ПФР.

В примере бинарное содержимое сгенерированного файла индивидуальных сведений и имя файла возвращаются в параметрах XMLBLOB и XMLNAME анонимного блока.

```
declare
```



```

l_org pf_xml6.org_rc
l_form pf_xml6.form_rc;
l_cxml CLOB;
begin
--инициализация сведениями об организации и отчетном периоде (IV кв 2014г)
l_org.otype:='Организация';
l_org.opfn:=pf_xml6.pfn2num('123-456-789012');
l_org.oinn:=1234567890;
l_org.okpp:=123456789;
l_org.pyear:=2014;
l_org.pindex:=4;
pf_xml6.init(l_org);
--сведения о застрахованном лице и форме
l_form.hdr.lname:='Иванов';
l_form.hdr.fname:='Иван';
l_form.hdr.sname:='Иванович';
l_form.hdr.snils:=snils2num('123-456-789 00');
l_form.hdr.emptytype:='HP';      --код категории застрахованного лица
l_form.hdr.ftype:='КОР';        --корректирующая форма за 2 квартал 2011г
l_form.hdr.cpyear:=2011;
l_form.hdr.cindex:=2;
l_form.hdr.isum:=12345;         --начислено на страховую часть
l_form.hdr.cisum:=9876;        --уплачено в корректируемом периоде
--сведения о периодах работы
l_form.periods(1).nn:=1;
l_form.periods(1).sdate:=to_date('1.4.2011','DD.MM.YYYY');
l_form.periods(1).edate:=to_date('30.6.2011','DD.MM.YYYY');
l_form.periods(1).terr:='ПКС';
--добавить форму (создается 1 файл индивидуальных сведений с xid=1)
l_form.hdr.fid:=pf_xml6.form_add(l_form);
--сгенерировать содержимое xml-файла, содержащего одну форму СЗВ-6-1
l_cxml:=pf_xml6.generate(1);
--получить имя файла
:XMLNAME:=pf_xml6.file_info_get(l_cxml).xname;
:XMLBLOB:=pf_xml6.clob2blob(l_cxml);
-- освободить память
pf_xml6.form_clear(l_form);
dbms_lob.freetemporary(l_cxml);
init;
end;

```

5.3 Пример анализа формы РСВ-1

Параметр процедуры - бинарное содержимое XML-файла. В процедуре использованы функции преобразования записи в строку пакета PF_CALC.

```

create or replace procedure "RSV_TEST"
(p_bxml BLOB)
is
l_clob CLOB;
l_number;
l_info pf_xml6.xml_info_tp;
begin
pf_xml6.init;
l_clob:=pf_xml6.blob2clob(p_bxml);
-- получить и вывести информацию о файле
l_info:=pf_xml6.file_info_get(l_clob);
-- преобразовать записи в строку и вывести
dbms_output.put_line(pf_calc.rc2char(l_info.org));

```

```

dbms_output.put_line(pf_calc.rc2char(l_info.xfile));
if l_info.xfile.fn = pf_xml6.fn_rsv then
-- анализ файла
l:=pf_xml6.parse(l_clob);
-- просмотр по таблицам/строкам/колонкам PCB-1
for pt in (select column_value as name from table(pf_xml6.parts_tbl)) loop
for rt in
(select column_value as prow from table(pf_xml6.parts_tbl(pt.name)))
loop
for ct in
(select column_value as pcol
from table(pf_xml6.parts_tbl(pt.name,rt.prow)))
loop
-- вывод содержимого ячейки таблицы: раздел строка колонка значение
dbms_output.put_line(pt.name||
to_char(rt.prow,'999')||to_char(ct.pcol,'999')||' '||
pf_xml6.part_value_get(pt.name,rt.prow,ct.pcol));
end loop;
end loop;
end loop;
end if;
-- освободить память
pf_xml6.init;
dbms_lob.freetemporary(l_clob);
end;
```

6. Условия использования

Использование пакета регулируется лицензией MIT.

©2014-2017 miktim@mail.ru Петрозаводский государственный университет. РЦНИТ