```
globals [
  ;; Escapees in terms of seat color
  cyan-escapees
  yellow-escapees
  red-escapees
  beige-escapees

  ;; Fire victims in terms of seat color
  cyan-fire-victims
  yellow-fire-victims
  red-fire-victims
  beige-fire-victims

  ;; Stampede victims in terms of seat color
  cyan-stampede-victims
  yellow-stampede-victims
  red-stampede-victims
  beige-stampede-victims

  ;; Escapees in terms of gender
  female-escapees
  male-escapees

  ;; Fire victims in terms of gender
  fire-victims-female
  fire-victims-male

  ;; Stampede victims in terms of gender
  stampede-victims-female
  stampede-victims-male


  ;; Escapees in terms of age
  child-escapees
  adult-escapees
  elderly-escapees

  ;; Fire victims in terms of age
  fire-victims-child
  fire-victims-adult
  fire-victims-elderly

  ;; Stampede victims in terms of
  stampede-victims-child
  stampede-victims-adult
  stampede-victims-elderly

  oldgoal
]

breed [survivors survivor]
breed[doors door]

;; Distance to all 8 exits from the agent position and distance to fire
patches-own [
  distance1
  distance2
  distance3
  distance4
```

```
    distance5
    distance6
    distance7
    distance8
    distancefire
]

survivors-own [
    visual-range
    gender
    age
    mass
    goal
    base-speed
    speed
    health ; agent own health is calculated using mass * speed * threshold
    knowledge   ; to be used later, agent knowçeged of the world
]

to setup
    ca
    ;; setting up the theatre and agents
    setup-theatre

    ;; Since, we are designing exits in only one side, y is same. x varies.
    let stairs-x-coor [-89 -68 -50 -30 24 45 64 84]

    ;; Setting exit path
    (foreach stairs-x-coor[ [x] ->
        create-doors 1 [setxy x 21 set shape "square" set color lime set heading 180
set size 2]
      ])

    ;; compute initial distance of each agent to all exit stairs
    ask patches [set distance1 [distance myself] of door 9537]
    ask patches [set distance2 [distance myself] of door 9538]
    ask patches [set distance3 [distance myself] of door 9539]
    ask patches [set distance4 [distance myself] of door 9540]
    ask patches [set distance5 [distance myself] of door 9541]
    ask patches [set distance6 [distance myself] of door 9542]
    ask patches [set distance7 [distance myself] of door 9543]
    ask patches [set distance8 [distance myself] of door 9544]

    ;; Nearest exit is considered as goal
    ask survivors[
      set goal min-one-of doors [distance myself]
    ]

    set-survivors-attributes

    let origin patch 0 135

    ; set n fires at random location inside the theatre
    let n_fires range(number-fires)
    foreach n_fires[ [x] ->
      set origin patch random-xcor random-ycor
      while [ [ pcolor ] of origin = black ] [
        set origin one-of patches
      ]
```

```
    ask origin [
      draw-rectangle pxcor pycor 5 5 orange ; plot the fire
    ]
  ]

  ask patches [set distancefire distancexy 0 135]

  reset-ticks
end

to go
  spread-fire
  ifelse behaviour = "smart"
  [ smart-strategy ] ; Have knowledge about the exits
  [ follow-strategy ] ; Follow the person in front of them


  ask survivors [
    ;; If force > health, agent is considered as dead
    ;; Updating victim counts by their respective seat color
    if calculate-force patch-here >= health [
      ifelse color = cyan
        [ set cyan-stampede-victims cyan-stampede-victims + 1 ]
        [ ifelse color = yellow
          [ set yellow-stampede-victims yellow-stampede-victims + 1 ]
          [ ifelse color = red
            [ set red-stampede-victims red-stampede-victims + 1 ]
            [ if color = 29 ;; beige
              [ set beige-stampede-victims beige-stampede-victims + 1 ]
            ]

          ]
        ]

      ;; update victim counts based on their gender
      ifelse gender = "female"
      [ set stampede-victims-female stampede-victims-female + 1 ]
      [ set stampede-victims-male stampede-victims-male + 1 ]

      ;; update victim counts based on their age
      ifelse age = "child"
      [ set stampede-victims-child stampede-victims-child + 1 ]
      [ ifelse age = "adult"
        [ set stampede-victims-adult stampede-victims-adult + 1 ]
        [ set stampede-victims-elderly stampede-victims-elderly + 1 ]
      ]

      ;; agent is dead
      die
    ]
  ]

  tick
end


to-report calculate-force [ p ]
  let force 0
  ;; Force exerted on the whole patch
```

```
    ask survivors-on p [
      set force force + mass * speed
    ]
    report force
end

to spread-fire
  ;; Fire spreads
  if ticks mod (20 - fire-speed) = 0 [
    ask patches with [ pcolor = orange ] [
      ask neighbors with [ pcolor != black ] [
        set pcolor orange
      ]
    ]
  ]

  ask survivors [
    ;; update fire victims based on their seat color
    if [ pcolor ] of patch-here = orange [

      ifelse color = cyan
      [ set cyan-fire-victims cyan-fire-victims + 1 ]
      [ ifelse color = yellow
        [ set yellow-fire-victims yellow-fire-victims + 1 ]
        [ ifelse color = red
          [ set red-fire-victims red-fire-victims + 1 ]
          [ if color = 29 ;; beige
            [ set beige-fire-victims beige-fire-victims + 1 ]
          ]
        ]
      ]


      ;; update fire victims based on their gender
      ifelse gender = "female"
      [ set fire-victims-female fire-victims-female + 1 ]
      [ set fire-victims-male fire-victims-male + 1 ]

      ;; update fire victims based on their age
      ifelse age = "child"
      [ set fire-victims-child fire-victims-child + 1 ]
      [ ifelse age = "adult"
        [ set fire-victims-adult fire-victims-adult + 1 ]
        [ set fire-victims-elderly fire-victims-elderly + 1 ]
      ]

      ; agent is dead
      die
    ]
  ]

  ;; Change the goal if that particular stairs is on fire
  set oldgoal FALSE

  ask doors [
    if [ pcolor ] of patch-here = orange [set oldgoal true
      die ]
  ]
```

```
  ;; Setting the nearest exit
  ask survivors [ set goal min-one-of doors [distance myself]
  ]
end

;; Agents following "Smart" Strategy
to smart-strategy
  ask survivors [
    let next-patch 0
    ifelse goal = door 9537 [set next-patch min-one-of neighbors [distance1]] [
      ifelse goal = door 9538 [set next-patch min-one-of neighbors [distance2]] [
        ifelse goal = door 9539 [set next-patch min-one-of neighbors [distance3]] [
          ifelse goal = door 9540 [set next-patch min-one-of neighbors [distance4]]
[
            ifelse goal = door 9541 [set next-patch min-one-of neighbors
[distance5]] [
              ifelse goal = door 9542 [set next-patch min-one-of neighbors
[distance6]] [
                ifelse goal = door 9543 [set next-patch min-one-of neighbors
[distance7]] [
                  ifelse goal = door 9544 [set next-patch min-one-of neighbors
[distance8]] []]]]]]]]
    repeat speed [
      ;; Setting initial distance to high number
      while [ [pcolor] of next-patch != grey] [
        ask next-patch [
          set distance1 10000000
          set distance2 10000000
          set distance3 10000000
          set distance4 10000000
          set distance5 10000000
          set distance6 10000000
          set distance7 10000000
          set distance8 10000000
        ]
        ;; Setting distance
        ifelse goal = door 9537 [set next-patch min-one-of neighbors [distance1]] [
          ifelse goal = door 9538 [set next-patch min-one-of neighbors [distance2]]
[
            ifelse goal = door 9539 [set next-patch min-one-of neighbors
[distance3]] [
              ifelse goal = door 9540 [set next-patch min-one-of neighbors
[distance4]] [
                ifelse goal = door 9541 [set next-patch min-one-of neighbors
[distance5]] [
                  ifelse goal = door 9542 [set next-patch min-one-of neighbors
[distance6]] [
                    ifelse goal = door 9543 [set next-patch min-one-of neighbors
[distance7]] [
                      ifelse goal = door 9544 [set next-patch min-one-of neighbors
[distance8]] []]]]]]]]
      ]
      ;; Agents can only move to a patch when it is not crowded
      if not patch-overcrowded? next-patch [ move-to next-patch ]
    ]

    ;; Updating escapees based on their seat color
    if any? doors-here [
      ifelse color = cyan
```

```
          [ set cyan-escapees cyan-escapees + 1 ]
          [ ifelse color = yellow
            [ set yellow-escapees yellow-escapees + 1 ]
            [ ifelse color = red
              [ set red-escapees red-escapees + 1 ]
              [ ifelse color = 29
                [ set beige-escapees beige-escapees + 1 ]
                []
              ]
            ]
          ]

      ;; Updating escapees based on their gender
      ifelse gender = "female"
      [ set female-escapees female-escapees + 1 ]
      [ set male-escapees male-escapees + 1 ]

      ;; Updating escapees based on their age
      ifelse age = "child"
      [ set child-escapees child-escapees + 1 ]
      [ ifelse age = "adult"
        [ set adult-escapees adult-escapees + 1 ]
        [ set elderly-escapees elderly-escapees + 1 ]
      ]

      ;; Agent is dead
      die
    ]
  ]
end

;; Agents following "Follow" strategy
to follow-strategy
  ask survivors [
    let next-patch 0
    ;; If agent can see exit in their visual range, they use that exit
    ifelse any? patches in-radius visual-range with [any? doors-here] [
        set goal min-one-of doors [distance myself]
        ifelse goal = door 9537 [set next-patch min-one-of neighbors [distance1]]
[
            ifelse goal = door 9538 [set next-patch min-one-of neighbors
[distance2]] [
              ifelse goal = door 9539 [set next-patch min-one-of neighbors
[distance3]] [
                ifelse goal = door 9540 [set next-patch min-one-of neighbors
[distance4]] [
                  ifelse goal = door 9541 [set next-patch min-one-of neighbors
[distance5]] [
                    ifelse goal = door 9542 [set next-patch min-one-of neighbors
[distance6]] [
                      ifelse goal = door 9543 [set next-patch min-one-of neighbors
[distance7]] [
                        ifelse goal = door 9544 [set next-patch min-one-of
neighbors [distance8]] []]]]]]]]

          while [ [pcolor] of next-patch != grey] [
            ask next-patch [
              set distance1 10000000
```

```
                 set distance2 10000000
                 set distance3 10000000
                 set distance4 10000000
                 set distance5 10000000
                 set distance6 10000000
                 set distance7 10000000
                 set distance8 10000000
            ]
            ifelse goal = door 9537 [set next-patch min-one-of neighbors
[distance1]] [
              ifelse goal = door 9538 [set next-patch min-one-of neighbors
[distance2]] [
                ifelse goal = door 9539 [set next-patch min-one-of neighbors
[distance3]] [
                  ifelse goal = door 9540 [set next-patch min-one-of neighbors
[distance4]] [
                    ifelse goal = door 9541 [set next-patch min-one-of neighbors
[distance5]] [
                      ifelse goal = door 9542 [set next-patch min-one-of neighbors
[distance6]] [
                        ifelse goal = door 9543 [set next-patch min-one-of
neighbors [distance7]] [
                          ifelse goal = door 9544 [set next-patch min-one-of
neighbors [distance8]] []]]]]]]]


          ]
      ][

      ifelse is-patch? patch-at-heading-and-distance (180 + heading) visual-range
and is-patch? patch-at-heading-and-distance (90 + heading) visual-range and is-
patch? patch-at-heading-and-distance (270 + heading) visual-range [
        ifelse [pcolor] of patch-ahead visual-range = orange or [pcolor] of patch-
at-heading-and-distance (180 + heading) visual-range = orange or [pcolor] of patch-
at-heading-and-distance (90 + heading) visual-range = orange or [pcolor] of patch-
at-heading-and-distance (270 + heading) visual-range = orange [
          set next-patch max-one-of neighbors [distancefire]
          while [ [pcolor] of next-patch != grey] [
            ask next-patch [set distancefire 0]
            set next-patch max-one-of neighbors [distancefire]
          ]

          ;; Agents can only move to a patch when it is not crowded
          if not patch-overcrowded? next-patch [ move-to next-patch ]
        ][
          ;; Agents follow the agent in front of them
          if any? turtles-on neighbors [
            ;; Check where majority agents are moving within their visual range
            let avg mean [heading] of turtles-on neighbors
            ifelse avg <= 90 [set heading 90][
              ifelse avg <= 180 [set heading 180][
                ifelse avg <= 270 [set heading 270][
                  set heading 0
                ]
              ]
            ]

            if [pcolor] of patch-ahead 1 = gray [fd 1]
          ]
        ]
```

```
      ][]
    ]
]
end

to setup-theatre
  ;; Creating seats of different colors
  draw-rectangle -110 135 215 105 gray

  create-cyan1
  create-cyan2
  create-cyan3
  create-yellow1
  create-yellow2
  create-yellow3
  create-yellow4
  create-yellow5
  create-lime1
  create-lime2
  create-lime3

  draw-rectangle -250 25 500 8 gray

  ;; Creating stairs facing left and right directions
  create-stairs1
  create-stairs2
end

to set-survivors-attributes
  ask survivors [
    ;; Setting agent's gender
    ifelse random-float 1.0 < perc-male-female
    [ set gender "male" ]
    [ set gender "female" ]

    ;; Setting agent's age
    ifelse random-float 1.0 <= perc-child
    [ set age "child" ]
    [ ifelse random-float 1.0 < perc-adults
      [ set age "adult" ]
      [ set age "elderly" ]
    ]

    ifelse age = "child"
    [ set base-speed 0.3889 ]
    [ ifelse age = "adult"
      ;; Setting the speed after converting min and max speed's km/h to m/s
      [ set base-speed 1.4778 + random-float  1.5083 ]
      [ set base-speed 1.2528 + random-float 1.3194 ]
    ]
    set speed base-speed
        ; set base speed

    ;; Setting agent's mass in normal distribution ran
    ifelse age = "child"
    [ ifelse gender = "male"
      [ set mass random-normal 40 4]
      [ set mass random-normal 35 4]
```

```
    ]
    [ ifelse gender = "male"
      [ set mass random-normal mean-mass-male 4]
      [ set mass random-normal mean-mass-female 4]
    ]

    ;; Setting agent's health
    set health mass * speed * threshold

    ;; Setting agent's visual range
    set visual-range random max-vision
  ]
end


to-report patch-overcrowded? [ p ]
  report count turtles-on p >  num-survivors-per-patch
end

;; Create left facing stairs
to create-stairs1
  let xlist left-stairs-xlist 82
  let ylist stairs-ylist 5 30
  (foreach xlist ylist [ [x y] ->
    draw-rectangle x y 2 1 gray
    ])
  set xlist left-stairs-xlist 44
  set ylist stairs-ylist 5 30
  (foreach xlist ylist [ [x y] ->
    draw-rectangle x y 2 1 gray
    ])
  set xlist left-stairs-xlist -33
  set ylist stairs-ylist 5 30
  (foreach xlist ylist [ [x y] ->
    draw-rectangle x y 2 1 gray
    ])
  set xlist left-stairs-xlist -70
  set ylist stairs-ylist 5 30
  (foreach xlist ylist [ [x y] ->
    draw-rectangle x y 2 1 gray
    ])
end

;; Create right facing stairs
to create-stairs2
  let xlist right-stairs-xlist 64
  let ylist stairs-ylist 5 30
  (foreach xlist ylist [ [x y] ->
    draw-rectangle x y 2 1 gray
    ])
  set xlist right-stairs-xlist 26
  set ylist stairs-ylist 5 30
  (foreach xlist ylist [ [x y] ->
    draw-rectangle x y 2 1 gray
    ])
  set xlist right-stairs-xlist -50
  set ylist stairs-ylist 5 30
  (foreach xlist ylist [ [x y] ->
    draw-rectangle x y 2 1 gray
```

```
    ])
  set xlist right-stairs-xlist -88
  set ylist stairs-ylist 5 30
  (foreach xlist ylist [ [x y] ->
    draw-rectangle x y 2 1 gray
    ])
end

;; Creating cyan seats
to create-cyan1
  let ylist seat-ylist 17 135
  let agent-list create-agentlist 17 -106
  (foreach ylist [ [y] ->
    draw-rectangle -106 y 17 1 cyan
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color cyan
set heading 180]])
    ])
  set ylist seat-ylist 17 100
  (foreach ylist [ [y] ->
    draw-rectangle -106 y 17 1 cyan
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color cyan
set heading 180]])
    ])
  set ylist seat-ylist 17 65
  (foreach ylist [ [y] ->
    draw-rectangle -106 y 17 1 cyan
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color cyan
set heading 180]])
    ])
end

;; Creating cyan seats
to create-cyan2
  let ylist seat-ylist 17 135
  let agent-list create-agentlist 17 -87
  (foreach ylist [ [y] ->
    draw-rectangle -87 y 17 1 cyan
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color cyan
set heading 180]])
    ])
  set ylist seat-ylist 17 100
  (foreach ylist [ [y] ->
    draw-rectangle -87 y 17 1 cyan
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color cyan
set heading 180]])
    ])
  set ylist seat-ylist 17 65
  (foreach ylist [ [y] ->
    draw-rectangle -87 y 17 1 cyan
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color cyan
set heading 180]])
    ])
end

;; Creating cyan seats
to create-cyan3
  let ylist seat-ylist 17 135
  let agent-list create-agentlist 17 -68
  (foreach ylist [ [y] ->
```

```
    draw-rectangle -68 y 17 1 cyan
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color cyan
set heading 180]])
    ])
  set ylist seat-ylist 17 100
  (foreach ylist [ [y] ->
    draw-rectangle -68 y 17 1 cyan
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color cyan
set heading 180]])
    ])
  set ylist seat-ylist 17 65
  (foreach ylist [ [y] ->
    draw-rectangle -68 y 17 1 cyan
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color cyan
set heading 180]])
    ])
end

;; Creating yellow seats
to create-yellow1
  let ylist seat-ylist 17 135
  let agent-list create-agentlist 17 -49
  (foreach ylist [ [y] ->
    draw-rectangle -49 y 17 1 yellow
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color yellow
set heading 180]])
    ])
  set ylist seat-ylist 17 100
  (foreach ylist [ [y] ->
    draw-rectangle -49 y 17 1 yellow
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color yellow
set heading 180]])
    ])
  set ylist seat-ylist 17 65
  (foreach ylist [ [y] ->
    draw-rectangle -49 y 17 1 yellow
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color yellow
set heading 180]])
    ])
end

;; Creating yellow seats
to create-yellow2
  let ylist seat-ylist 17 135
  let agent-list create-agentlist 17 -30
  (foreach ylist [ [y] ->
    draw-rectangle -30 y 17 1 yellow
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color yellow
set heading 180]])
    ])
  set ylist seat-ylist 17 100
  (foreach ylist [ [y] ->
    draw-rectangle -30 y 17 1 red
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color red
set heading 180]])
    ])
  set ylist seat-ylist 17 65
  (foreach ylist [ [y] ->
    draw-rectangle -30 y 17 1 red
```

```
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color red
set heading 180]])
    ])
end

;; Creating yellow seats
to create-yellow3
  let ylist seat-ylist 17 135
  let agent-list create-agentlist 17 -11
  (foreach ylist [ [y] ->
    draw-rectangle -11 y 17 1 yellow
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color yellow
set heading 180]])
    ])
  set ylist seat-ylist 17 100
  (foreach ylist [ [y] ->
    draw-rectangle -11 y 17 1 red
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color red
set heading 180]])
    ])
  set ylist seat-ylist 17 65
  (foreach ylist [ [y] ->
    draw-rectangle -11 y 17 1 red
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color red
set heading 180]])
    ])
end

;; Creating yellow seats
to create-yellow4
  let ylist seat-ylist 17 135
  let agent-list create-agentlist 17 8
  (foreach ylist [ [y] ->
    draw-rectangle 8 y 17 1 yellow
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color yellow
set heading 180]])
    ])
  set ylist seat-ylist 17 100
  (foreach ylist [ [y] ->
    draw-rectangle 8 y 17 1 red
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color red
set heading 180]])
    ])
  set ylist seat-ylist 17 65
  (foreach ylist [ [y] ->
    draw-rectangle 8 y 17 1 red
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color red
set heading 180]])
    ])
end

;; Creating yellow seats
to create-yellow5
  let ylist seat-ylist 17 135
  let agent-list create-agentlist 17 27
  (foreach ylist [ [y] ->
    draw-rectangle 27 y 17 1 yellow
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color yellow
set heading 180]])
```

```
      ])
  set ylist seat-ylist 17 100
  (foreach ylist [ [y] ->
    draw-rectangle 27 y 17 1 yellow
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color yellow
set heading 180]])
    ])
  set ylist seat-ylist 17 65
  (foreach ylist [ [y] ->
    draw-rectangle 27 y 17 1 yellow
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color yellow
set heading 180]])
    ])
end

;; Creating lime seats
to create-lime1
  let ylist seat-ylist 17 135
  let agent-list create-agentlist 17 46
  (foreach ylist [ [y] ->
    draw-rectangle 46 y 17 1 29
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color 29 set
heading 180]])
    ])
  set ylist seat-ylist 17 100
  (foreach ylist [ [y] ->
    draw-rectangle 46 y 17 1 29
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color 29 set
heading 180]])
    ])
  set ylist seat-ylist 17 65
  (foreach ylist [ [y] ->
    draw-rectangle 46 y 17 1 29
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color 29 set
heading 180]])
    ])
end

;; Creating lime seats
to create-lime2
  let ylist seat-ylist 17 135
  let agent-list create-agentlist 17 65
  (foreach ylist [ [y] ->
    draw-rectangle 65 y 17 1 29
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color 29 set
heading 180]])
    ])
  set ylist seat-ylist 17 100
  (foreach ylist [ [y] ->
    draw-rectangle 65 y 17 1 29
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color 29 set
heading 180]])
    ])
  set ylist seat-ylist 17 65
  (foreach ylist [ [y] ->
    draw-rectangle 65 y 17 1 29
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color 29 set
heading 180]])
    ])
```

```
end

;; Creating lime seats
to create-lime3
  let ylist seat-ylist 17 135
  let agent-list create-agentlist 17 84
  (foreach ylist [ [y] ->
    draw-rectangle 84 y 17 1 29
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color 29 set
heading 180]])
    ])
  set ylist seat-ylist 17 100
  (foreach ylist [ [y] ->
    draw-rectangle 84 y 17 1 29
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color 29 set
heading 180]])
    ])
  set ylist seat-ylist 17 65
  (foreach ylist [ [y] ->
    draw-rectangle 84 y 17 1 29
    (foreach agent-list [ [ppl] -> create-survivors 1 [setxy ppl y set color 29 set
heading 180]])
    ])
end

;; X coordinates list for right facing stairs
to-report right-stairs-xlist [ip]
  report n-values (5) [[i] -> ip - i * 1]
end

;; X coordinates list for left facing stairs
to-report left-stairs-xlist [ip]
  report n-values (5) [[i] -> ip + i * 1]
end

;; Y coordinates for stairs as it is same for all the stairs.
to-report stairs-ylist [ip1 ip2]
  report n-values (ip1) [[i] -> ip2 - i * 1]
end

;; Y coordinates for seats in the theatre
to-report seat-ylist [ip1 ip2]
  report n-values (ip1) [[i] -> ip2 - i * 2]
end

to-report create-agentlist [ip1 ip2]
  report n-values (ip1) [[i] -> ip2 + i]
end

to draw-rectangle [ x y w l patch-color ]
  ask patches with
  [ w + x > pxcor and pxcor >= x
    and
    y >= pycor and pycor > (y - l) ] [ set pcolor patch-color ]
end
@#$#@#$#@
GRAPHICS-WINDOW
422
10
```

1357
542
-1
-1
3.85
1
10
1
1
1
0
0
0
1
-120
120
0
135
1
1
1
ticks
1.0

BUTTON
110
281
194
314
NIL
setup
NIL
1
T
OBSERVER
NIL
NIL
NIL
NIL
1

MONITOR
269
437
385
482
Survivors Count
count survivors
17
1
11

CHOOSER
116
14
293
59
behaviour
behaviour

"smart" "follow"
0

MONITOR
269
373
382
418
Agents Exited count
cyan-escapees + yellow-escapees + red-escapees + beige-escapees
17
1
11

SLIDER
15
239
194
272
threshold
threshold
10
100
60.0
10
1
NIL
HORIZONTAL

PLOT
13
371
244
521
Survivors
NIL
NIL
0.0
10.0
0.0
10.0
true
true
"" ""
PENS
"Survivors" 1.0 0 -16777216 true "" "plot count survivors"
"Escapees" 1.0 0 -13840069 true "" "plot cyan-escapees + yellow-escapees + red-escapees + beige-escapees"
"Stampede" 1.0 0 -2674135 true "" "plot cyan-stampede-victims + yellow-stampede-victims + red-stampede-victims + beige-stampede-victims"
"Fire-deaths" 1.0 0 -955883 true "" "plot cyan-fire-victims + yellow-fire-victims + red-fire-victims + beige-fire-victims"

MONITOR
1370
62
1497
107
Fire victims count

```
cyan-fire-victims + yellow-fire-victims + red-fire-victims + beige-fire-victims
17
1
11

MONITOR
1369
11
1498
56
Stampede victims count
cyan-stampede-victims + yellow-stampede-victims + red-stampede-victims + beige-
stampede-victims
17
1
11

BUTTON
121
321
282
354
go until no survivors
go\nif count survivors = 0 [ stop ]
T
1
T
OBSERVER
NIL
NIL
NIL
NIL
1

MONITOR
1519
37
1637
82
Accessible Exits Count
count doors
17
1
11

PLOT
1334
565
1578
720
Escapees by Color
NIL
NIL
0.0
10.0
0.0
10.0
true
true
```

```
"" ""
PENS
"cyan" 1.0 0 -11221820 true "" "plot cyan-escapees"
"yellow" 1.0 0 -1184463 true "" "plot yellow-escapees"
"red" 1.0 0 -2674135 true "" "plot red-escapees"
"beige" 1.0 0 -204336 true "" "plot beige-escapees"

PLOT
1065
564
1322
717
Stampede victims by Color
NIL
NIL
0.0
10.0
0.0
10.0
true
true
"" ""
PENS
"cyan" 1.0 0 -11221820 true "" "plot cyan-stampede-victims"
"yellow" 1.0 0 -1184463 true "" "plot yellow-stampede-victims"
"red" 1.0 0 -2674135 true "" "plot red-stampede-victims"
"beige" 1.0 0 -204336 true "" "plot beige-stampede-victims"

PLOT
815
564
1052
714
Fire victims by Color
NIL
NIL
0.0
10.0
0.0
10.0
true
true
"" ""
PENS
"cyan" 1.0 0 -11221820 true "" "plot cyan-fire-victims"
"red" 1.0 0 -2674135 true "" "plot red-fire-victims"
"yellow" 1.0 0 -1184463 true "" "plot yellow-fire-victims"
"beige" 1.0 0 -204336 true "" "plot beige-fire-victims"

SLIDER
15
195
195
228
max-vision
max-vision
20
100
60.0
```

```
5
1
NIL
HORIZONTAL

BUTTON
205
283
286
316
NIL
go
T
1
T
OBSERVER
NIL
NIL
NIL
NIL
1

PLOT
1617
208
1856
358
Escapees by Gender
NIL
NIL
0.0
10.0
0.0
10.0
true
true
"" ""
PENS
"female" 1.0 0 -2674135 true "" "plot female-escapees"
"male" 1.0 0 -13345367 true "" "plot male-escapees"

PLOT
14
565
249
715
Escapees by Age Group
NIL
NIL
0.0
10.0
0.0
10.0
true
true
"" ""
PENS
"child" 1.0 0 -2064490 true "" "plot child-escapees"
"adult" 1.0 0 -5825686 true "" "plot adult-escapees"
```

```
"senior" 1.0 0 -8630108 true "" "plot elderly-escapees"

PLOT
1371
119
1605
269
Fire Deaths by Gender
NIL
NIL
0.0
10.0
0.0
10.0
true
true
"" ""
PENS
"female" 1.0 0 -2674135 true "" "plot fire-victims-female"
"male" 1.0 0 -13345367 true "" "plot fire-victims-male"

PLOT
1373
289
1608
439
Stampede Deaths by Gender
NIL
NIL
0.0
10.0
0.0
10.0
true
true
"" ""
PENS
"female" 1.0 0 -2674135 true "" "plot stampede-victims-female"
"male" 1.0 0 -13345367 true "" "plot stampede-victims-male"

PLOT
260
564
510
716
Fire Deaths by Age Group
NIL
NIL
0.0
10.0
0.0
10.0
true
true
"" ""
PENS
"child" 1.0 0 -2064490 true "" "plot fire-victims-child"
"adult" 1.0 0 -5825686 true "" "plot fire-victims-adult"
"senior" 1.0 0 -8630108 true "" "plot fire-victims-elderly"
```

```
PLOT
524
564
804
716
Stampede Deaths by Age Group
NIL
NIL
0.0
10.0
0.0
10.0
true
true
"" ""
PENS
"child" 1.0 0 -2064490 true "" "plot stampede-victims-child"
"adult" 1.0 0 -5825686 true "" "plot stampede-victims-adult"
"senior" 1.0 0 -8630108 true "" "plot stampede-victims-elderly"

SLIDER
16
111
194
144
fire-speed
fire-speed
1
20
10.0
1
1
NIL
HORIZONTAL

SLIDER
15
68
194
101
number-fires
number-fires
1
5
1.0
1
1
NIL
HORIZONTAL

SLIDER
206
69
384
102
perc-male-female
perc-male-female
0.0
```

```
1.0
0.55
0.05
1
NIL
HORIZONTAL

SLIDER
206
112
385
145
perc-child
perc-child
0.0
1.0
0.15
0.05
1
NIL
HORIZONTAL

SLIDER
206
155
384
188
perc-adults
perc-adults
0.0
1.0
0.85
0.05
1
NIL
HORIZONTAL

SLIDER
16
153
194
186
num-survivors-per-patch
num-survivors-per-patch
5
20
20.0
1
1
NIL
HORIZONTAL

SLIDER
206
196
383
229
mean-mass-female
mean-mass-female
```

55
80
55.0
2
1
NIL
HORIZONTAL

SLIDER
206
238
384
271
mean-mass-male
mean-mass-male
65
100
65.0
5
1
NIL
HORIZONTAL

@#$#@#$#@
WHAT IS IT?

(a general understanding of what the model is trying to show or explain)

HOW IT WORKS

(what rules the agents use to create the overall behavior of the model)

HOW TO USE IT

(how to use the model, including a description of each of the items in the Interface tab)

THINGS TO NOTICE

(suggested things for the user to notice while running the model)

THINGS TO TRY

(suggested things for the user to try to do (move sliders, switches, etc.) with the model)

EXTENDING THE MODEL

(suggested things to add or change in the Code tab to make the model more complicated, detailed, accurate, etc.)

NETLOGO FEATURES

(interesting or unusual features of NetLogo that the model uses, particularly in the Code tab; or where workarounds were needed for missing features)

RELATED MODELS

(models in the NetLogo Models Library and elsewhere which are of related interest)

CREDITS AND REFERENCES

(a reference to the model's URL on the web if it has one, as well as any other
necessary credits, citations, and links)
@#$#@#$#@
default
true
0
Polygon -7500403 true true 150 5 40 250 150 205 260 250

airplane
true
0
Polygon -7500403 true true 150 0 135 15 120 60 120 105 15 165 15 195 120 180 135
240 105 270 120 285 150 270 180 285 210 270 165 240 180 180 285 195 285 165 180 105
180 60 165 15

arrow
true
0
Polygon -7500403 true true 150 0 0 150 105 150 105 293 195 293 195 150 300 150

box
false
0
Polygon -7500403 true true 150 285 285 225 285 75 150 135
Polygon -7500403 true true 150 135 15 75 150 15 285 75
Polygon -7500403 true true 15 75 15 225 150 285 150 135
Line -16777216 false 150 285 150 135
Line -16777216 false 150 135 15 75
Line -16777216 false 150 135 285 75

bug
true
0
Circle -7500403 true true 96 182 108
Circle -7500403 true true 110 127 80
Circle -7500403 true true 110 75 80
Line -7500403 true 150 100 80 30
Line -7500403 true 150 100 220 30

butterfly
true
0
Polygon -7500403 true true 150 165 209 199 225 225 225 255 195 270 165 255 150 240
Polygon -7500403 true true 150 165 89 198 75 225 75 255 105 270 135 255 150 240
Polygon -7500403 true true 139 148 100 105 55 90 25 90 10 105 10 135 25 180 40 195
85 194 139 163
Polygon -7500403 true true 162 150 200 105 245 90 275 90 290 105 290 135 275 180
260 195 215 195 162 165
Polygon -16777216 true false 150 255 135 225 120 150 135 120 150 105 165 120 180
150 165 225
Circle -16777216 true false 135 90 30
Line -16777216 false 150 105 195 60
Line -16777216 false 150 105 105 60

car
false

0
Polygon -7500403 true true 300 180 279 164 261 144 240 135 226 132 213 106 203 84
185 63 159 50 135 50 75 60 0 150 0 165 0 225 300 225 300 180
Circle -16777216 true false 180 180 90
Circle -16777216 true false 30 180 90
Polygon -16777216 true false 162 80 132 78 134 135 209 135 194 105 189 96 180 89
Circle -7500403 true true 47 195 58
Circle -7500403 true true 195 195 58

circle
false
0
Circle -7500403 true true 0 0 300

circle 2
false
0
Circle -7500403 true true 0 0 300
Circle -16777216 true false 30 30 240

cow
false
0
Polygon -7500403 true true 200 193 197 249 179 249 177 196 166 187 140 189 93 191
78 179 72 211 49 209 48 181 37 149 25 120 25 89 45 72 103 84 179 75 198 76 252 64
272 81 293 103 285 121 255 121 242 118 224 167
Polygon -7500403 true true 73 210 86 251 62 249 48 208
Polygon -7500403 true true 25 114 16 195 9 204 23 213 25 200 39 123

cylinder
false
0
Circle -7500403 true true 0 0 300

dot
false
0
Circle -7500403 true true 90 90 120

face happy
false
0
Circle -7500403 true true 8 8 285
Circle -16777216 true false 60 75 60
Circle -16777216 true false 180 75 60
Polygon -16777216 true false 150 255 90 239 62 213 47 191 67 179 90 203 109 218 150
225 192 218 210 203 227 181 251 194 236 217 212 240

face neutral
false
0
Circle -7500403 true true 8 7 285
Circle -16777216 true false 60 75 60
Circle -16777216 true false 180 75 60
Rectangle -16777216 true false 60 195 240 225

face sad
false
0

```
Circle -7500403 true true 8 8 285
Circle -16777216 true false 60 75 60
Circle -16777216 true false 180 75 60
Polygon -16777216 true false 150 168 90 184 62 210 47 232 67 244 90 220 109 205 150
198 192 205 210 220 227 242 251 229 236 206 212 183

fish
false
0
Polygon -1 true false 44 131 21 87 15 86 0 120 15 150 0 180 13 214 20 212 45 166
Polygon -1 true false 135 195 119 235 95 218 76 210 46 204 60 165
Polygon -1 true false 75 45 83 77 71 103 86 114 166 78 135 60
Polygon -7500403 true true 30 136 151 77 226 81 280 119 292 146 292 160 287 170 270
195 195 210 151 212 30 166
Circle -16777216 true false 215 106 30

flag
false
0
Rectangle -7500403 true true 60 15 75 300
Polygon -7500403 true true 90 150 270 90 90 30
Line -7500403 true 75 135 90 135
Line -7500403 true 75 45 90 45

flower
false
0
Polygon -10899396 true false 135 120 165 165 180 210 180 240 150 300 165 300 195
240 195 195 165 135
Circle -7500403 true true 85 132 38
Circle -7500403 true true 130 147 38
Circle -7500403 true true 192 85 38
Circle -7500403 true true 85 40 38
Circle -7500403 true true 177 40 38
Circle -7500403 true true 177 132 38
Circle -7500403 true true 70 85 38
Circle -7500403 true true 130 25 38
Circle -7500403 true true 96 51 108
Circle -16777216 true false 113 68 74
Polygon -10899396 true false 189 233 219 188 249 173 279 188 234 218
Polygon -10899396 true false 180 255 150 210 105 210 75 240 135 240

house
false
0
Rectangle -7500403 true true 45 120 255 285
Rectangle -16777216 true false 120 210 180 285
Polygon -7500403 true true 15 120 150 15 285 120
Line -16777216 false 30 120 270 120

leaf
false
0
Polygon -7500403 true true 150 210 135 195 120 210 60 210 30 195 60 180 60 165 15
135 30 120 15 105 40 104 45 90 60 90 90 105 105 120 120 120 105 60 120 60 135 30
150 15 165 30 180 60 195 60 180 120 195 120 210 105 240 90 255 90 263 104 285 105
270 120 285 135 240 165 240 180 270 195 240 210 180 210 165 195
Polygon -7500403 true true 135 195 135 240 120 255 105 255 105 285 135 285 165 240
165 195
```

```
line
true
0
Line -7500403 true 150 0 150 300

line half
true
0
Line -7500403 true 150 0 150 150

pentagon
false
0
Polygon -7500403 true true 150 15 15 120 60 285 240 285 285 120

person
false
0
Circle -7500403 true true 110 5 80
Polygon -7500403 true true 105 90 120 195 90 285 105 300 135 300 150 225 165 300
195 300 210 285 180 195 195 90
Rectangle -7500403 true true 127 79 172 94
Polygon -7500403 true true 195 90 240 150 225 180 165 105
Polygon -7500403 true true 105 90 60 150 75 180 135 105

plant
false
0
Rectangle -7500403 true true 135 90 165 300
Polygon -7500403 true true 135 255 90 210 45 195 75 255 135 285
Polygon -7500403 true true 165 255 210 210 255 195 225 255 165 285
Polygon -7500403 true true 135 180 90 135 45 120 75 180 135 210
Polygon -7500403 true true 165 180 165 210 225 180 255 120 210 135
Polygon -7500403 true true 135 105 90 60 45 45 75 105 135 135
Polygon -7500403 true true 165 105 165 135 225 105 255 45 210 60
Polygon -7500403 true true 135 90 120 45 150 15 180 45 165 90

sheep
false
15
Circle -1 true true 203 65 88
Circle -1 true true 70 65 162
Circle -1 true true 150 105 120
Polygon -7500403 true false 218 120 240 165 255 165 278 120
Circle -7500403 true false 214 72 67
Rectangle -1 true true 164 223 179 298
Polygon -1 true true 45 285 30 285 30 240 15 195 45 210
Circle -1 true true 3 83 150
Rectangle -1 true true 65 221 80 296
Polygon -1 true true 195 285 210 285 210 240 240 210 195 210
Polygon -7500403 true false 276 85 285 105 302 99 294 83
Polygon -7500403 true false 219 85 210 105 193 99 201 83

square
false
0
Rectangle -7500403 true true 30 30 270 270
```

```
square 2
false
0
Rectangle -7500403 true true 30 30 270 270
Rectangle -16777216 true false 60 60 240 240

star
false
0
Polygon -7500403 true true 151 1 185 108 298 108 207 175 242 282 151 216 59 282 94
175 3 108 116 108

target
false
0
Circle -7500403 true true 0 0 300
Circle -16777216 true false 30 30 240
Circle -7500403 true true 60 60 180
Circle -16777216 true false 90 90 120
Circle -7500403 true true 120 120 60

tree
false
0
Circle -7500403 true true 118 3 94
Rectangle -6459832 true false 120 195 180 300
Circle -7500403 true true 65 21 108
Circle -7500403 true true 116 41 127
Circle -7500403 true true 45 90 120
Circle -7500403 true true 104 74 152

triangle
false
0
Polygon -7500403 true true 150 30 15 255 285 255

triangle 2
false
0
Polygon -7500403 true true 150 30 15 255 285 255
Polygon -16777216 true false 151 99 225 223 75 224

truck
false
0
Rectangle -7500403 true true 4 45 195 187
Polygon -7500403 true true 296 193 296 150 259 134 244 104 208 104 207 194
Rectangle -1 true false 195 60 195 105
Polygon -16777216 true false 238 112 252 141 219 141 218 112
Circle -16777216 true false 234 174 42
Rectangle -7500403 true true 181 185 214 194
Circle -16777216 true false 144 174 42
Circle -16777216 true false 24 174 42
Circle -7500403 false true 24 174 42
Circle -7500403 false true 144 174 42
Circle -7500403 false true 234 174 42

turtle
true
```

```
0
Polygon -10899396 true false 215 204 240 233 246 254 228 266 215 252 193 210
Polygon -10899396 true false 195 90 225 75 245 75 260 89 269 108 261 124 240 105
225 105 210 105
Polygon -10899396 true false 105 90 75 75 55 75 40 89 31 108 39 124 60 105 75 105
90 105
Polygon -10899396 true false 132 85 134 64 107 51 108 17 150 2 192 18 192 52 169 65
172 87
Polygon -10899396 true false 85 204 60 233 54 254 72 266 85 252 107 210
Polygon -7500403 true true 119 75 179 75 209 101 224 135 220 225 175 261 128 261 81
224 74 135 88 99

wheel
false
0
Circle -7500403 true true 3 3 294
Circle -16777216 true false 30 30 240
Line -7500403 true 150 285 150 15
Line -7500403 true 15 150 285 150
Circle -7500403 true true 120 120 60
Line -7500403 true 216 40 79 269
Line -7500403 true 40 84 269 221
Line -7500403 true 40 216 269 79
Line -7500403 true 84 40 221 269

wolf
false
0
Polygon -16777216 true false 253 133 245 131 245 133
Polygon -7500403 true true 2 194 13 197 30 191 38 193 38 205 20 226 20 257 27 265
38 266 40 260 31 253 31 230 60 206 68 198 75 209 66 228 65 243 82 261 84 268 100
267 103 261 77 239 79 231 100 207 98 196 119 201 143 202 160 195 166 210 172 213
173 238 167 251 160 248 154 265 169 264 178 247 186 240 198 260 200 271 217 271 219
262 207 258 195 230 192 198 210 184 227 164 242 144 259 145 284 151 277 141 293 140
299 134 297 127 273 119 270 105
Polygon -7500403 true true -1 195 14 180 36 166 40 153 53 140 82 131 134 133 159
126 188 115 227 108 236 102 238 98 268 86 269 92 281 87 269 103 269 113

x
false
0
Polygon -7500403 true true 270 75 225 30 30 225 75 270
Polygon -7500403 true true 30 75 75 30 270 225 225 270
@#$#@#$#@
NetLogo 6.2.0
@#$#@#$#@
@#$#@#$#@
@#$#@#$#@
<experiments>
  <experiment name="run" repetitions="1" runMetricsEveryStep="false">
    <setup>setup</setup>
    <go>go</go>
    <timeLimit steps="1000"/>
    <exitCondition>count survivors = 0</exitCondition>
    <metric>female-escapees</metric>
    <metric>female-fire-deaths</metric>
    <metric>female-stampede-deaths</metric>
    <metric>male-escapees</metric>
    <metric>male-fire-deaths</metric>
```

```xml
    <metric>male-stampede-deaths</metric>
    <metric>child-escapees</metric>
    <metric>child-fire-deaths</metric>
    <metric>child-stampede-deaths</metric>
    <metric>adult-escapees</metric>
    <metric>adult-fire-deaths</metric>
    <metric>adult-stampede-deaths</metric>
    <metric>elderly-escapees</metric>
    <metric>elderly-fire-deaths</metric>
    <metric>elderly-stampede-deaths</metric>
    <enumeratedValueSet variable="perc-adults">
      <value value="0.8708"/>
    </enumeratedValueSet>
    <enumeratedValueSet variable="perc-male-female">
      <value value="0.55"/>
    </enumeratedValueSet>
    <enumeratedValueSet variable="behaviour">
      <value value="&quot;follow&quot;"/>
    </enumeratedValueSet>
    <enumeratedValueSet variable="use-panic?">
      <value value="true"/>
    </enumeratedValueSet>
    <enumeratedValueSet variable="num-survivors-per-patch">
      <value value="10"/>
    </enumeratedValueSet>
    <enumeratedValueSet variable="use-stairs?">
      <value value="true"/>
    </enumeratedValueSet>
    <enumeratedValueSet variable="perc-child">
      <value value="0.15"/>
    </enumeratedValueSet>
    <enumeratedValueSet variable="number-fires">
      <value value="1"/>
    </enumeratedValueSet>
    <enumeratedValueSet variable="fire-speed">
      <value value="10"/>
    </enumeratedValueSet>
    <enumeratedValueSet variable="max-vision">
      <value value="100"/>
    </enumeratedValueSet>
    <enumeratedValueSet variable="random-fire?">
      <value value="false"/>
    </enumeratedValueSet>
    <enumeratedValueSet variable="threshold">
      <value value="60"/>
    </enumeratedValueSet>
  </experiment>
</experiments>
@#$#@#$#@
@#$#@#$#@
default
0.0
-0.2 0 0.0 1.0
0.0 1 1.0 0.0
0.2 0 0.0 1.0
link direction
true
0
Line -7500403 true 150 150 90 180
```

```
Line -7500403 true 150 150 210 180
@#$#@#$#@
0
@#$#@#$#@
```