

รายการอ้างอิง



ภาษาไทย

กระทรวงสาธารณสุข, ศูนย์ติดตามอาการอันไม่พึงประสงค์จากการใช้ยา. International Adverse Drug Reaction Monitoring. กรุงเทพมหานคร: โรงพิมพ์ชุมนุมสหกรณ์การเกษตรแห่งประเทศไทยจำกัด, 2528.

กำพล ศรีวัฒนกุล, บรรณาธิการ. คู่มือการใช้ยาฉบับสมบูรณ์. พิมพ์ครั้งที่ 1. กรุงเทพมหานคร: บริษัทเมดาร์ทจำกัด, 2527.

บุตร ประดิษฐ์วิชัย. แพทย์-ยา วิชาชีพหรือการค้า. กรุงเทพมหานคร: สำนักพิมพ์หมอชาวบ้าน, 2532.

มหาวิทยาลัยมหิดล, คณะเภสัชศาสตร์. ตำราเภสัชวิทยา. เล่มที่ 1. พิมพ์ครั้งที่ 1. กรุงเทพมหานคร: ห้างหุ้นส่วนอักษรบัณฑิต, 2528.

ภาษาอังกฤษ

Date, C.J. Introduction to Database Management System. Vol. 1, 4th ed. Reading, MA: Addison-Wesley, 1986.

Davies, D.M. Textbook of Adverse Drug Reactions. 3rd ed. New York: Oxford University, 1986.

Elmasri, R. and Navathe, S.B. Fundamentals of Database System. Redwood City, CA: The Benjamin/Cummings, 1989.

Fleming, C.C. and Halle, B.V. Handbook of Relational Database Design. Reading, MA: Addison-Wesley, 1989.

Korth, H.F. and Silberschatz, A. Database System Concepts. New York: McGraw-Hill, 1986.

Salton, G. Automatic Text Processing, The Transformation, Analysis and Retrieval of Information by Computer. Reading, MA: Addison-Wesley, 1989.

Senn, J.A. Analysis and Design of Information Systems. New York: McGraw-Hill, 1989.

Tenopir, C. and Lundeen, G. Managing your Information How to Design and Create a Textual Database on your Microcomputer. New York: Neal-Schuman, 1988.

ภาคผนวก ก

โปรแกรมบรรจุข้อมูลต้นฉบับ

```

/* Loading data Example source code */

*-----*
* System : ADR Retrieval System
* Function: Process Loading Data
*-----*
function lod_prce
  local fn_array:={},ft_array:={},fn_list:='',i:=0
  local fn_max:=0,fn_comma:=0,fn_crlf:=0,nl:=chr(13)+chr(10)
  fn_list:=memoread('load.cfg')
  do while at(',',fn_list)>0
    fn_max++
    fn_crlf:=at(nl,fn_list)
    fn_comma:=at(',',fn_list)
    aadd(fn_array,left(fn_list,fn_comma-1))
    aadd(ft_array,substr(fn_list,fn_comma+1,fn_crlf-fn_comma-1))
    fn_list:=substr(fn_list,fn_crlf+2)
  enddo
  for i:=1 to fn_max
    if file(fn_array[i])
      do case
        case upper(ft_array[i])='REFER'
          lod_refr(fn_array[i])
        case upper(ft_array[i])='NAME'
          lod_name(fn_array[i])
        case upper(ft_array[i])='TERM'
          lod_term(fn_array[i])
        case upper(ft_array[i])='ADR'
          lod_adrx(fn_array[i])
      endcase
    endif
  next
  save to adr.cfg all like s_*
return nil
*-----*

*-----*
* System : ADR Retrieval System
* Function: Cut String at Chars
*-----*
static function cut_read(cut_string,cut_chars)
  local cut_point:=0,ret_string:=''
  cut_point:=at(cut_chars,cut_string)
  if cut_point>0
    ret_string:=left(cut_string,cut_point-1)
    cut_string:=substr(cut_string,cut_point+len(cut_chars))
  else
    ret_string:=cut_string
    cut_string:=''
  endif
return ret_string
*-----*

*-----*
* System : ADR Retrieval System
* Function: Loading Refer Data
*-----*
static function lod_refr(fn_refr)

```

```

local fn_buffer:=space(1024),fn_data:='',fn_line:='',fn_handle:=0
local nl:=chr(13)+chr(10)
fn_handle:=fopen(fn_refr,0)
use refer.dat index refrlc,refrln new
do while fread(fn_handle,@fn_buffer,1024)>0
  fn_data+=fn_buffer
  fn_buffer:=space(1024)
  do while at(nl,fn_data)>0
    fn_line:=cut_read(@fn_data,nl)
    if len(trim(fn_line))>0
      refer->(dbappend())
      refer->refr_code:=upper(cut_read(@fn_line,','))
      refer->refr_name:=cut_read(@fn_line,',')
      refer->refr_srce:=cut_read(@fn_line,',')
      refer->refr_year:=cut_read(@fn_line,',')
    endif
  enddo
enddo
fclose(fn_handle)
close databases
return nil
*-----*

*-----*
* System : ADR Retrieval System
* Function: Loading Name Data
*-----*
static function lod_name(fn_name)
local fn_buffer:=space(1024),fn_data:='',fn_line:='',fn_handle:=0
local nl:=chr(13)+chr(10)
fn_handle:=fopen(fn_name,0)
use combine.dat index comblc,new
use name.dat index nameic,namein new
do while fread(fn_handle,@fn_buffer,1024)>0
  fn_data+=fn_buffer
  fn_buffer:=space(1024)
  do while at(nl,fn_data)>0
    fn_line:=cut_read(@fn_data,nl)
    if len(trim(fn_line))>0
      name->(dbappend())
      name->drug_code:=upper(cut_read(@fn_line,','))
      name->drug_name:=cut_read(@fn_line,',')
      name->drug_type:=upper(cut_read(@fn_line,','))
      do while len(trim(fn_line))>0
        combine->(dbappend())
        combine->drug_code:=name->drug_code
        combine->form_code:=upper(cut_read(@fn_line,','))
      enddo
    endif
  enddo
enddo
fclose(fn_handle)
close databases
return nil
*-----*

*-----*
* System : ADR Retrieval System

```

```

* Function: Loading ADR Terms
*-----*
static function lod_term(fn_term)
  local fn_buffer:=space(1024),fn_data:='',fn_line:='',fn_handle:=0
  local nl:=chr(13)+chr(10)
  fn_handle:=fopen(fn_term,0)
  use term.dat index term1c,term1n new
  do while fread(fn_handle,@fn_buffer,1024)>0
    fn_data+=fn_buffer
    fn_buffer:=space(1024)
    do while at(nl,fn_data)>0
      fn_line:=cut_read(@fn_data,nl)
      if len(trim(fn_line))>0
        term->(dbappend())
        term->term_code:=strtran(str(s_termno++,6),' ','0')
        term->term_name:=cut_read(@fn_line,',')
      endif
    enddo
  enddo
  fclose(fn_handle)
  close databases
return nil
*-----*

*-----*
* System : ADR Retrieval System
* Function: Loading ADR/Interaction Data
*-----*
static function lod_adrx(fn_adrx)
  local fn_buffer:=space(1024),fn_data:='',fn_line:='',fn_handle:=0
  local refrcode:=space(4),datacode:=space(8),drugcode:=space(10)
  local datatext:='',termcode:='',nl:=chr(13)+chr(10)
  local intrcode:={},termcode:={},termname:={}
  fn_handle:=fopen(fn_adrx,0)
  use drug.dat index drug1c,drug2c,drug3c new
  use term.dat index term1c,term1n new
  use adrx.dat index adrx1c new
  use adry.dat index adry1c new
  use adrz.dat index adrz1c new
  do while fread(fn_handle,@fn_buffer,1024)>0
    fn_data+=fn_buffer
    fn_buffer:=space(1024)
    do while at(nl,fn_data)>0
      fn_line:=cut_read(@fn_data,nl)
      if len(trim(fn_line))>0
        do case
          case upper(left(fn_line,3))='.RM'
          case upper(left(fn_line,3))='.RF'
            refrcode:=upper(substr(fn_line,at('[',fn_line)+1,4))
          case upper(left(fn_line,3))='.DN'
            if drugcode!=space(10)
              upd_term(termname)
              idx_adrx(datatext,termcode)
              datatext:=pak_adrx(datatext)
              upd_adrx(refrcode,datacode,drugcode,datatext,intrcode,termcode)
            endif
          default
            datatext:=''
            intrcode:={}
        endcase
      endif
    enddo
  enddo

```

```

        termcode:={ }
        termname:={ }
        datacode:=strtran(str(s_datano++), ' ', '0')
        tempcode:=cut_read(@fn_line, '')
        drugcode:=upper(cut_read(@fn_line, ''))
    case upper(left(fn_line,3))='.IN'
        tempcode:=cut_read(@fn_line, '')
        aadd(intrcode, upper(cut_read(@fn_line, '')))
    case upper(left(fn_line,3))='.KW'
        tempcode:=cut_read(@fn_line, '')
        do while len(trim(fn_line))>0
            aadd(termname, upper(cut_read(@fn_line, '')))
        enddo
    otherwise
        datatext+=fn_line+nl
    endcase
endif
enddo
enddo
upd_term(termname)
idx_adrx(datatext, termcode)
datatext:=pak_adrx(datatext)
upd_adrx(refrcode, datacode, drugcode, datatext, intrcode, termcode)
fclose(fn_handle)
close databases
return nil
*-----*

*-----*
* System : ADR Retrieval System
* Function: Update ADRs Terms
*-----*
static function upd_term(termname)
    local i, max
    if len(termname)>0
        max:=len(termname)
        for i:=1 to max
            term->(dbappend())
            term->term_code:=strtran(str(s_termno++), ' ', '0')
            term->term_name:=upper(termname[i])
        next
    endif
return nil
*-----*

*-----*
* System : ADR Retrieval System
* Function: Index ADRs Terms
*-----*
static function idx_adrx(srcetext, termcode)
    local NL:=chr(13)+chr(10)
    select term
    set order to 0
    srcetext:=strtran(srcetext, NL, ' ')
    srcetext:=strtran(srcetext, ' ', ' ')
    srcetext:=upper(strtran(srcetext, ' ', ' '))
    go top
    do while !eof()

```

```

        if trim(term->term_name)$arcetext
            aadd(terrcode,term->term_code)
        endif
        term->(dbskip())
    enddo
    set order to 1
    select adrz
return nil
*-----*
*-----*
* System : ADR Retrieval System
* Function: Update ADRs Data
*-----*
static function upd_adrx(refrcode,datacode,drugcode,datatext,intrcode,termcode)
    local i,max
    select drug
    if len(intrcode)>0
        max:=len(intrcode)
        for i:=1 to max
            drug->(dbappend())
            drug->drug_code:=drugcode
            drug->intr_code:=intrcode[i]
            drug->data_code:=datacode
        next
    else
        drug->(dbappend())
        drug->drug_code:=drugcode
        drug->data_code:=datacode
    endif
    select adrx
    max:=len(termcode)
    for i:=1 to max
        adrx->(dbappend())
        adrx->term_code:=termcode[i]
        adrx->data_code:=datacode
    next
    select adry
    adry->(dbappend())
    adry->data_code:=datacode
    adry->refr_code:=refrcode
    select adrz
    max:=int(len(datatext)/60)+1
    for i:=1 to max
        adrz->(dbappend())
        adrz->data_code:=datacode
        adrz->data_line:=strtran(str(i,4),' ','0')
        adrz->data_text:=substr(datatext,(i-1)*60+1,60)
    next
    select adrz
return nil
*-----*

```

ภาคผนวก ข
โปรแกรมสืบค้นข้อมูลอาคารไม่พึงประสงค์ฉบับ


```

/* Searching data Example source code */

*-----*
* System : ADR Retrieval System
* Function: Process Search Adverse Reactions Data
*-----*
function prc_adrs(r_file,chckcode)
  local tempcode,r_info:='',r_title,r_foot,i,j,max,NL:=chr(13)+chr(10)
  r_title:='*** SEARCH ADVERSE REACTIONS DATA ***'
  r_foot :='*** Process by '+s_dept
  r_info +=repl('=',80)+NL+space((80-len(r_title))/2)+r_title+NL+repl('=',80)+NL
  set print to {r_file}
  set print on
  set console off
  qqout(r_info)
  max:=len(chckcode)
  for i:=1 to max
    if chckcode[i]!=space(10)
      if sch_name(chckcode[i])='C'
        prc_adrx(chckcode[i])
        if combine->(dbseek(chckcode[i],.f.))
          do while combine->drug_code==chckcode[i]
            tempcode:=combine->form_code
            sch_name(@tempcode)
            prc_adrx(tempcode)
            combine->(dbskip())
          enddo
        else
          qqout('*** Combine code drug lists not found ***'+NL)
        endif
      else
        prc_adrx(chckcode[i])
      endif
      qqout(repl('-',80)+NL)
    endif
    j:=inkey()
    if j==27
      qqout('*** Process interrupted by press [Esc] key ***'+NL)
      exit
    endif
  next
  qqout(r_foot+NL+NL+NL)
  set print off
  set print to
  set console on
return nil
*-----*

*-----*
* System : ADR Retrieval System
* Function: Process Search One Drug Adverse Reactions Data
*-----*
static function prc_adrx(schxcode)
  local r_info:='',c_long,datacode,o_select,NL:=chr(13)+chr(10)
  o_select:=select()
  c_long :=len(trim(schxcode))
  select drug
  if c_long>2.and.dbseek(left(schxcode,3)+space(7),.f.)

```

```

do while drug->drug_code==left(schxcode,3)+space(7)
  if drug->intr_code==space(10)
    datacode:=drug->data_code
    prc_adry(datacode)
  endif
  drug->(dbskip())
enddo
endif
if c_long>4.and.dbseek(left(schxcode,5)+space(5),.f.)
do while drug->drug_code==left(schxcode,5)+space(5)
  if drug->intr_code==space(10)
    datacode:=drug->data_code
    prc_adry(datacode)
  endif
  drug->(dbskip())
enddo
endif
if c_long>6.and.dbseek(left(schxcode,7)+space(3),.f.)
do while drug->drug_code==left(schxcode,7)+space(3)
  if drug->intr_code==space(10)
    datacode:=drug->data_code
    prc_adry(datacode)
  endif
  drug->(dbskip())
enddo
endif
if c_long>7.and.dbseek(left(schxcode,8)+space(2),.f.)
do while drug->drug_code==left(schxcode,8)+space(2)
  if drug->intr_code==space(10)
    datacode:=drug->data_code
    prc_adry(datacode)
  endif
  drug->(dbskip())
enddo
endif
if c_long>9.and.dbseek(schxcode,.f.)
do while drug->drug_code==schxcode
  if drug->intr_code==space(10)
    datacode:=drug->data_code
    prc_adry(datacode)
  endif
  drug->(dbskip())
enddo
endif
select (o_select)
return nil
*-----*

*-----*
* System   : ADR Retrieval System
* Function: Search One Adverse Reactions Data
*-----*

static function sch_name(srchcode)
local r_info:='',r_return,r_select,NL:=chr(13)+chr(10)
r_select:=select()
select name
if dbseek(srchcode,.f.)
  qqout([''+srchcode+' ']+name->drug_name+', '+drg_type(name->drug_type)+NL)

```

```

if right(trim(srchcode),1)=='S'
  srchcode:=left(left(srchcode,len(trim(srchcode))-1)+space(10),10)
  if dbseek(srchcode,.f.)
    qqout([''+srchcode+'] '+name->drug_name+', '+drg_type(name->drug_type)+NL)
    r_return:=name->drug_type
  else
    qqout([''+srchcode+'] search code not found'+NL)
    r_return:='X'
  endif
else
  r_return:=name->drug_type
endif
else
  qqout([''+srchcode+'] search code not found'+NL)
  r_return:='X'
endif
select (r_select)
return r_return
*-----*

*-----*
* System : ADR Retrieval System
* Function: Drug Items Type
*-----*
static function drg_type(dtype)
  local rtype:=space(18)
  do case
  case dtype='M'
    rtype:='Drug Main Groups '
  case dtype='N'
    rtype:='Drug Sub Groups '
  case dtype='O'
    rtype:='Drug Others Groups'
  case dtype='G'
    rtype:='Drug Generic Name '
  case dtype='T'
    rtype:='Drug Trad Name '
  case dtype='C'
    rtype:='Combined Drug Name'
  case dtype='S'
    rtype:='Drug Synonyms Name'
  case dtype='D'
    rtype:='Diseases/Syndromes'
  case dtype='F'
    rtype:='Foods/Nutrients '
  case dtype='A'
    rtype:='Allergic drugs '
  case dtype='I'
    rtype:='Often used drugs '
  endcase
  return rtype
*-----*

*-----*
* System : ADR Retrieval System
* Function: Search One Adverse Reactions Data
*-----*
static function prc_adry(datacode)

```

```

local r_info:='',tempinfo:='',tempcode,o_select,NL:=chr(13)+chr(10)
o_select:=select()
select adry
if dbseek(datacode,.f.)
  tempcode:=adry->refr_code
  select refer
  if dbseek(tempcode,.f.)
    qqout('['+tempcode+' '+refer->refr_name+NL)
    qqout(space(7)+trim(refer->refr_srce)+' '+refer->refr_year+NL)
  else
    qqout('['+tempcode+' *** Reference code not found ***'+NL)
  endif
select adrz
if dbseek(datacode,.f.)
  do while adrz->data_code==datacode
    tempinfo+=trim(adrz->data_text)
    adrz->(dbskip())
  enddo
  tempinfo:=unp_adrx(tempinfo)
  qqout(tempinfo+NL)
else
  qqout('['+datacode+' *** Adverse reactions data not found ***'+NL)
endif
select adry
else
  qqout('['+datacode+' data code not found'+NL)
endif
select (o_select)
return nil
*-----*

*-----*
* System : ADR Retrieval System
* Function: Process Interactions Check
*-----*

function prc_intr(r_file,chckcode)
local r_info:='',r_title,r_foot,i,j,max,chcklist:={},chck2list:={},intrlist:='',NL:=chr(13)+chr(10)
r_title:='*** SEARCH INTERACTIONS DATA ***'
r_foot :='*** Process by '+s_dept
r_info +=repl('=',80)+NL+space((80-len(r_title))/2)+r_title+NL+repl('=',80)+NL
set print to (r_file)
set print on
set console off
qqout(r_info)
max:=len(chckcode)
for i:=1 to max
  if chckcode[i]!=space(10)
    aadd(chcklist,chckcode[i])
  endif
endif
next
max:=len(chcklist)
if max=1
  if sch_name(chcklist[1])=='C'
    if combine->(dbseek(chcklist[1],.f.))
      do while combine->drug_code==chcklist[1]
        tempcode:=combine->form_code
        qqout('->')
        sch_name(@tempcode)
      enddo
    endif
  endif
endif

```

```

        aadd(chcklist,tempcode)
        combine->(dbskip())
    enddo
else
    qqout('*** Combine code drug lists not found ***'+NL)
endif
max:=len(chcklist)
for i:=1 to max
    lst_intr(chcklist[i])
    j:=inkey()
    if j=27
        qqout('*** Process interrupted by press [Esc] key ***'+NL)
        exit
    endif
next
else
    lst_intr(chcklist[1])
endif
qqout(repl('-',80)+NL)
elseif max>1
for i:=1 to max
    aadd(chck2list,chcklist[i])
    if sch_name(chcklist[i])='C'
        if combine->(dbseek(chcklist[i],.f.))
            do while combine->drug_code==chcklist[i]
                tempcode:=combine->form_code
                qqout('->')
                sch_name(@tempcode)
                aadd(chck2list,tempcode)
                combine->(dbskip())
            enddo
        else
            qqout('*** Combine code drug lists not found ***'+NL)
        endif
    endif
next
max:=len(chck2list)
qqout(NL+'*** CHECK DRUG INTERACTIONS ***'+NL)
qqout(repl('-',80)+NL)
for i:=1 to max-1
    intrlist:=''
    for j:=i+1 to max
        intrlist+=chck2list[j]+';'
    next
    chk_intr(chck2list[i],intrlist)
    j:=inkey()
    if j=27
        qqout('*** Process interrupted by press [Esc] key ***'+NL)
        exit
    endif
next
qqout(repl('-',80)+NL)
endif
qqout(r_foot+NL+NL)
set print off
set print to
set console on
return nil

```

```

*-----*

*-----*
* System : ADR Retrieval System
* Function: List Interactions Drugs
*-----*
static function lst_intr(chckcode)
  local c_long,tempcode,intrcode,datacode,o_select,NL:=chr(13)+chr(10)
  o_select:=select()
  qqout(NL+'*** LIST DRUG INTERACTIONS ***'+NL)
  qqout(repl('-',80)+NL)
  select drug
  c_long :=len(trim(chckcode))
  tempcode:=left(chckcode,3)+space(7)
  if c_long>2.and.dbseek(tempcode,.f.)
    do while drug->drug_code==tempcode
      if drug->intr_code!=space(10)
        intrcode:=drug->intr_code
        datacode:=drug->data_code
        sch_name(tempcode)
        sch_name(intrcode)
        prc_adry(datacode)
      endif
      drug->(dbskip())
    enddo
  endif
  tempcode:=left(chckcode,5)+space(5)
  if c_long>4.and.dbseek(tempcode,.f.)
    do while drug->drug_code==tempcode
      if drug->intr_code!=space(10)
        intrcode:=drug->intr_code
        datacode:=drug->data_code
        sch_name(tempcode)
        sch_name(intrcode)
        prc_adry(datacode)
      endif
      drug->(dbskip())
    enddo
  endif
  tempcode:=left(chckcode,7)+space(3)
  if c_long>6.and.dbseek(tempcode,.f.)
    do while drug->drug_code==tempcode
      if drug->intr_code!=space(10)
        intrcode:=drug->intr_code
        datacode:=drug->data_code
        sch_name(tempcode)
        sch_name(intrcode)
        prc_adry(datacode)
      endif
      drug->(dbskip())
    enddo
  endif
  tempcode:=left(chckcode,8)+space(2)
  if c_long>7.and.dbseek(tempcode,.f.)
    do while drug->drug_code==tempcode
      if drug->intr_code!=space(10)
        intrcode:=drug->intr_code
        datacode:=drug->data_code

```

```

        sch_name(tempcode)
        sch_name(intrcode)
        prc_adry(datacode)
    endif
    drug->(dbskip())
enddo
endif
tempcode:=chckcode
if c_long>9.and.dbseek(tempcode,.f.)
do while drug->drug_code==tempcode
if drug->intr_code!=space(10)
intrcode:=drug->intr_code
datacode:=drug->data_code
sch_name(tempcode)
sch_name(intrcode)
prc_adry(datacode)
endif
drug->(dbskip())
enddo
endif
select (o_select)
return nil
*-----*

*-----*
* System : ADR Retrieval System
* Function: Check Interactions Drugs
*-----*

static function chk_intr(chckcode,chcklist)
local c_long,tempcode,intrcode,datacode,o_select,NL:=chr(13)+chr(10)
o_select:=select()
select drug
c_long :=len(trim(chckcode))
tempcode:=left(chckcode,3)+space(7)
if c_long>2.and.dbseek(tempcode,.f.)
do while drug->drug_code==tempcode
if trim(drug->intr_code)$chcklist
intrcode:=drug->intr_code
datacode:=drug->data_code
sch_name(tempcode)
sch_name(intrcode)
prc_adry(datacode)
endif
drug->(dbskip())
enddo
endif
tempcode:=left(chckcode,5)+space(5)
if c_long>4.and.dbseek(tempcode,.f.)
do while drug->drug_code==tempcode
if trim(drug->intr_code)$chcklist
intrcode:=drug->intr_code
datacode:=drug->data_code
sch_name(tempcode)
sch_name(intrcode)
prc_adry(datacode)
endif
drug->(dbskip())
enddo

```

```

endif
tempcode:=left(chckcode,7)+space(3)
if c_long>6.and.dbseek(tempcode,.f.)
  do while drug->drug_code==tempcode
    if trim(drug->intr_code)$chcklist
      intrcode:=drug->intr_code
      datacode:=drug->data_code
      sch_name(tempcode)
      sch_name(intrcode)
      prc_adry(datacode)
    endif
    drug->(dbskip())
  enddo
endif
tempcode:=left(chckcode,8)+space(2)
if c_long>7.and.dbseek(tempcode,.f.)
  do while drug->drug_code==tempcode
    if trim(drug->intr_code)$chcklist
      intrcode:=drug->intr_code
      datacode:=drug->data_code
      sch_name(tempcode)
      sch_name(intrcode)
      prc_adry(datacode)
    endif
    drug->(dbskip())
  enddo
endif
tempcode:=chckcode
if c_long>9.and.dbseek(tempcode,.f.)
  do while drug->drug_code==tempcode
    if trim(drug->intr_code)$chcklist
      intrcode:=drug->intr_code
      datacode:=drug->data_code
      sch_name(tempcode)
      sch_name(intrcode)
      prc_adry(datacode)
    endif
    drug->(dbskip())
  enddo
endif
select (o_select)
return nil
*-----*

*-----*
* System : ADR Retrieval System
* Function: Process Adverse Reactions Drug List
*-----*
function prc_drug(r_file,chckcode)
  local r_info:='',r_title,r_foot,o_select,i,max,NL:=chr(13)+chr(10)
  local codelist:={},namelist:={},datalist:={},data2list:={}
  local drugcode,intrcode,datacode
  r_title:='*** SEARCH ADVERSE REACTIONS DRUG LIST ***'
  r_foot :='*** Process by '+s_dept
  r_info +=repl('=',80)+NL+space((80-len(r_title))/2)+r_title+NL+repl('=',80)+NL
  set print to (r_file)
  set print on
  set console off

```



```

qqout(r_info)
o_select:=select()
max:=len(chckcode)
select term
qqout('*** SEARCH TERMS LIST ***'+NL)
for i:=1 to max
  if chckcode[i]!=""
    if dbseek(chckcode[i],.f.)
      aadd(codelist,chckcode[i])
      aadd(namelist,term->term_name)
      qqout('['+chckcode[i]+' ] '+term->term_name+NL)
    else
      qqout('['+chckcode[i]+' ] term name not found'+NL)
    endif
  endif
endif
next
qqout(repl('-',80)+NL)
max:=len(codelist)
select adrx
for i:=1 to max
  aadd(datalist,{})
  if dbseek(codelist[i],.f.)
    do while adrx->term_code==codelist[i]
      aadd(datalist[i],adrx->data_code)
      adrx->(dbskip())
    enddo
  else
    aadd(datalist[i],repl('X',8))
  endif
endif
next
max:=len(datalist)
data2list:=intersac(datalist)
max:=len(data2list)
if max>0
  select drug
  set order to 3
  for i:=1 to max
    if dbseek(data2list[i],.f.)
      drugcode:=drug->drug_code
      intrcode:=drug->intr_code
      datacode:=drug->data_code
      sch_name(drugcode)
      if intrcode!=""
        sch_name(intrcode)
      endif
      prc_adry(datacode)
      j:=inkey()
      if j=27
        qqout('*** Process interrupted by press [Esc] key ***'+NL)
        exit
      endif
    else
      qqout('*** No Adverse reactions data found'+NL)
    endif
  next
  set order to 1
else
  qqout('*** No Adverse reactions data match'+NL)
endif

```

```

endif
select (o_select)
qqout(repl('-',80)+NL)
qqout(r_foot+NL+NL)
set print off
set print to
set console on
return nil
*-----*

*-----*
* System : ADR Retrieval System
* Function: Intersaction data code list
*-----*
static function intersac(chcklist)
  local retnlist:={},tempcode,tempflag,i,j,i_max,j_max
  j_max:=len(chcklist)
  if j_max=0
    retnlist:={}
  elseif j_max=1
    retnlist:=chcklist[1]
  else
    i_max:=len(chcklist[1])
    for i:=1 to i_max
      tempcode:=chcklist[1,i]
      tempflag:=.t.
      for j:=2 to j_max
        tempflag:=tempflag.and.ascan(chcklist[j],tempcode)!=0
      next
      if tempflag
        aadd(retnlist,tempcode)
      endif
    next
  endif
return retnlist
*-----*

*-----*
* System : ADR Retrieval System
* Function: Process Patient Prescriptions
*-----*
function prc_patn(r_file,chckcode,chcktype,patncode,patnname,patnsex,patnage)
  local r_info:='',r_title,r_foot,o_select,codelist:={},typelist:={},NL:=chr(13)+chr(10)
  local i,j,i_max,j_max,tempcode,chcklist:={},chck2list:={},intrlist:=''
  r_title:='*** PATIENT PRESCRIPTIONS ***'
  r_foot := '*** Process by '+s_dept
  r_info +=repl('-',80)+NL+space((80-len(r_title))/2)+r_title+NL+repl('-',80)+NL
  set print to (r_file)
  set print on
  set console off
  qqout(r_info)
  qqout([''+patncode+' ']+patnname+' Sex = '+patnsex+' Age = '+str(patnage,2)+' Years'+NL)
  qqout(NL+'*** PATIENT PRIVATE DATA ***'+NL)
  o_select:=select()
  select profile
  if dbseek(patncode,.f.)
    do while profile->patn_code==patncode
      aadd(codelist,profile->drug_code)
    enddo
  endif
endfunction

```

```

        aadd(typelist,profile->drug_type)
        qqout([''+profile->drug_code+' '+name->(drg_name(profile->drug_code))+',';
            +drg_type(profile->drug_type)+NL)
        profile->(dbskip()) ~
    enddo
else
    qqout('No private patient data'+NL)
endif
qqout(repl('-',80)+NL)
qqout('*** SEARCH DRUG LIST ***'+NL)
i_max:=len(chckcode)
for i:=1 to i_max
    if chckcode[i]!=space(10)
        qqout([''+chckcode[i]+' '+name->(drg_name(chckcode[i]))+', '+drg_type(chcktype[i])+NL)
    endif
next
qqout(repl('-',80)+NL)
qqout('*** ALLERGIC DRUG LIST ***'+NL)
i_max:=len(codelist)
for i:=1 to i_max
    if typelist[i]='A'
        aadd(chck2list,codelist[i])
    endif
next
i_max:=len(chckcode)
for i:=1 to i_max
    if chckcode[i]!=space(10)
        tempcode:=chckcode[i]
        if chcktype[i]='C'
            if chk_ally(chck2list,tempcode)
                sch_name(tempcode)
            endif
            if combine->(dbseek(tempcode,.f.))
                do while combine->drug_code==chckcode[i]
                    tempcode:=combine->form_code
                    if chk_ally(chck2list,tempcode)
                        qqout([''+chckcode[i]+''+NL+'->')
                        sch_name(tempcode)
                    endif
                    combine->(dbskip())
                enddo
            else
                qqout('*** Combine code drug lists not found ***'+NL)
            endif
        else
            if chk_ally(chck2list,tempcode)
                sch_name(tempcode)
            endif
        endif
    endif
j:=inkey()
if j==27
    qqout('*** Process interrupted by press [Esc] key ***'+NL)
    exit
endif
next
qqout(repl('-',80)+NL)
qqout('*** CHECK DRUG INTERACTIONS ***'+NL)
i_max:=len(chckcode)

```

```

for i:=1 to i_max
  if chkckode[i]!=space(10)
    aadd(chcklist,chkckode[i])
  endif
next
i_max:=len(codelist)
for i:=1 to i_max
  if typelist[i]='I'.or.typelist[i]='D'
    aadd(chcklist,codelist[i])
  endif
next
i_max:=len(chcklist)
chck2list:={}
if i_max>1
  for i:=1 to i_max
    aadd(chck2list,chcklist[i])
    if sch_name(chcklist[i])='C'
      if combine->(dbseek(chcklist[i],.f.))
        do while combine->drug_code==chcklist[i]
          tempcode:=combine->form_code
          qqout('->')
          sch_name(@tempcode)
          aadd(chck2list,tempcode)
          combine->(dbskip())
        enddo
      else
        qqout('*** Combine code drug lists not found ***'+NL)
      endif
    endif
  next
  i_max:=len(chck2list)
  qqout(NL)
  for i:=1 to i_max-1
    intrlist:=''
    for j:=i+1 to i_max
      intrlist+=chck2list[j]+';'
    next
    chk_intr(chck2list[i],intrlist)
    j:=inkey()
    if j=27
      qqout('*** Process interrupted by press [Esc] key ***'+NL)
      exit
    endif
  next
else
  qqout('Not enough item for interactions check'+NL)
endif
qqout(repl('-',80)+NL)
qqout('*** SEARCH ADVERSE REACTIONS DATA ***'+NL)
i_max:=len(chckcode)
for i:=1 to i_max
  if chkckode[i]!=space(10)
    if sch_name(chckcode[i])='C'
      prc_adrx(chckcode[i])
      if combine->(dbseek(chckcode[i],.f.))
        do while combine->drug_code==chckcode[i]
          tempcode:=combine->form_code
          sch_name(@tempcode)
        enddo
      endif
    endif
  endif
enddo

```

```

        prc_adrx(tempcode)
        combine->(dbskip())
    enddo
    else
        qqout('*** Combine code drug lists not found ***'+NL)
    endif
    else
        prc_adrx(chckcode[i])
    endif
endif
j:=inkey()
if j==27
    qqout('*** Process interrupted by press [Esc] key ***'+NL)
    exit
endif
next
qqout(repl('-',80)+NL)
select (o_select)
qqout(r_foot+NL+NL)
set print off
set print to
set console on
return nil
*-----*

*-----*
* System : ADR Retrieval System
* Function: Check allergic drug
*-----*
static function chk_ally(chcklist,chckcode)
    local retnvalu:=.f.,tempcode,c_long
    c_long :=len(trim(chckcode))
    tempcode:=left(chckcode,3)+space(7)
    if c_long>2.and.ascan(chcklist,tempcode)>0
        retnvalu:=.t.
    endif
    tempcode:=left(chckcode,5)+space(5)
    if c_long>4.and.ascan(chcklist,tempcode)>0
        retnvalu:=.t.
    endif
    tempcode:=left(chckcode,7)+space(3)
    if c_long>6.and.ascan(chcklist,tempcode)>0
        retnvalu:=.t.
    endif
    tempcode:=left(chckcode,8)+space(2)
    if c_long>7.and.ascan(chcklist,tempcode)>0
        retnvalu:=.t.
    endif
    tempcode:=chckcode
    if c_long>9.and.ascan(chcklist,tempcode)>0
        retnvalu:=.t.
    endif
return retnvalu
*-----*

*-----*
* System : ADR Retrieval System
* Function: Return Drug Name

```

```
*-----*
static function drg_name(seekcode)
  set order to 1
  if len(trim(seekcode))!=0.and.dbseek(upper(seekcode))
    return name->drug_name
  endif
return space(40)
*-----*
```



ประวัติผู้เขียน

นายสุรศักดิ์ ยุติธรรมนนท์ เกิดเมื่อวันที่ 14 กุมภาพันธ์ พ.ศ. 2510 ที่ อำเภอเมือง จังหวัดสุราษฎร์ธานี สำเร็จการศึกษา ได้รับปริญญา เกษศาสตรบัณฑิต จากคณะเกษตรศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย เมื่อปีการศึกษา 2532 และได้เข้าศึกษาต่อระดับปริญญาโทบัณฑิตใน สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์ ภาควิชา วิศวกรรมคอมพิวเตอร์ คณะ วิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย เมื่อปีการศึกษา 2534