```
; Model name: HouseholdDemographics.nlogo
; Version: 3 (10 July 2015)
; Author: Philip Verhagen (VU University Amsterdam, Faculty of Humanities)

; This model is an appendix to the paper
; Verhagen, P., J. Joyce and M. Groenhuijzen 2015. 'Modelling the dynamics of
demography in the Dutch limes zone' in: Proceedings of LAC2014 Conference, Rome,
19-20 September 2014

; list of global variables
; [n-deaths] number of deaths per tick/year (an integer number)
; [f-deaths] number of adult female deaths per year (an integer number)
; [sum-age-at-death] the summed age of all humans who died (an integer number)
; [sum-n-children-at-death] the sum of the number of children per deceased adult
female per year (an integer number)
; [n-children-per-female] the number of children per deceased adult female per year
(an integer number)
; [n-born] the number of children born per year (an integer number)

globals [n-deaths f-deaths sum-age-at-death sum-n-children-at-death n-children-per-
female n-born]

; list of agent-sets
; [humans] are agents representing a single human
; [households] are agents representing a single household, containing a certain
number of humans

breed [humans human]
breed [households household]

; attributes for the agent-set 'humans':
; [age] records the age of each human in number of years (an integer number; =
number of ticks)
; [gender] records the gender of each human (a string; options are "F" (female) or
"M" (male))
; [fertility] records the fertility rate of a female human (a floating point number
between 0.0 and 1.0)
; [recruit] records the number of years that a recruited male human has served in
the army (an integer number)
; [widowed] records whether the human is widowed (a binary number 0/1)
; [n-children] records the number of children born to a human (an integer number;
recorded for females only?)
; [my-household] records the household of the human (a single agent from the agent-
set households)
; [my-mother] records the mother of the human (a single agent)
; [my-father] records the father of the human (a single agent)
; [my-spouse] records the spouse of the human (a single agent; can be no-one)

humans-own [age gender fertility recruit widowed n-children my-household my-mother
my-father my-spouse]

; attributes for the agent-set 'households':
; [household-members] records the agents who form part of the households (a number
of agents from the agent-set humans)

households-own [household-members]
```

```
to setup

; setup creates a base set of 200 humans, with a 50% chance of them being either
male or female
; first, the ages of the humans are determined in the procedure 'to age-
determination', and are taken from the life table chosen in the graphical interface
; (see 'to-report mortality' for details on the life tables)

; then, all females over 18 years of age will be coupled to a spouse of the right
age bracket (when available) and they will form a household
; humans who are not married will be distributed at random over the households;
this is not a realistic assumption, but is only done for quick model initialisation
; for the same reason, there are in this stage no widows and no recruits, and [n-
children] equals 0

  ca

  create-humans 200
  [

    ; determination of the age of each human is done in the module age-
determination

    age-determination

    ; determination of the gender of each human, with a 50% chance of them being
either male of female

    ifelse random-float 1 < 0.5
     [ set gender "M" ]
     [ set gender "F" ]

    ; the value of the variables [widowed], [recruit], [n-children] and [my-
household] are set to 0

    set widowed 0
    set recruit 0
    set n-children 0
    set my-household 0

  ]

  ask humans with [gender = "F" and age > 17]

  ; all females over 18 are coupled to a spouse

  [
    ; a male is eligible as a husband when he is between 7 and 15 years older than
the female

    let f-age age
    let husband one-of humans with [gender = "M" and my-spouse = 0 and age - f-age
> 6 and age - f-age < 16]

    ; if a husband is found, he is coupled to the female, and vice versa

    if husband != nobody
```

```
    [
      set my-spouse husband

      ask husband [
        set my-spouse myself
      ]

      ; the couple (a temporary agent-set) then will 'hatch' a new household, which
only consists of the couple itself

      let couple (turtle-set self husband)

      hatch-households 1
      [
        set household-members couple
        ask couple
        [
          set my-household myself
        ]
      ]
    ]
  ]

  ask humans with [my-household = 0]

  ; those humans who could not be married, are now added to a random household

  [
    ask one-of households [
      set household-members (turtle-set household-members myself)
      ask myself [
        set my-household myself]
    ]
  ]

  ; the global variables are now all initialized to 0

  set n-deaths 0
  set f-deaths 0
  set sum-age-at-death 0
  set sum-n-children-at-death 0
  set n-children-per-female 0

  reset-ticks

end

to go

; the model is run in four consecutive steps, executing the procedures 'to dying',
'to reproducing', 'to recruiting' and 'to marrying'
; each tick represents one year
; the order of execution implies that the steps are taken consecutively for the
whole agentset of humans, so not for one human at a time
; 1 - it is determined how many new humans will be hatched this year
; 2 - it is determined how many humans will die this year
; 3 - it is determined how many males in age 18-25 will be recruited for military
```

```
service this year (making them unavailable as spouses)
; 4 - it is determined how many females (unmarried or widowed) will marry this year

  reproducing

  dying

  recruiting

  marrying

  tick

  if f-deaths > 0 [
   set n-children-per-female sum-n-children-at-death / f-deaths
  ]

  ; the model will stop after 200 ticks/years

  if ticks = 201 [

    stop

  ]

  set n-deaths 0
  set f-deaths 0
  set sum-age-at-death 0
  set sum-n-children-at-death 0

end

to age-determination

; determine the age of the population
; for each human, an age is attributed according to the following rules:
; the probability of having an age in a 5-year cohort is determined on the basis
; of the life table selected at set up (see 'to-report mortality' for more details)
; the age within the 5-year cohort is then determined at random, so a human
; in the age cohort 25-29 years will have an equal (20%) chance of being either 25,
26, 27, 28 or 29 years old

   ask humans
   [
     let a-number random-float 1

     if Life_table = "West 3 Female"[

      if a-number < 0.1472
      [ set age 0 ]
      if a-number >= 0.1472 and a-number < 0.2900
      [ set age random 4 + 1 ]
      if a-number >= 0.2900 and a-number < 0.4190
      [ set age random 5 + 5 ]
      if a-number >= 0.4190 and a-number < 0.5319
      [ set age random 5 + 10 ]
      if a-number >= 0.5319 and a-number < 0.6294
```

```
 [ set age random 5 + 15 ]
 if a-number >= 0.6294 and a-number < 0.7124
 [ set age random 5 + 20 ]
 if a-number >= 0.7124 and a-number < 0.7821
 [ set age random 5 + 25 ]
 if a-number >= 0.7821 and a-number < 0.8396
 [ set age random 5 + 30 ]
 if a-number >= 0.8396 and a-number < 0.8860
 [ set age random 5 + 35 ]
 if a-number >= 0.8860 and a-number < 0.9226
 [ set age random 5 + 40 ]
 if a-number >= 0.9226 and a-number < 0.9505
 [ set age random 5 + 45 ]
 if a-number >= 0.9505 and a-number < 0.9707
 [ set age random 5 + 50 ]
 if a-number >= 0.9707 and a-number < 0.9843
 [ set age random 5 + 55 ]
 if a-number >= 0.9843 and a-number < 0.9926
 [ set age random 5 + 60 ]
 if a-number >= 0.9926 and a-number < 0.9971
 [ set age random 5 + 65 ]
 if a-number >= 0.9971 and a-number < 0.9991
 [ set age random 5 + 70 ]
 if a-number >= 0.9991 and a-number < 0.9998
 [ set age random 5 + 75 ]
 if a-number >= 0.9998 and a-number < 0.99998
 [ set age random 5 + 80 ]
 if a-number >= 0.99998
 [ set age random 10 + 85 ]

]

if Life_table = "Pre-industrial Standard"[

 if a-number < 0.1346
 [ set age 0 ]
 if a-number >= 0.1346 and a-number < 0.2661
 [ set age random 4 + 1 ]
 if a-number >= 0.2661 and a-number < 0.3867
 [ set age random 5 + 5 ]
 if a-number >= 0.3867 and a-number < 0.4945
 [ set age random 5 + 10 ]
 if a-number >= 0.3867 and a-number < 0.4945
 [ set age random 5 + 15 ]
 if a-number >= 0.4945 and a-number < 0.5899
 [ set age random 5 + 20 ]
 if a-number >= 0.5899 and a-number < 0.6732
 [ set age random 5 + 25 ]
 if a-number >= 0.6732 and a-number < 0.7452
 [ set age random 5 + 30 ]
 if a-number >= 0.7452 and a-number < 0.8063
 [ set age random 5 + 35 ]
 if a-number >= 0.8063 and a-number < 0.8573
 [ set age random 5 + 40 ]
 if a-number >= 0.8573 and a-number < 0.8988
 [ set age random 5 + 45 ]
 if a-number >= 0.8988 and a-number < 0.9316
```

```
 [ set age random 5 + 50 ]
 if a-number >= 0.9316 and a-number < 0.9565
 [ set age random 5 + 55 ]
 if a-number >= 0.9565 and a-number < 0.9744
 [ set age random 5 + 60 ]
 if a-number >= 0.9744 and a-number < 0.9864
 [ set age random 5 + 65 ]
 if a-number >= 0.9864 and a-number < 0.9936
 [ set age random 5 + 70 ]
 if a-number >= 0.9936 and a-number < 0.9991
 [ set age random 5 + 75 ]
 if a-number >= 0.9991 and a-number < 0.9997
 [ set age random 5 + 80 ]
 if a-number >= 0.9997
 [ set age random 10 + 85 ]

]

if Life_table = "Woods 2007 South 25"[

 if a-number < 0.1547
 [ set age 0 ]
 if a-number >= 0.1547 and a-number < 0.3046
 [ set age random 4 + 1 ]
 if a-number >= 0.3046 and a-number < 0.4389
 [ set age random 5 + 5 ]
 if a-number >= 0.4389 and a-number < 0.5547
 [ set age random 5 + 10 ]
 if a-number >= 0.5547 and a-number < 0.6528
 [ set age random 5 + 15 ]
 if a-number >= 0.6528 and a-number < 0.7345
 [ set age random 5 + 20 ]
 if a-number >= 0.7345 and a-number < 0.8015
 [ set age random 5 + 25 ]
 if a-number >= 0.8015 and a-number < 0.8557
 [ set age random 5 + 30 ]
 if a-number >= 0.8557 and a-number < 0.8987
 [ set age random 5 + 35 ]
 if a-number >= 0.8987 and a-number < 0.9320
 [ set age random 5 + 40 ]
 if a-number >= 0.9320 and a-number < 0.9571
 [ set age random 5 + 45 ]
 if a-number >= 0.9571 and a-number < 0.9750
 [ set age random 5 + 50 ]
 if a-number >= 0.9750 and a-number < 0.9870
 [ set age random 5 + 55 ]
 if a-number >= 0.9870 and a-number < 0.9943
 [ set age random 5 + 60 ]
 if a-number >= 0.9943 and a-number < 0.9979
 [ set age random 5 + 65 ]
 if a-number >= 0.9979 and a-number < 0.9994
 [ set age random 5 + 70 ]
 if a-number >= 0.9994 and a-number < 0.9999
 [ set age random 5 + 75 ]
 if a-number >= 0.9999 and a-number < 0.999996
 [ set age random 5 + 80 ]
 if a-number >= 0.999996
```

```
      [ set age random 10 + 85 ]

     ]

   ]

end

to reproducing

  ; procedure to determine if any females reproduce
  ; this depends on marriage and age; fertility ratios are determined in procedure
'to report fertility-rate'

  ; first, set the number of newborns for this year to 0

  set n-born 0

  fertility-rate ; determine the fertility rate of the female for this year

  ; then determine for each married female whether she will give birth

  ask humans with [gender = "F" and my-spouse != 0]
  [

    ; the fertility rate is a floating-point number between 0.0 and 1.0 determined
in 'to-report fertility-rate', and is based on age and the fertility estimates from
Coale and Trussell (1978)

    if random-float 1 < fertility

    ; for each married female, a random number will determine whether she will
become a mother

    [

      let mother self
      let father my-spouse

      hatch-humans 1 ; the possibility of having twins is not incorporated in this
stage, as it is not clear how this relates to the fertility estimates used; see
notes in info-section for details

      [

        ; hatched humans automatically inherit the attributes of their parents, so
these should be adapted

        set age 0
        set my-spouse 0
        set fertility 0
        set n-children 0
        set my-mother mother
        set my-father father
        if random-float 1 < 0.5 ; the child's gender needs to be determined; since
the child is produced by a female human, it will automatically be hatched with
gender "F"
```

```
        [
          set gender "M"
        ]

        ; add the newborn to the household of its parents; the child will
automatically be hatched with my-household of the mother

        ask my-household [
         set household-members (turtle-set household-members myself)
        ]
      ]

      ; update the count of newborns for this year

      set n-born n-born + 1

      ; update the count of children of the mother

      set n-children n-children + 1

      ; update the count of children of the father (this feature is not used in the
current version of the model)

      ask humans with [my-spouse = myself]
      [
        set n-children n-children + 1
      ]
    ]
  ]

end

to dying

; procedure to determine which humans will die this year
; the risk of dying is determined on the basis of the model life table selected at
setup
; statistics will be collected to determine the number of children left behind per
adult female

  ask humans
  [

    ; the risk of dying for each human is a floating-point number between 0.0 and
1.0 determined in 'to-report mortality', and is based on age and the life table
chosen at setup

    let risk-of-dying mortality

    ; for each human, a random number will determine whether they will have died

    if random-float 1 < risk-of-dying
    [

      set n-deaths n-deaths + 1 ; increase the number of humans who died by 1
      set sum-age-at-death sum-age-at-death + age ; get the sum of ages of humans
who died
```

```
      if gender = "F" and age > 17 [
        set f-deaths f-deaths + 1 ; increase the number of adult females who died
by 1
        set sum-n-children-at-death sum-n-children-at-death + n-children ; get the
sum of the number of offspring of adult females who died
      ]

      ; the spouse, if applicable, will become widowed

      ask humans with [my-spouse = myself]
      [
        set my-spouse 0 ; it should be set to 0, otherwise my-spouse will be set to
nobody, i.e. the turtle that is about to die, creating problems down the line when
selecting married/unmarried turtles
        set widowed 1
      ]

      die
    ]

    ; for those humans who did not die, increase age by 1 year/tick

    set age age + 1

  ]

  ; it may be that the person who died was the last one of a household; in this
case, the household will be deleted

  ask households with [count household-members = 0]
  [
    die
  ]

end


to recruiting

  ; this procedures determines whether unmarried males between 18 and 25 years old
will be recruited for army service
  ; this age is thought to be a realistic reflection of actual recruitment
practices of the Roman army
  ; the recruitment rate is set using the slide at setup, and can vary between 0.0
and 0.2 (with steps of 0.01)
  ; recruited males are not available as spouses until they have finished their
service term
  ; this may not be a completely realistic assumption, but it is used here to
understand the
  ; consequences of removing a certain proportion of males from the reproduction
pool

  ; recruitment will start after stabilization of the model at ticks = 100

  if ticks > 100 [
```

```
   ask humans with [gender = "M" and age > 17 and age < 26 and my-spouse = 0]

   ; for each unmarried male between 18 and 25 years old, a random number will
determine whether he will be recruited

   [
     if random-float 1 < recruitment
     [
       set recruit 1
     ]
   ]

   ; for every year served, the value of [recruit] will be increase by 1
   ; after serving a 25-years term in the army, the male will be added to the
reproduction pool, and will be available for marriage again

   ask humans with [recruit > 0]
   [
     set recruit recruit + 1
     if recruit > 25
     [
       set recruit 0
     ]
   ]

  ]

end

to marrying

  ; this procedure will try to get unmarried females over 18 married; they will
start a new household if necessary
  ; in this model, they will always be married when a suitable unmarried male is
present, but many more options could be explored here; see info-section for more
details

  ask humans with [gender = "F" and age > 17 and my-spouse = 0]

  [

    ; any unmarried male over 25 is a potential partner; this includes widowers and
soldiers returning from their army service

    let husband one-of humans with [gender = "M" and age > 25 and my-spouse = 0 and
recruit = 0]

    ; when a suitable husband is found, determine if a new household should be
started

    if husband != nobody [
      set my-spouse husband
      set widowed 0

      let couple (turtle-set self husband)

      ; if the male is widowed, then the female will be added to his household
```

```
      ; else the couple will start a new household
      ; in this model, this feature is not used for any particular purpose, but it
serves to keep the number of agents as low as possible

      ask husband [
        set my-spouse myself
        ifelse widowed = 0
         [
           hatch-households 1
           [
            set household-members couple
            ask couple
            [
             set my-household myself
            ]
           ]
         ]
         [
          ask my-household [
            set household-members (turtle-set household-members self)
          ]
          set widowed 0
         ]
       ]
     ]
   ]

end

to-report mortality

  ; in this procedure, the mortality rate (risk of dying) of each human is
determined; it is based on one the three life tables from which the user can choose
at setup; these are:

  ; Coale and Demeny's (1966) Model West Level 3 Female
  ; Wood's (2007) South High Mortality with e0=25, and
  ; and Séguy and Buchet's (2013) Pre-Industrial Standard table
  ; N.B. the first two are adapted versions taken from Hin (2013)!

  ; the life tables used here represent mortality rates per 5-year cohort, so
mortality will only change when the human has lived for another 5 years (passes
into the next cohort)
  ; this could be a little bit more sophisticated (see e.g. Danielisová et al.
2015)

  let mortality-5year 0

  if Life_table = "West 3 Female" [
   if age = 0 [set mortality-5year 0.3056]
   if age > 0 and age <= 4 [set mortality-5year 0.2158 / 4]
   if age > 4 and age <= 9 [set mortality-5year 0.0606 / 5]
   if age > 9 and age <= 14 [set mortality-5year 0.0474 / 5]
   if age > 14 and age <= 19 [set mortality-5year 0.0615 / 5]
   if age > 19 and age <= 24 [set mortality-5year 0.0766 / 5]
   if age > 24 and age <= 29 [set mortality-5year 0.0857 / 5]
   if age > 29 and age <= 34 [set mortality-5year 0.0965 / 5]
```

```
 if age > 34 and age <= 39 [set mortality-5year 0.1054 / 5]
 if age > 39 and age <= 44 [set mortality-5year 0.1123 / 5]
 if age > 44 and age <= 49 [set mortality-5year 0.1197 / 5]
 if age > 49 and age <= 54 [set mortality-5year 0.1529 / 5]
 if age > 54 and age <= 59 [set mortality-5year 0.1912 / 5]
 if age > 59 and age <= 64 [set mortality-5year 0.2715 / 5]
 if age > 64 and age <= 69 [set mortality-5year 0.3484 / 5]
 if age > 69 and age <= 74 [set mortality-5year 0.4713 / 5]
 if age > 74 and age <= 79 [set mortality-5year 0.6081 / 5]
 if age > 79 and age <= 84 [set mortality-5year 0.7349 / 5]
 if age > 84 and age <= 89 [set mortality-5year 0.8650 / 5]
 if age > 89 and age <= 94 [set mortality-5year 0.9513 / 5]
 if age > 94 [set mortality-5year 1.000 / 5]
]
if Life_table = "Pre-industrial Standard"[
 if age = 0 [set mortality-5year 0.200]
 if age > 0 and age <= 4 [set mortality-5year 0.150 / 4]
 if age > 4 and age <= 9 [set mortality-5year 0.052 / 5]
 if age > 9 and age <= 14 [set mortality-5year 0.029 / 5]
 if age > 14 and age <= 19 [set mortality-5year 0.038 / 5]
 if age > 19 and age <= 24 [set mortality-5year 0.049 / 5]
 if age > 24 and age <= 29 [set mortality-5year 0.054 / 5]
 if age > 29 and age <= 34 [set mortality-5year 0.060 / 5]
 if age > 34 and age <= 39 [set mortality-5year 0.068 / 5]
 if age > 39 and age <= 44 [set mortality-5year 0.079 / 5]
 if age > 44 and age <= 49 [set mortality-5year 0.093 / 5]
 if age > 49 and age <= 54 [set mortality-5year 0.115 / 5]
 if age > 54 and age <= 59 [set mortality-5year 0.152 / 5]
 if age > 59 and age <= 64 [set mortality-5year 0.202 / 5]
 if age > 64 and age <= 69 [set mortality-5year 0.275 / 5]
 if age > 69 and age <= 74 [set mortality-5year 0.381 / 5]
 if age > 74 and age <= 79 [set mortality-5year 0.492 / 5]
 if age > 79 and age <= 84 [set mortality-5year 0.657 / 5]
 if age > 84 [set mortality-5year 1.00 / 3.55]
]
if Life_table = "Woods 2007 South 25"[
 if age = 0 [set mortality-5year 0.2900]
 if age > 0 and age <= 4 [set mortality-5year 0.1900 / 4]
 if age > 4 and age <= 9 [set mortality-5year 0.0546 / 5]
 if age > 9 and age <= 14 [set mortality-5year 0.0429 / 5]
 if age > 14 and age <= 19 [set mortality-5year 0.0707 / 5]
 if age > 19 and age <= 24 [set mortality-5year 0.1065 / 5]
 if age > 24 and age <= 29 [set mortality-5year 0.1234 / 5]
 if age > 29 and age <= 34 [set mortality-5year 0.1301 / 5]
 if age > 34 and age <= 39 [set mortality-5year 0.1366 / 5]
 if age > 39 and age <= 44 [set mortality-5year 0.1392 / 5]
 if age > 44 and age <= 49 [set mortality-5year 0.1490 / 5]
 if age > 49 and age <= 54 [set mortality-5year 0.1655 / 5]
 if age > 54 and age <= 59 [set mortality-5year 0.1857 / 5]
 if age > 59 and age <= 64 [set mortality-5year 0.2613 / 5]
 if age > 64 and age <= 69 [set mortality-5year 0.3853 / 5]
 if age > 69 and age <= 74 [set mortality-5year 0.5288 / 5]
 if age > 74 and age <= 79 [set mortality-5year 0.6403 / 5]
 if age > 79 and age <= 84 [set mortality-5year 0.7431 / 5]
 if age > 84 [set mortality-5year 1.00 / 3.55]
]
```

```
  report mortality-5year

end

to fertility-rate

  ; in this procedure, the fertility rate (probability of reproducing) of each
female is determined, based on the figures given in Coale & Trussell (1978)
  ; the figures used here represent fertility rates per 5-year cohort, so fertility
will only change when the female has lived for another 5 years (passes into the
next cohort)

  ; a more realistic approach would take into account the time that has passed
since the previous birth
  ; however, this would make the model much slower, since we would then have to use
time steps of one month

  ; not that fertility will be determined once a female has reached age 15;
however, in this model reproduction is not allowed until the female is married

  ask humans with [gender = "F"]
  [
     if age < 15
     [
       set fertility 0.000
     ]
     if age > 14 and age <= 19
     [
       set fertility 0.411
     ]
     if age > 19 and age <= 24
     [
       set fertility 0.46
     ]
     if age > 24 and age <= 29
     [
       set fertility 0.431
     ]
     if age > 29 and age <= 34
     [
       set fertility 0.395
     ]
     if age > 34 and age <= 39
     [
       set fertility 0.322
     ]
     if age > 39 and age <= 44
     [
       set fertility 0.167
     ]
     if age > 45 and age <= 49
     [
       set fertility 0.024
     ]
     if age > 49
     [
       set fertility 0.000
```

```
      ]
    ]

end
```