breed [ players player ]

breed [ monsters monster ]

breed [ flags flag]

breed [ enemys enemy ]

turtles-own [ home-pos ]

globals [

level ;; Current Level

player-lives ;; Remaining Lives

level-over? ;; True when a level is complete

dead? ;; True when player is loses a life

tool ;; The currently selected tool

]

;;;;;;;;;;;;;;;;;;;;;;

;; Setup Procedures ;;

;;;;;;;;;;;;;;;;;;;;;;

;; Setup a new level

to new

if user-yes-or-no? "Do you really want to clear the level?"

[

clear-all

set level 0

set player-lives 3

set level-over? false

set dead? false

set tool "Eraser"

create-players 1

[

set shape "player"

setxy 10 10

]

create-monsters 1

[

set shape "monster"

setxy 10 9

set heading 0

]

create-flags 1

[

set shape "flag"

setxy 10 8

]

foreach range 3 [ n ->

create-enemys 1

[

set shape "enemy"

set color item n enemy-colors

setxy 10 (7 - n)

set heading 0

]

]

show enemy-colors

ask turtles

[ set home-pos list xcor ycor ]

]

end

;; npc colors

to-report enemy-colors

report [ 15 45 95 ]

end

;;;;;;;;;;;;;;;;;;;;;;;;

;; Runtime Procedures ;;

;;;;;;;;;;;;;;;;;;;;;;;;

;; If the mouse is down, use the Current Tool on the patch the mouse is over

to draw

if mouse-down?

[

;; Eraser Tool

if tool = "Eraser"

[ erase ]

;; Wall Tool - Draws a Wall

if tool = "Draw Wall"

[ draw-boundary gray ]

;; Gate Tool - Draws Gate

if tool = "Draw Gate"

[ draw-boundary brown ]

;; flag Tool

if tool = "Place Flag"

[ place-flag ]

;; Place water

if tool = "Draw Water"

[ draw-boundary cyan ]

;; Place lava

if tool = "Draw Lava"

[ draw-boundary orange ]

;; Place Marsh

if tool = "Draw Marsh"

[ draw-boundary 53 ]

;; Player Tool

if tool = "Place Player"

[ place-player ]

;; Monster Tool

if tool = "Place Monster"

[ place-monster ]

;; ;;npc Tool

if tool = "Place Enemy"

[ place-enemy ]

]

end

;;;;;;;;;;;;;;;;;;;;;

;; Tool Procedures ;;

;;;;;;;;;;;;;;;;;;;;;

;; Clears structures

to erase

ask patch (round mouse-xcor) (round mouse-ycor)

[

set pcolor black

]

end

;; Draws structures

to draw-boundary [ boundary-color ]

ask patch (round mouse-xcor) (round mouse-ycor)

[

ifelse not any? turtles-here

[ set pcolor boundary-color ]

[

ifelse boundary-color = gray

[

ifelse any? players-here

[ user-message "You cannot place a gate on top of player or a pellet." ]

[ set pcolor boundary-color ]

]

[ user-message "You cannot place a wall on top of player, a ghost, or a pellet." ]

]

]

end

;; Place flag

to place-flag

ifelse [pcolor] of patch round mouse-xcor mouse-ycor != black

[ user-message "You must place flag on a corridor space, not a wall or a gate." ]

[

ask flags

[ setxy (round mouse-xcor) (round mouse-ycor) ]

]

end

;; Place player

to place-player

ifelse [pcolor] of patch round mouse-xcor mouse-ycor != black

[ user-message "You must place player on a corridor space, not a wall or a gate." ]

[

ask players

[ setxy (round mouse-xcor) (round mouse-ycor) ]

]

end

;; Place monster

to place-monster

ifelse [pcolor] of patch mouse-xcor mouse-ycor != black

[ user-message "You must place player on a corridor space, not a wall or a gate." ]

[

ask one-of monsters

[ setxy (round mouse-xcor) (round mouse-ycor) ]

]

end

;; Place npc's

to place-enemy

ifelse [pcolor] of patch round mouse-xcor mouse-ycor != black

[ user-message "You must place player on a corridor space, not a wall or a gate." ]

[

ask one-of enemys with [ color = item (which-enemy - 1) enemy-colors ]

[ setxy (round mouse-xcor) (round mouse-ycor) ]

]

end

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;; Saving and Loading Procedures ;;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;; Change the Level

to set-level

if level > 0

[

if not user-yes-or-no? "Are you sure you want to change the level number of this map?"

[ stop ]

]

let temp 0

while[ temp <= 0 ]

[

set temp read-from-string user-input "Input New Level Number:"

if temp <= 0

[ user-message "The level must be a positive number." ]

]

set level temp

end

;;save the file

to save-level

if level <= 0

[

user-message "You must choose a positive level number before saving."

set-level

]

let filepath (word "../playermap" level ".csv")

ifelse user-yes-or-no? (word "File will be saved at: " filepath

"\nIf this file already exists, it will be overwritten.\nAre you sure you want to save?")

[

export-world filepath

user-message "File Saved."

]

[ user-message "Save Canceled. File not saved." ]

end

;; Load a level

to load-level

let choice 0

while[ choice <= 0 ]

[

set choice read-from-string user-input "Load What Level? (1 or greater)"

if choice <= 0

[ user-message "You must choose a positive level number to load." ]

]

let filepath (word "playermap" choice ".csv")

ifelse user-yes-or-no? (word "Load File: " filepath

"\nThis will clear your current level and replace it with the level loaded."

"\nAre you sure you want to Load?")

[

import-world filepath

set tool "Eraser"

user-message "File Loaded."

]

[ user-message "Load Canceled. File not loaded." ]

end