1. Создание таблиц

CREATE TABLE TBL\_SLUISE

(

ID NUMBER(10) NOT NULL

, CODE VARCHAR2(20) NOT NULL

, NAME VARCHAR2(100),

REF\_PAR NUMBER(10) NOT NULL

, CONSTRAINT ID\_PK PRIMARY KEY

(

ID

)

ENABLE

);

ALTER TABLE TBL\_SLUISE

ADD CONSTRAINT REF\_PAR\_UK1 UNIQUE

(

REF\_PAR

)

ENABLE;

CREATE TABLE TBL\_SLUISE\_DOP\_PAR

(

ID NUMBER(10) NOT NULL

, CODE VARCHAR2(20) NOT NULL

, NAME VARCHAR2(100)

, VAL VARCHAR2 (100)

, COLLECTION\_ID NUMBER(10) NOT NULL

, CONSTRAINT COLLECTION\_ID\_FK FOREIGN KEY (COLLECTION\_ID)

REFERENCES TBL\_SLUISE(REF\_PAR)

, CONSTRAINT ID2\_PK PRIMARY KEY

(

ID

)

ENABLE

);

INSERT INTO tbl\_sluise (ID, CODE, NAME, REF\_PAR)

VALUES (123456789, '#CBR', 'Центральный Банк России', 987456321);

INSERT INTO tbl\_sluise\_dop\_par (ID, CODE, NAME, VAL, COLLECTION\_ID)

VALUES (123456789, 'URL', 'Адрес (URL) центрального банка России', 'http://www.cbr.ru/scripts/XML\_daily.asp', 987456321);

INSERT INTO tbl\_sluise\_dop\_par (ID, CODE, NAME, VAL, COLLECTION\_ID)

VALUES (123456788, 'TIMEOUT\_CONNECT', 'Таймаут подключения к сервису сек.', '4', 987456321);

INSERT INTO tbl\_sluise\_dop\_par (ID, CODE, NAME, VAL, COLLECTION\_ID)

VALUES (123456787, 'TIMEOUT\_READ', 'Таймаут вычитывания ответа сервиса сек.', '7', 987456321);

CREATE TABLE VALUTE

(

VALCODE VARCHAR2(20) NOT NULL

, CURMARKET VARCHAR2(20) NOT NULL

);

create or replace java source named sbtm\_encode as

import java.io.\*;

import java.net.MalformedURLException;

import java.net.URL;

import java.net.URLConnection;

public class ConnUrl {

public static String connectingUrl (String url) {

String str = null;

URL u;

try {

u = new URL(url);

URLConnection conn = u.openConnection();

DataInputStream in = new DataInputStream(conn.getInputStream());

BufferedReader d = new BufferedReader(new InputStreamReader(in, "windows-1251"));

StringBuilder sb = new StringBuilder();

while ((str = d.readLine()) != null) {

sb.append(str);

}

str = sb.toString();

d.close();

return str;

} catch (MalformedURLException ex) {

System.out.println(ex.toString());

return null;

} catch (IOException e) {

e.printStackTrace();

}

return str;

}

}

CREATE OR REPLACE PACKAGE XML\_PACK IS

PROCEDURE XML\_PARSE

(url in varchar2, p\_out out varchar2);

FUNCTION fConn

(url in varchar2)

RETURN varchar2;

FUNCTION docommand

(p\_in in varchar2, p\_out out varchar2)

return varchar2;

END XML\_PACK;

CREATE OR REPLACE PACKAGE BODY XML\_PACK IS

PROCEDURE XML\_PARSE

(url in varchar2, p\_out out varchar2)

IS

v\_doc dbms\_xmldom.domdocument;

v\_dom\_element dbms\_xmldom.domelement;

v\_buffer varchar2(10000);

v\_buffer2 varchar2(10000) := null;

v\_dom\_node dbms\_xmldom.domnode;

v\_variable XMLType;

v\_chil\_node dbms\_xmldom.domnode;

v\_node\_list dbms\_xmldom.domnodelist;

loc\_xml varchar2(10000);

i int(3);

v\_url varchar2(100);

v\_date varchar2(100);

v2 varchar2(200);

begin

i:=0;

v\_url := url;

loc\_xml := (to\_char(fConn(v\_url)));

v2:=SUBSTR(loc\_xml,INSTR(loc\_xml, '>')+1,INSTR(loc\_xml, '>'));

p\_out:=SUBSTR(V2,INSTR(V2, '"')+1, INSTR(V2, '"')-5);

v\_variable := xmltype(loc\_xml);

v\_doc := DBMS\_XMLDOM.newdomdocument(v\_variable);

loop

v\_dom\_element := dbms\_xmldom.getdocumentelement(doc => v\_doc);

v\_node\_list := dbms\_xmldom.getelementsbytagname(elem => v\_dom\_element,name => 'CharCode');

v\_dom\_node := dbms\_xmldom.item(nl => v\_node\_list,idx => i);

v\_chil\_node := dbms\_xmldom.getFirstChild(n => v\_dom\_node);

dbms\_xmldom.writetobuffer(n => v\_chil\_node, buffer => v\_buffer);

v\_node\_list := dbms\_xmldom.getelementsbytagname(elem => v\_dom\_element,name => 'Value');

v\_dom\_node := dbms\_xmldom.item(nl => v\_node\_list,idx => i);

v\_chil\_node := dbms\_xmldom.getFirstChild(n => v\_dom\_node);

dbms\_xmldom.writetobuffer(n => v\_chil\_node, buffer => v\_buffer2);

IF V\_BUFFER IS NOT NULL OR V\_BUFFER2 IS NOT NULL THEN

INSERT INTO VALUTE (VALCODE,curmarket) VALUES (V\_BUFFER, V\_BUFFER2);

END IF;

i:=i+1;

exit when v\_buffer is null;

end loop;

end XML\_PARSE;

FUNCTION fConn

(url in varchar2)

RETURN varchar2

AS LANGUAGE JAVA

NAME 'ConnUrl.connectingUrl (java.lang.String)

return java.lang.String';

FUNCTION docommand

(p\_in in varchar2, p\_out out varchar2)

return varchar2

is

URL VARCHAR2(100);

LOC VARCHAR2(150);

V\_VAR1 VARCHAR2(100);

V\_RES1 VARCHAR2(100);

V\_VAR2 VARCHAR2(100);

V\_RES2 VARCHAR2(100);

V\_CODE VARCHAR2(10);

curm VARCHAR2(20);

exc exception;

res\_conn number(2);

v\_loc varchar(100);

v\_date varchar2(50);

BEGIN

LOC := p\_in;

v\_var1:=SUBSTR(LOC,1,INSTR(LOC, '|')-1);

V\_RES1 := SUBSTR(V\_VAR1,INSTR(V\_VAR1, '<')+1, 4); --code

V\_VAR2:=SUBSTR(LOC,INSTR(LOC, '|')+1,(LENGTH(LOC)));

V\_RES2 := SUBSTR(V\_VAR2,INSTR(V\_VAR2, '<')+1, 3); -- valcode

SELECT CODE INTO V\_CODE FROM tbl\_sluise;

IF V\_CODE LIKE V\_RES1 THEN

SELECT VAL INTO URL FROM TBL\_SLUISE\_DOP\_PAR WHERE CODE LIKE 'URL';

XML\_PARSE (URL,v\_date);

SELECT CURMARKET INTO curm FROM VALUTE WHERE valcode = v\_res2;

delete from valute;

COMMIT;

v\_loc := null;

p\_out := 'RESULT\_CODE: <' || '0' || '>| RESULT\_TEXT: <' || 'succesfully' || '>| NAZN: VALCODE: <' || V\_RES1 || '>; CURMARKET: <' || curm || '>; CURDATE: <' || v\_date || '>;|';

return v\_loc;

ELSE

v\_loc := 'Preset code is not found';

p\_out := 'RESULT\_CODE: <' || '-1' || '>| RESULT\_TEXT: <' || 'Preset code is not found';

return v\_loc;

END IF;

exception

when no\_data\_found then

v\_loc := 'Preset valcode is not found';

p\_out := 'RESULT\_CODE: <' || '-1' || '>| RESULT\_TEXT: <' || 'Preset code is not found';

return v\_loc;

END docommand;

END XML\_PACK;

**ПРИМЕР:**

declare

st varchar2(1000);

st2 varchar2(100):='ACT:GET\_CURVAL\_<#CBR>| NAZN: VALCODE: <AUD>;|';

s varchar2(100);

begin

S:=XML\_PACK.DOCOMMAND(ST2,st);

dbms\_output.put\_line(st);

end;

**ВЫВОД:**

