

# AM\_FILL Пакет использования OOXML .xlsx файлов Microsoft Office в качестве шаблонов отчетных форм (Версия 2.70329)

Rev 70329

## 1. Назначение

Использование .xlsx файлов Microsoft Office (OOXML) в качестве шаблонов отчетных форм.

Пакет реализован на PL/SQL Oracle (Oracle-XE) 11g с использованием и на основе кода пакетов (автор Anton Scheffer):

AS\_ZIP ( [http://technology.amis.nl/wp-content/uploads/2010/06/as\\_zip7.txt](http://technology.amis.nl/wp-content/uploads/2010/06/as_zip7.txt) ),

AS\_XLSX ( [https://technology.amis.nl/wp-content/uploads/2011/02/as\\_xlsx11.txt](https://technology.amis.nl/wp-content/uploads/2011/02/as_xlsx11.txt) ) и

AS\_READ\_XLSX ( [https://technology.amis.nl/wp-content/uploads/2013/01/as\\_read\\_xlsx9.txt](https://technology.amis.nl/wp-content/uploads/2013/01/as_read_xlsx9.txt) ).

Поддерживаются типы данных Oracle number, date, varchar2 и неявно преобразуемые к перечисленным.

## 2. Интерфейс пакета

**version** constant varchar2(16):='70329';

### Procedure init

( p\_xtemplate BLOB - шаблон .xlsx  
, p\_options varchar2:='' - опции  
);

Процедура инициализирует пакет xlsx-шаблоном.

p\_options: e - разрешить exception в случае ошибки адреса ячейки, по-умолчанию вывод данных игнорируется.

### Procedure init;

Процедура освобождает внутренние структуры данных пакета.

### Procedure in\_field

( p\_value date - значение  
, p\_cell\_addr varchar2 - адрес ячейки в стиле A1 или именованный диапазон  
, p\_options varchar2:='' - опции  
);

### Procedure in\_field

( p\_value number  
, p\_cell\_addr varchar2  
, p\_options varchar2:=''  
);

### Procedure in\_field

( p\_value varchar2  
, p\_cell\_addr varchar2  
, p\_options varchar2:=''  
);

Процедуры заполнения ячеек/именованных диапазонов одиночными значениями соответствующих типов (или неявно к ним преобразуемых). Лист, в который производится печать, становится текущим.

p\_cell\_addr: адрес ячейки ('Лист1!B12') или левая верхняя ячейка именованного диапазона ('диапазон') или именованный диапазон с относительным смещением (при 'диапазон'=Лист1!C4:C4 'диапазон!B12'=Лист1!D15 ).

p\_options: i - режим вставки с последовательным сдвигом строк вниз с сохранением стиля исходной ячейки;

### Procedure in\_table

( p\_query CLOB - текст SQL запроса (без завершающей ; )  
, p\_cell\_addr varchar2 - адрес ячейки в стиле A1 или именованный диапазон  
, p\_options varchar2:='' - опции  
);

Процедура вывода результатов запроса в строки/колонки, начиная от указанной ячейки. Лист, в который производится печать, становится текущим.

p\_options: h - печать заголовков (имен полей);

i - режим вставки записей со сдвигом строк вниз.

В режиме вставки вертикальные объединения ячеек разрываются, т.е. одна запись = одна строка листа.

#### Procedure in\_sheet

```
( p_sheet_name varchar2      - имя исходного листа или диапазон листа
, p_newsheet_name varchar2   - имя нового листа
, p_options varchar2:=''     - опции
)
```

Процедура сохраняет данные указанного листа в лист с новым именем ПОСЛЕ исходного. Данные исходного листа очищаются для повторного использования. Сохраненный лист становится видимым.

p\_options: h – скрыть исходный лист;  
b – вставить ПЕРЕД исходным.

Не контролируется корректность нового имени (разрешенные символы, длина). Попытка сохранить с именем существующего листа игнорируется.

#### Procedure finish

```
( p_xfile in out nocopy BLOB - сформированный .xlsx файл
);
```

Завершает формирование xlsx. При формировании УДАЛЯЕТ ДАННЫЕ ОТЛИЧНЫЕ ОТ СТРОК, ДАТ И ЧИСЕЛ из заполненных листов. Очищает внутренние структуры пакета. Возвращает .xlsx файл.

#### Function address

```
( p_row pls_integer          - номер строки >0
, p_col pls_integer          - номер колонки >0
, p_range_name varchar2:=''  - имя листа или именованного диапазона
) return varchar2;
```

Функция возвращает адрес ячейки в стиле A1.

Диапазоны разыменовываются до ячейки листа, относительный адрес в диапазоне рассчитывается с учетом размеров объединенных ячеек : address(12,4,'диапазон') = 'Лист1!F15'.

p\_range\_name = null возвращает имя текущего листа.

#### Function new\_workbook

```
return BLOB;
```

Возвращает новую книгу MS Excel с одним листом 'Sheet1' в ячейке A1 которого дата в стиле 'YYYY-MM-DD'.

### 3. Примеры использования.

#### 3.1 В качестве шаблона возьмем известный большинству документ:

Расчетный листок НИИЧАВО				
Расчетная дата:				
Должность:				
Подразделение:				
Оклад/Тариф:				
Начисления			Удержания	
Вид начисления	Сумма	Дней / Часов	Вид удержания	Сумма
Начислено			Удержано	
К выплате:		Выплачено через кассу (банк):		
		Долг за предприятием (долг за работником):		

Заполним его, используя базовые процедуры пакета:

```

001 Procedure am_fill_example
002 ( p_xlsx in out nocopy BLOB -- шаблон листка
003 )
004 as
005     type tp_names is table of varchar2(100);
006     type tp_sums is table of number(10,2);
007     -- Для вывода переменной части листка необходима таблица (матрица)
008     -- значений, которая для простоты представлена набором векторов.
009     -- Расчетная часть опущена.
010     l_inames tp_names := tp_names('Оклад','Премия','Отпуск','всего');
011     l_isums tp_sums := tp_sums(60000, 120000, 40000, 220000);
012     l_idays tp_names := tp_names('32 дн','I кв','20 дн');
013     l_rnames tp_names := tp_names('НДФЛ','','','всего удержано');
014     l_rsums tp_sums := tp_sums(28600, null, null, 28600);
015 begin
016     -- инициализация шаблоном с разрешением exception по ошибке имени
017     am_fill.init(p_xlsx,'e');
018     -- заполнение именованных областей титульной части
019     am_fill.in_field(sysdate, 'Расчетная_дата');
020     am_fill.in_field('Привалов Александр Иванович', 'ФИО_сотрудника');
021     am_fill.in_field('заведующий', 'Должность');
022     am_fill.in_field('вычислительный центр', 'Подразделение');
023     am_fill.in_field(l_isums(1), 'Оклад');
024     -- заполнение переменной части в режиме последовательной вставки строк
025     for k in 1..3 loop
026         -- колонки (поля) предпочтительно именовать
027         am_fill.in_field(l_inames(i), 'Нач_Вид', 'i');
028         am_fill.in_field(l_isums(i), 'C8', 'i');
029         am_fill.in_field(l_idays(i), 'D8', 'i');
030         am_fill.in_field(l_rnames(i), 'E8', 'i');
031         am_fill.in_field(l_rsums(i), 'G8', 'i');
032     end loop;
033     -- заполнение итоговой части
034     am_fill.in_field(l_isums(4), 'C9');
035     am_fill.in_field(l_rsums(4), 'G9');
036     am_fill.in_field(l_isums(4)-l_rsums(4), 'К_выплате');
037     -- формирование документа
038     am_fill.finish(p_xlsx);
039 end;
```

Заполненный шаблон выглядит следующим образом:

Нач_Вид		fx		Оклад		
A	B	C	D	E	F	G
1	Расчетный листок					
2	НИИЧАВО					
3	Расчетная дата: Май 2015 г.					
4	Привалов Александр Иванович		Должность: заведующий			
5	Подразделение: вычислительный центр		Оклад/Тариф: 60 000,00			
6	Начисления			Удержания		
7	Вид начисления	Сумма	Дней / Часов	Вид удержания	Сумма	
8	Оклад	60 000,00	32 дн	НДФЛ	28 600,00	
9	Премия	120 000,00	1 кв			
10	Отпуск	40 000,00	20 дн			
11	Начислено	220 000,00		Удержано	28 600,00	
12						
13	К выплате:	191 400,00	Выплачено через кассу (банк):			
14			Долг за предприятием (долг за работником):			
15						
16						

### 3.2 Использование встроенной книги с запросами

```
001 Function am_fill_example() return BLOB
002 as
003     l_query varchar2(2000);
004     l_bxlsx BLOB;
005 begin
006     /* инициализация встроенной книгой */
007     am_fill.init(am_fill.new_workbook(),'e');
008     /* одна таблица на листе, запросы без завершающей ; */
009     l_query := 'SELECT rownum rn, a.* FROM ALL_OBJECTS a
010               WHERE OBJECT_TYPE IN (''FUNCTION'', ''PROCEDURE'', ''PACKAGE'')';
011     am_fill.in_table(l_query, 'A1', 'h');
012     am_fill.in_sheet('', 'Одна таблица на листе');
013     /* две таблицы на листе */
014     am_fill.in_table(l_query, 'A1', 'hi');
015     l_query := 'SELECT OBJECT_NAME, OBJECT_TYPE FROM USER_OBJECTS';
016     am_fill.in_table(l_query, 'A4', 'h');
017     /* сохраним с новым именем и скроем исходный лист */
018     am_fill.in_sheet('', 'Две таблицы на листе', 'h');
019     /* сформируем xlsx */
020     am_fill.finish(l_bxlsx);
021     return l_bxlsx;
022 end;
```