package com.crimsonlogic.mnis.cpdv2.util;

import java.text.ParseException;

import java.text.SimpleDateFormat;

import java.util.Calendar;

import java.util.Date;

import java.util.Locale;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

/\*\*

\* http://docs.oracle.com/javase/7/docs/api/java/text/SimpleDateFormat.html

\* http://www.a2ztechguide.com/2011/08/java-date-utility-methods.html

\*

\* @author sachinpravind 23 Aug, 2013

\*

\*/

public class DateUtil {

Date now = new Date();

private static final Logger LOGGER = LoggerFactory.getLogger(DateUtil.class);

public String getCurrentDateinDDMYYYY() {

// Get various information from the Date object.

Calendar cal = Calendar.getInstance();

int day = cal.get(Calendar.DATE);

int month = cal.get(Calendar.MONTH) + 1;

int year = cal.get(Calendar.YEAR);

System.out.println("Current Date: " + cal.getTime());

System.out.println("Day: " + day);

System.out.println("Month: " + month);

System.out.println("Year: " + year);

return now.getDate() + "/" + now.getMonth() + "/" + Calendar.getInstance().get(Calendar.YEAR);

}

// Get current date in YYYYMMDDHHMiSS

public static String getCurrentDateinYYYYMMDDHHMiSS() {

// Get various information from the Date object.

SimpleDateFormat sdf = new SimpleDateFormat("yyyyMMddHHmmss");

Calendar cal = Calendar.getInstance();

String todaysDate = sdf.format(cal.getTime());

return todaysDate;

}

public String getStrDMONYYYY(final String strDate) {

SimpleDateFormat formatter = new SimpleDateFormat("dd-MMM-yyyy");

// String dateInString = "7-Jun-2013";

Date date = null;

try {

date = formatter.parse(strDate);

} catch (ParseException e) {

e.printStackTrace();

}

return formatter.format(date);

}

public Date getDMONYYYY(final String strDate) {

SimpleDateFormat formatter = new SimpleDateFormat("dd-MMM-yyyy");

// String dateInString = "7-Jun-2013";

Date date = null;

try {

date = formatter.parse(strDate);

System.out.println(date);

System.out.println(formatter.format(date));

} catch (ParseException e) {

e.printStackTrace();

}

return date;

}

/\*

\* Date validation using SimpleDateFormat it will take a string and make sure

\* it's in the proper format as defined by you

\*/

public static boolean isValidDate(final String date) {

/\*

\* set date format, this can be changed to whatever format you want,

\* MM-dd-yyyy, MM.dd.yyyy, dd.MM.yyyy etc. you can read more about it here:

\* http://java.sun.com/j2se/1.4.2/docs/api/index.html

\*/

SimpleDateFormat sdf = new SimpleDateFormat("dd/MM/yy", Locale.ENGLISH);

boolean isValid = true;

sdf.setLenient(false);

try {

sdf.parse(date);

} catch (Exception e) {

isValid = false;

}

// return true if date is valid otherwise return false

return isValid;

}

/\*\*

\* http://stackoverflow.com/questions/19184117/how-to-allow-date-as-00-00-0000

\* -in-regular-expression \*

\*

\* @param date

\* @return

\*/

public static boolean isValidFormatDate(final String date) {

String patternString = "\\d{2}/\\d{2}/\\d{4}";

Pattern pattern = Pattern.compile(patternString);

Matcher matcher = pattern.matcher(date);

return matcher.matches();

}

/\*\*

\* Test for 00/00/1927 00/00/1947 Issue 26190

\*

\* @param date

\* @return

\*/

public static boolean isValidFormatDate1(final String date) {

//String patternString = "^\\d{1,2}\\/\\d{1,2}\\/\\d{4}$";

String patternString = "^(0?[1-9]|[12][0-9]|3[01])[\\/\\-](0?[1-9]|1[012])[\\/\\-]\\d{4}$";

Pattern pattern = Pattern.compile(patternString);

Matcher matcher = pattern.matcher(date);

return matcher.matches();

}

/\*

\* Convert date string from one format to another format using

\* SimpleDateFormat. Here format of fromDate is dd/MM/yy

\*/

public static String convertDateFromOneFormatToOther(final String fromDate) {

String convertedDate = "";

try {

// create SimpleDateFormat object with source string date format

SimpleDateFormat sdfFrom = new SimpleDateFormat("dd/MM/yy");

sdfFrom.setLenient(false);

// parse the string into Date object

Date date = sdfFrom.parse(fromDate);

// create SimpleDateFormat object with desired date format

SimpleDateFormat sdfTo = new SimpleDateFormat("MM-dd-yyyy");

// parse the date into another format

convertedDate = sdfTo.format(date);

System.out.println("Date is converted from dd/MM/yy format to MM-dd-yyyy");

System.out.println("Converted date is : " + convertedDate);

} catch (ParseException pe) {

return ("Parse Exception : " + pe);

}

return convertedDate;

}

/\*

\* Method to parse date/time string into date. The return type of this method

\* is Date. This method throws ParseException if the given string cannot be

\* parsed as a date. todo : Cater for dd-MON-yyyy,dd-MMM-yyyy

\*/

public static Date convertStringToDate(final String dateString) {

Date date = null;

try {

// create SimpleDateFormat object with source string date format

SimpleDateFormat sdf = new SimpleDateFormat("dd/MM/yy");

sdf.setLenient(true);

// parse the string into Date object

date = sdf.parse(dateString);

DateUtil.LOGGER.debug(" Date Util :" + date);

} catch (ParseException pe) {

DateUtil.LOGGER.error("Parse Exception : " + pe);

}

return date;

}

public static Date convertStringToDateinDDMONYYYYFormat(final String dateString) {

Date date = null;

try {

// create SimpleDateFormat object with source string date format

SimpleDateFormat sdf = new SimpleDateFormat("dd-MMM-yyyy");

sdf.setLenient(true);

// parse the string into Date object

date = sdf.parse(dateString);

DateUtil.LOGGER.debug(" Date Util :" + date);

} catch (ParseException pe) {

DateUtil.LOGGER.error("Parse Exception : " + pe);

}

return date;

}

/\*\*

\* 03-05-1978 format

\*

\* @param dateString

\* @return

\*/

public static Date convertStringToDateinDD\_MM\_YYYYFormat(final String dateString) {

Date date = null;

try {

// create SimpleDateFormat object with source string date format

SimpleDateFormat sdf = new SimpleDateFormat("dd-MM-yyyy");

sdf.setLenient(true);

// parse the string into Date object

date = sdf.parse(dateString);

DateUtil.LOGGER.debug(" Date Util :" + date);

} catch (ParseException pe) {

DateUtil.LOGGER.error("Parse Exception : " + pe);

}

return date;

}

/\*\*

\* Convert todays Date in desired format

\*

\* @param pArgs

\* @throws Exception

\*/

public void convertTodatsDate() {

Calendar cal = Calendar.getInstance();

/\*

\* set date format, this can be changed to whatever format you want,

\* MM-dd-yyyy, MM.dd.yyyy, dd.MM.yyyy etc. you can read more about it here:

\* http://java.sun.com/j2se/1.4.2/docs/api/index.html

\*/

SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");

System.out.print(sdf.format(cal.getTime()));

}

/\*\*

\* Add a Month

\*

\* @param pArgs

\* @throws Exception

\*/

public static String addMonthInDate(final int month) {

String date = "";

Calendar now = Calendar.getInstance();

now.add(Calendar.MONTH, month);

date = date + now.get(Calendar.DATE) + "-" + (now.get(Calendar.MONTH) + 1) + "-" + now.get(Calendar.YEAR);

return date;

}

/\*\*

\* Subtract Month

\*

\* @param pArgs

\* @throws Exception

\*/

public static String subtractMonthInDate(final int month) {

String date = "";

Calendar now = Calendar.getInstance();

now.add(Calendar.MONTH, new Integer("-" + month));

date = date + now.get(Calendar.DATE) + "-" + (now.get(Calendar.MONTH) + 1) + "-" + now.get(Calendar.YEAR);

return date;

}

/\*\*

\* Get Current MonthName

\*

\* @param pArgs

\* @throws Exception

\*/

public String getCurrentMonthName() {

String[] monthName = { "January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December" };

Calendar cal = Calendar.getInstance();

return monthName[cal.get(Calendar.MONTH)];

}

/\*\*

\* Convert Date object to String of desired format

\*

\* @param pArgs

\* @throws Exception

\*/

public String convertDateToString(final Date date) {

// Date dateNow = new Date();

// SimpleDateFormat sdfYYYYMMDD = new SimpleDateFormat("yyyyMMdd");

// SimpleDateFormat sdfMMDDYYYY = new SimpleDateFormat("MMddyyyy");

// SimpleDateFormat sdfddMMMyyyy = new SimpleDateFormat("dd-MMM-yyyy");

// String YYYYMMDD = sdfYYYYMMDD.format(dateNow);

// String MMDDYYYY = sdfMMDDYYYY.format(dateNow);

// String ddMMMyyyy = sdfddMMMyyyy.format(dateNow);

// System.out.println("In YYYYMMDD: '" + YYYYMMDD + "'");

// System.out.println("In MMDDYYYY: '" + MMDDYYYY + "'");

// System.out.println("In ddMMMyyyy: '" + ddMMMyyyy + "'");

String strDate = "";

try {

strDate = convertDateToStringMMDDYYYY(date);

} catch (Exception ex) {

DateUtil.LOGGER.debug(" Exception while formatting : " + ex.getMessage());

}

if (strDate.isEmpty()) {

try {

strDate = convertDateToStringDDMMYYYY(date);

} catch (Exception ex) {

DateUtil.LOGGER.debug(" Exception while formatting : " + ex.getMessage());

}

}

return strDate;

}

public static String convertDateToStringMMDDYYYY(final Date date) {

SimpleDateFormat sdf = new SimpleDateFormat("MM/dd/yyyy");

String strDate = sdf.format(date);

DateUtil.LOGGER.debug(" convertDateToString : " + strDate);

return strDate;

}

public static String convertDateToStringDDMMYYYY(final Date date) {

SimpleDateFormat sdf = new SimpleDateFormat("dd/MM/yyyy");

sdf.setLenient(false);

String strDate = sdf.format(date);

DateUtil.LOGGER.debug(" convertDateToString : " + strDate);

return strDate;

}

public static String convertDateToStringDD\_MM\_YYYY(final Date date) {

SimpleDateFormat sdf = new SimpleDateFormat("dd-MMM-yyyy");

sdf.setLenient(false);

String strDate = sdf.format(date);

DateUtil.LOGGER.debug(" convertDateToString : " + strDate);

return strDate;

}

public static String convertDateToStringDD\_M\_YYYY(final Date date) {

SimpleDateFormat sdf = new SimpleDateFormat("dd-MM-yyyy");

sdf.setLenient(false);

String strDate = sdf.format(date);

DateUtil.LOGGER.debug(" convertDateToString : " + strDate);

return strDate;

}

public static String convertDateToStringDD\_MM\_YYYY\_new(final Date date) {

SimpleDateFormat sdf = new SimpleDateFormat("ddMMyyyy");

sdf.setLenient(false);

String strDate = null;

if (date != null) {

strDate = sdf.format(date);

}

DateUtil.LOGGER.debug(" convertDateToString : " + strDate);

return strDate;

}

public static int convertDateToYYYYYMMDD(final Date date) {

SimpleDateFormat sdf = new SimpleDateFormat("yyyyMMdd");

String strDate = sdf.format(date);

DateUtil.LOGGER.debug(" convertDateToYYYYYMMDD : " + strDate);

return Integer.valueOf(strDate);

}

public static int convertDateToDDMMYYYY(final Date date) {

SimpleDateFormat sdf = new SimpleDateFormat("ddMMyyyy");

String strDate = sdf.format(date);

DateUtil.LOGGER.debug(" convertDateToYYYYYMMDD : " + strDate);

return Integer.valueOf(strDate);

}

public static boolean compareWithCurrentDate(final Date date) {

Date dateNow = new Date();

DateUtil.LOGGER.debug(" input date : " + date);

return date.after(dateNow);

}

public static void main(final String[] pArgs) throws Exception {

// Convert String to Date

String date = "13\07\1961";

String invalidDate = "00/00/1978";

// Date d = convertStringToDate(date);

Date d = convertStringToDateinDDMONYYYYFormat("03-MAY-1978");

System.out.println(d);

d = convertStringToDateinDD\_MM\_YYYYFormat("03-05-1978");

System.out.println(d);

}

public static String convertDateToStringYYYY\_MM\_DD(final Date date) {

if (date == null) {

return "";

}

SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");

sdf.setLenient(false);

String strDate = sdf.format(date);

DateUtil.LOGGER.debug(" convertDateToString : " + strDate);

return strDate;

}

public static long getTodayInMillis() {

Calendar cal = Calendar.getInstance();

cal.set(Calendar.HOUR\_OF\_DAY, 0); // set hour to midnight

cal.set(Calendar.MINUTE, 0); // set minute in hour

cal.set(Calendar.SECOND, 0); // set second in minute

cal.set(Calendar.MILLISECOND, 0);

return cal.getTimeInMillis();

}

public static String reverseDate(final String date) {

if (date != null) {

Date dateNow = null;

SimpleDateFormat sdf = new SimpleDateFormat("dd-MM-yyyy");

try {

dateNow = sdf.parse(date);

} catch (ParseException e) {

e.printStackTrace();

}

SimpleDateFormat sdf1 = new SimpleDateFormat("yyyyMMdd");

String strDate = sdf1.format(dateNow);

// DateUtil.LOGGER.debug("Before: " + date + " After converted : " + strDate);

return strDate;

} else {

return date;

}

}

public static Date getDateDMONYYYY(final String strDate) {

DateUtil du = new DateUtil();

return du.getDMONYYYY(strDate);

}

public static String convertDateToDD\_MM\_YYYY(final Date date) {

if (date != null) {

SimpleDateFormat sdf = new SimpleDateFormat("dd-MM-yyyy");

String strDate = sdf.format(date);

DateUtil.LOGGER.debug(" convertDateToYYYYYMMDD : " + strDate);

return strDate;

} else {

return null;

}

}

}