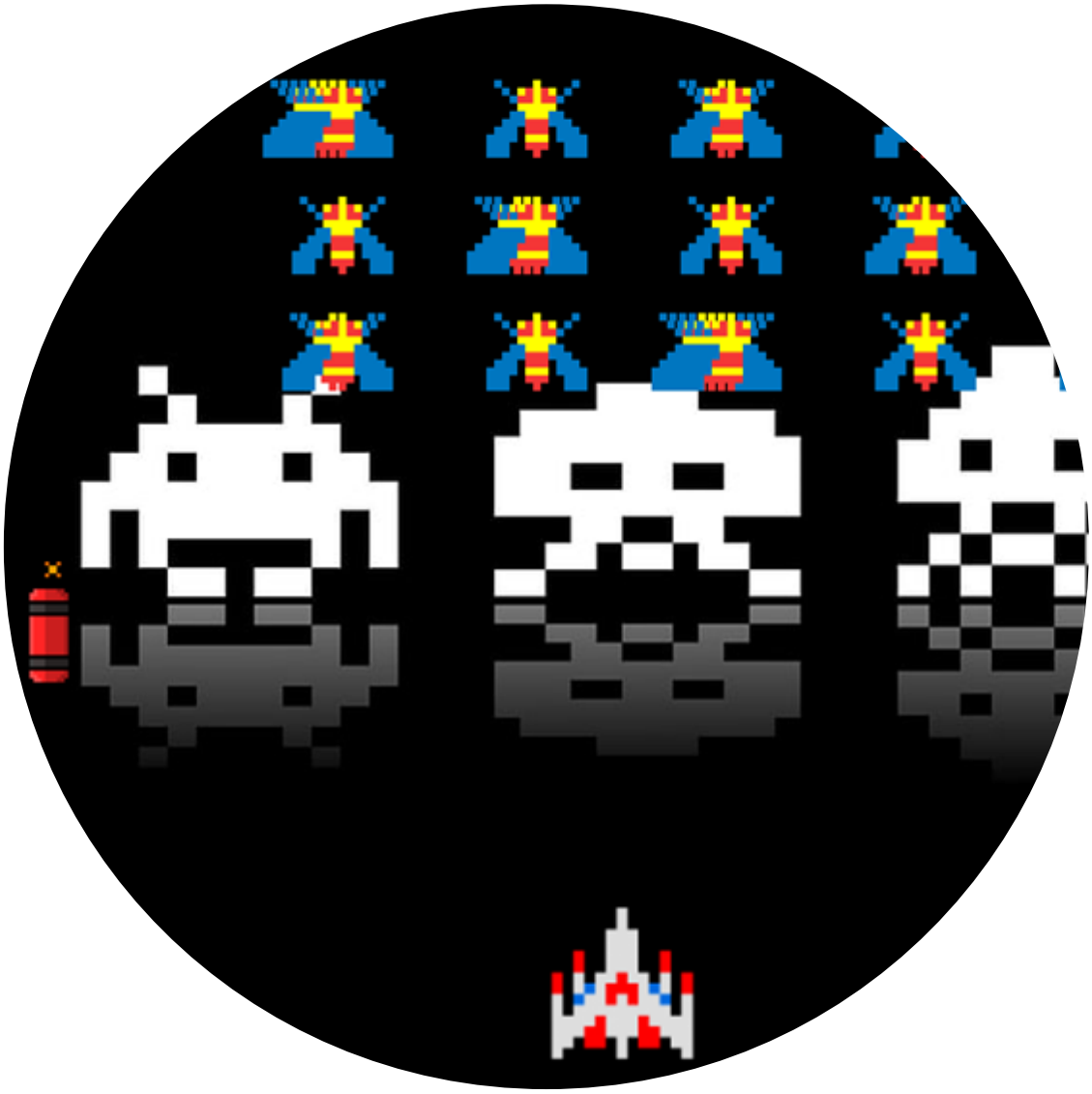
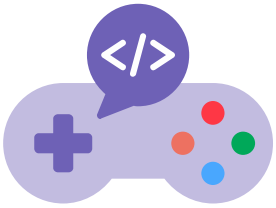


PYGAME #GAME7



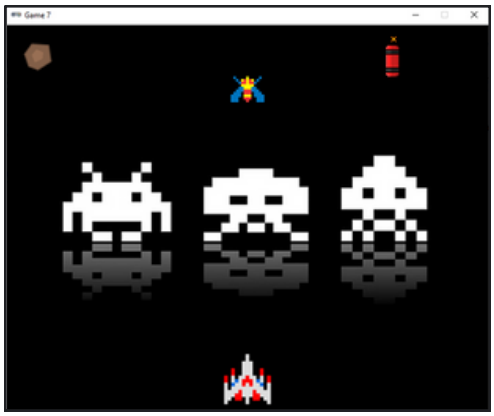
Update : 01May23

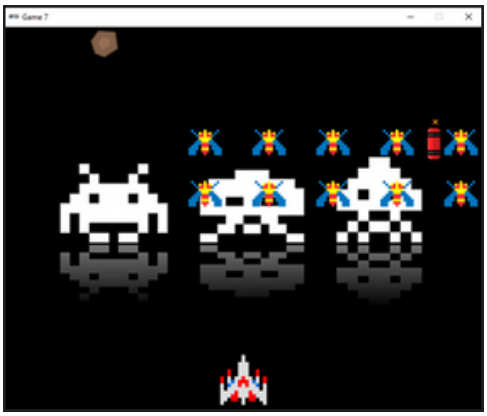
KIDDEE LAB

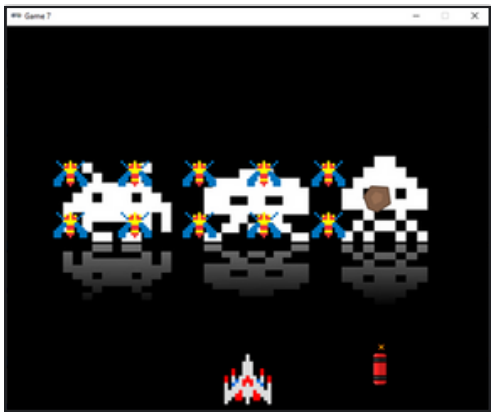
Contents

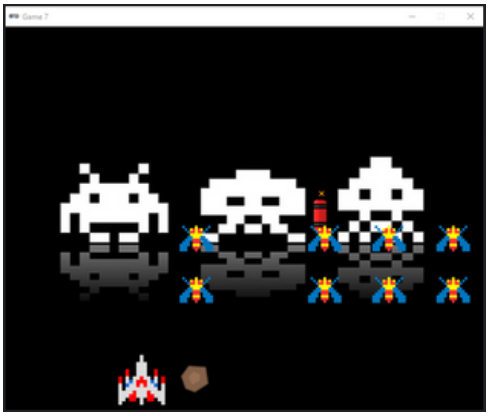
PYGAME#GAME7 OVERVIEW

Galaga









PYGAME#GAME7 OVERVIEW CONT.

VISIT THIS LINK FOR IMAGES/MUSIC/SOUNDS :

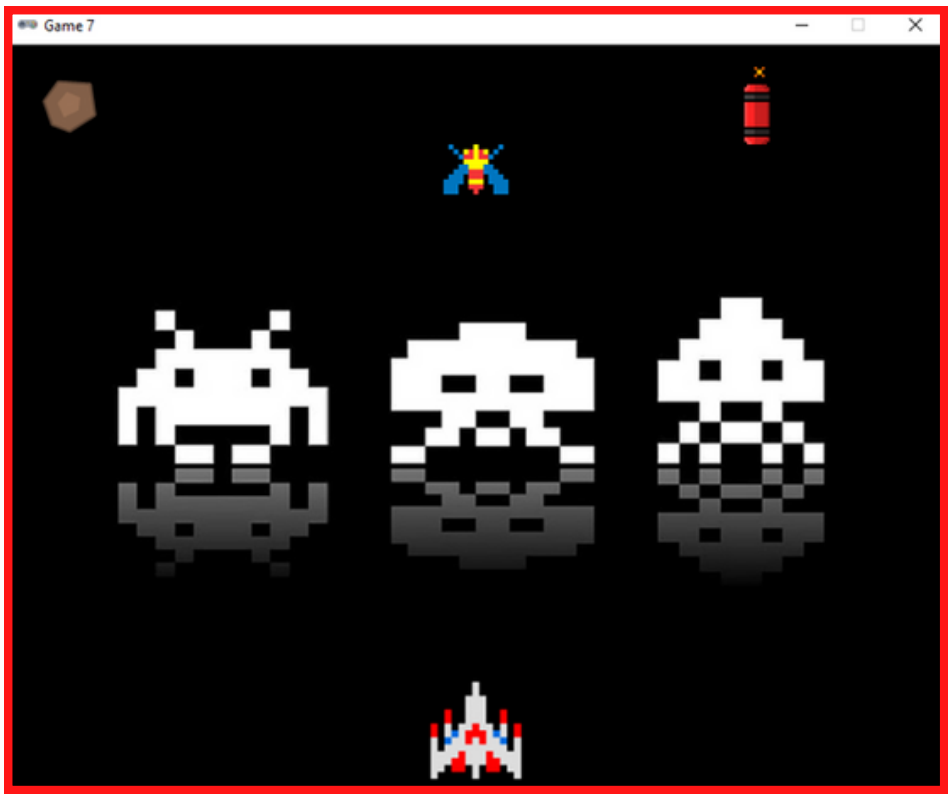
HTTPS://BIT.LY/40MTJ1I

SCAN ME

EXAMPLE 7.1 (CREATE SCREEN AND ACTORS)

CREATE PYGAME WITH FOLLOWING THE DETAILS

1. Create a screen of width and height according to the screen size.
2. Title of the screen is 'Game 7'.
3. Create a galaga background with given image as on the right.
3. Create sprites of
    - Galaga Space Ship
    - Bug
    - Dynamite
    - Meteor



Expected Result

EXAMPLE 7.1 CONT.

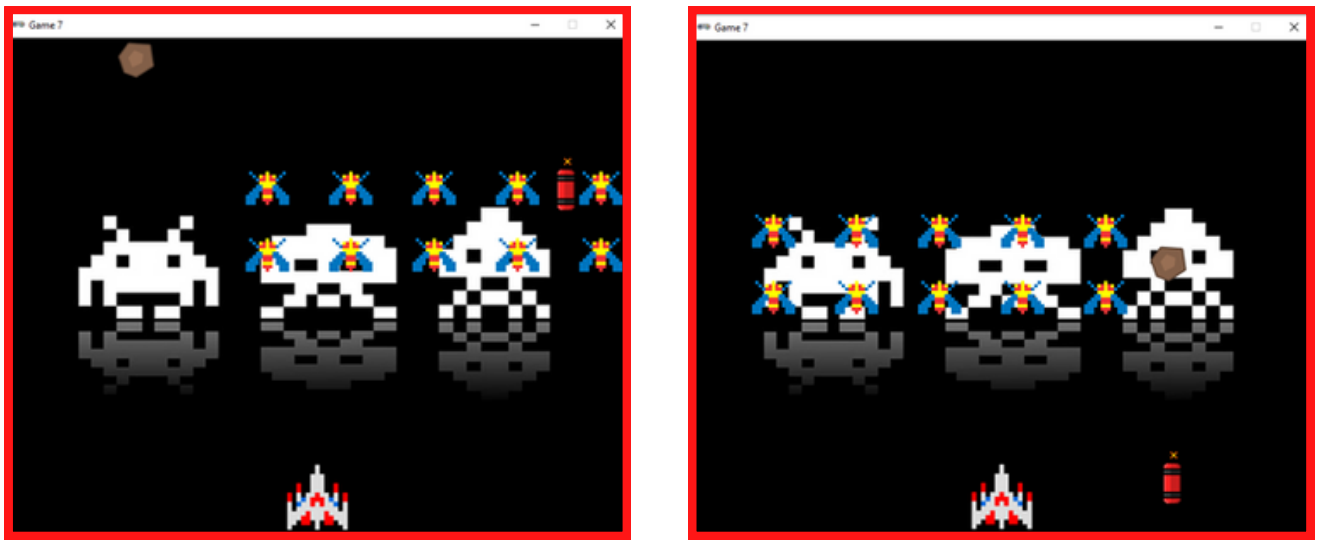
SOURCE CODE

```
##To set screen to be in the middle
import os
os.environ['SDL_VIDEO_WINDOW_POS'] = "%d,%d" % (300,50)
import pygame
import pgzrun
import random

HEIGHT = 600
WIDTH = 750
TITLE = 'Game 7'

#actors
galagaSprite = Actor('galaga')
galagaSprite.pos = (WIDTH/2, 550)
meteorSprite = Actor('meteor')
meteorSprite.pos = (50, 50)
bugSprite = Actor('bug')
bugSprite.pos = (WIDTH/2,100)
dynamiteSprite = Actor('dynamite')
dynamiteSprite.pos = (600, 50)

def update():
pass

def draw():
screen.blit('galagabg', (0,0))
galagaSprite.draw()
meteorSprite.draw()
bugSprite.draw()
dynamiteSprite.draw()

pgzrun.go()
```

EXAMPLE 7.2
(MOVING RANDOMLY OF SPRITES AND IMAGES)

FROM EXAMPLE 7.1, PERFORM THE FOLLOWING DETAILS

1. Define player to move only in x-axis direction. Player is to be moved with a and d from the keyboard.
2. Dynamite and Meteor are to be dropped in y-axis.
3. Create a group of 10 bug enemies (2 row, 5 column) :
    - A group of bug is to be moved in x-axis and y-axis.
    - If the first bug (from left) touch the x-axis = 0 of the screen, a group of bug will move 50 pixel down in y-axis and continue to move in +x-axis.
    - If the last bug (from right) touch the x-axis = width of the screen, a group of bug will move 50 pixel down in y-axis and continue to move in x-axis.



Expected Results

EXAMPLE 7.2 CONT.

SOURCE CODE

```
##To set screen to be in the middle
import os
os.environ['SDL_VIDEO_WINDOW_POS'] = "%d,%d" % (300,50)
import pygame
import pgzrun
import random

HEIGHT = 600
WIDTH = 750
TITLE = 'Game 7'

#actors
galagaSprite = Actor('galaga')
galagaSprite.pos = (WIDTH/2, 550)
meteorSprite = Actor('meteor')
meteorSprite.pos = (50, 50)
bugSprite = Actor('bug')
bugSprite.pos = (WIDTH/2,100)
dynamiteSprite = Actor('dynamite')
dynamiteSprite.pos = (600, 50)

#dynamic
meteorSpeed = 10
dynamiteSpeed = 10

#Group of enemy
enemies = []
bugXdirection = 1
bugMoveDown = False

def galagaMove():
     if keyboard.a == True :
           galagaSprite.x = galagaSprite.x - 10
     elif keyboard.d == True :
           galagaSprite.x = galagaSprite.x + 10

     if galagaSprite.x >= WIDTH :
           galagaSprite.x = WIDTH
     if galagaSprite.x <= 0 :
           galagaSprite.x = 0

def meteorMove():
     meteorSprite.y = meteorSprite.y + meteorSpeed
     if meteorSprite.y >= HEIGHT:
           meteorReset()

def meteorReset():
     global meteorSpeed
     meteorSprite.pos = (random.randint(0, WIDTH), 0)
     meteorSpeed = random.randint(5,15)
```

EXAMPLE 7.2 CONT.

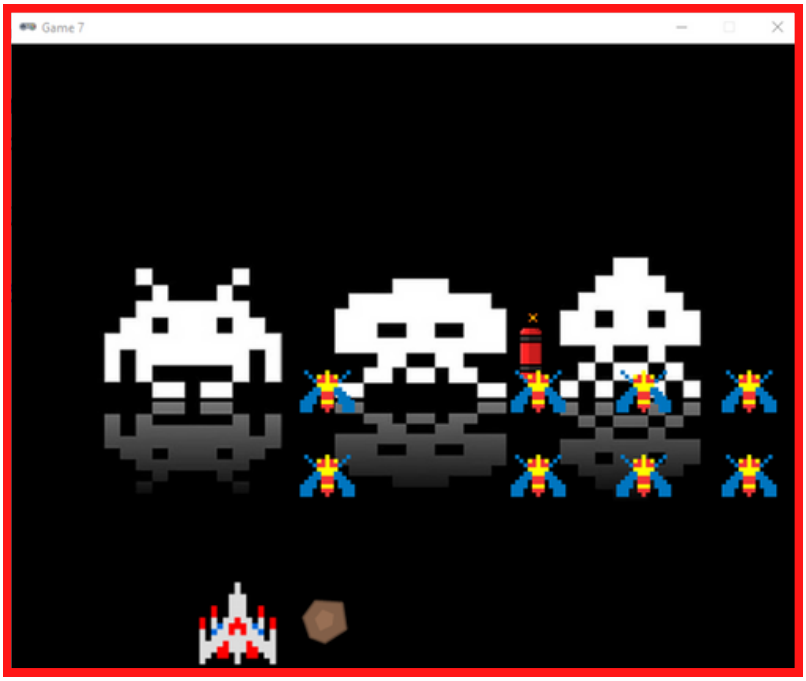
SOURCE CODE CONT.

```
def dynamiteMove():
     dynamiteSprite.y = dynamiteSprite.y + dynamiteSpeed
     if dynamiteSprite.y >= HEIGHT:
          dynamiteReset()

def dynamiteReset():
     global dynamiteSpeed
     dynamiteSprite.pos = (random.randint(0, WIDTH), 0)
     dynamiteSpeed = random.randint(10,20)

def update():
     galagaMove()
     meteorMove()
     dynamiteMove()
     enemyMove()

##Create a group of bug
def enemyCreate():
     for row in range(0,2,1):
          for column in range(0,5,1):
               enemies.append(Actor('bug'))
#enemies.append(bugSprite) --> not working
               enemies[-1].x = (100 + 100*column)
               enemies[-1].y = (80 + 80*row)
enemyCreate() #call only once

def enemyMove():
     global bugXdirection
     global bugMoveDown
     if (len(enemies)>0) and (enemies[-1].x >= WIDTH or
enemies[0].x <= 0 ):
          bugXdirection = bugXdirection*-1 #toggle direction
          bugMoveDown = True
     for enemy in enemies:
          enemy.x = enemy.x + 5*bugXdirection
          if bugMoveDown == True:
               enemy.y = enemy.y + 50
     bugMoveDown = False

def draw():
screen.blit('galagabg', (0,0))
galagaSprite.draw()
meteorSprite.draw()
#bugSprite.draw()
dynamiteSprite.draw()
     for enemy in enemies:
          enemy.draw()

pgzrun.go()
```

EXAMPLE 7.3
(GAME CONDITIONS: SHOOT THE ENEMY AND AVOID THE DYNAMITE AND METEOR)

FROM EXAMPLE 7.2, PERFORM THE FOLLOWING DETAILS

1. Player is to shoot the bug enemy.
    - When a bug is shot, bug and bullet will be removed.
    - When a bug is shot, score of 1 will be increment.
2. When score >= 5, show 'win' message.
3. When Player hit with Dynamite or Meteor, show 'gameover' message.
4. When Player hit with bug enemy, show 'gameover' message.



Expected Result

EXAMPLE 7.3 CONT.

SOURCE CODE

```
##To set screen to be in the middle
import os
os.environ['SDL_VIDEO_WINDOW_POS'] = "%d,%d" % (300,50)
import pygame
import pgzrun
import random

HEIGHT = 600
WIDTH = 750
TITLE = 'Game 7'

#actors
galagaSprite = Actor('galaga')
galagaSprite.pos = (WIDTH/2, 550)
meteorSprite = Actor('meteor')
meteorSprite.pos = (50, 50)
bugSprite = Actor('bug')
bugSprite.pos = (WIDTH/2,100)
dynamiteSprite = Actor('dynamite')
dynamiteSprite.pos = (600, 50)

#dynamic
meteorSpeed = 10
dynamiteSpeed = 10

#Group of enemy
enemies = []
bugXdirection = 1
bugMoveDown = False

#Bullets
bullets = []
bulletActive = False

#score
score = 0

def on_key_down(key):
     global bulletActive
     if key == keys.SPACE and bulletActive == False:
           bullets.append(Actor('laserred'))
           bullets[-1].pos = galagaSprite.pos
           bulletActive = True
```

EXAMPLE 7.3 CONT.

SOURCE CODE CONT.

```
def bulletMove():
     global bulletActive
     global score
     for bullet in bullets:
           if bullet.y <= 0:
                 bullets.remove(bullet)
                 bulletActive = False
           else :
                 bullet.y = bullet.y - 10

           for enemy in enemies :
                 if enemy.colliderect(bullet):
                       enemies.remove(enemy)
                       bullets.remove(bullet)
                       bulletActive = False
                       score = score + 1
                       print('score :', score)
def galagaMove():
if keyboard.a == True :
           galagaSprite.x = galagaSprite.x - 10
elif keyboard.d == True :
           galagaSprite.x = galagaSprite.x + 10

if galagaSprite.x >= WIDTH :
           galagaSprite.x = WIDTH
if galagaSprite.x <= 0 :
           galagaSprite.x = 0

     if galagaSprite.colliderect(dynamiteSprite):
           print('gameover1')
           dynamiteReset()
     elif galagaSprite.colliderect(meteorSprite):
           print('gameover2')
           meteorReset()
```

EXAMPLE 7.3 CONT.

SOURCE CODE CONT.

```
def meteorMove():
meteorSprite.y = meteorSprite.y + meteorSpeed
if meteorSprite.y >= HEIGHT:
            meteorReset()

def meteorReset():
global meteorSpeed
meteorSprite.pos = (random.randint(0, WIDTH), 0)
meteorSpeed = random.randint(5,15)

def dynamiteMove():
dynamiteSprite.y = dynamiteSprite.y + dynamiteSpeed
if dynamiteSprite.y >= HEIGHT:
            dynamiteReset()

def dynamiteReset():
global dynamiteSpeed
dynamiteSprite.pos = (random.randint(0, WIDTH), 0)
dynamiteSpeed = random.randint(10,20)

def update():
galagaMove()
meteorMove()
dynamiteMove()
enemyMove()
bulletMove()

##Create a group of bug
def enemyCreate():
for row in range(0,2,1):
            for column in range(0,5,1):
                  enemies.append(Actor('bug'))
#enemies.append(bugSprite) --> not working
                  enemies[-1].x = (100 + 100*column)
                  enemies[-1].y = (80 + 80*row)
enemyCreate()#call only once

def enemyMove():
global bugXdirection
global bugMoveDown
if (len(enemies)>0) and (enemies[-1].x >= WIDTH or enemies[0].x <= 0
):
            bugXdirection = bugXdirection*-1#toggle direction
            bugMoveDown = True
for enemy in enemies:
            enemy.x = enemy.x + 5*bugXdirection
            if bugMoveDown == True:
                  enemy.y = enemy.y + 50

            if enemy.colliderect(galagaSprite):
                  print('gameover3')

bugMoveDown = False
```

EXAMPLE 7.3 CONT.

SOURCE CODE CONT.

```
def meteorMove():
meteorSprite.y = meteorSprite.y + meteorSpeed
if meteorSprite.y >= HEIGHT:
            meteorReset()

def meteorReset():
global meteorSpeed
meteorSprite.pos = (random.randint(0, WIDTH), 0)
meteorSpeed = random.randint(5,15)

def dynamiteMove():
dynamiteSprite.y = dynamiteSprite.y + dynamiteSpeed
if dynamiteSprite.y >= HEIGHT:
            dynamiteReset()

def dynamiteReset():
global dynamiteSpeed
dynamiteSprite.pos = (random.randint(0, WIDTH), 0)
dynamiteSpeed = random.randint(10,20)

def update():
galagaMove()
meteorMove()
dynamiteMove()
enemyMove()
bulletMove()

##Create a group of bug
def enemyCreate():
for row in range(0,2,1):
            for column in range(0,5,1):
                  enemies.append(Actor('bug'))
#enemies.append(bugSprite) --> not working
                  enemies[-1].x = (100 + 100*column)
                  enemies[-1].y = (80 + 80*row)
enemyCreate()#call only once

def enemyMove():
global bugXdirection
global bugMoveDown
if (len(enemies)>0) and (enemies[-1].x >= WIDTH or enemies[0].x <= 0
):
            bugXdirection = bugXdirection*-1#toggle direction
            bugMoveDown = True
for enemy in enemies:
            enemy.x = enemy.x + 5*bugXdirection
            if bugMoveDown == True:
                  enemy.y = enemy.y + 50

            if enemy.colliderect(galagaSprite):
                  print('gameover3')

bugMoveDown = False
```

EXAMPLE 7.3 CONT.

SOURCE CODE CONT.

```
def draw():
screen.blit('galagabg', (0,0))
galagaSprite.draw()
meteorSprite.draw()
#bugSprite.draw()
dynamiteSprite.draw()
for enemy in enemies:
            enemy.draw()
for bullet in bullets:
            bullet.draw()

pgzrun.go()
```

EXAMPLE 7.4
(ADD GAME STATES : PLAY, GAME OVER AND WIN)

FROM EXAMPLE 7.3, PERFORM THE FOLLOWING DETAILS

1. Starting with the initial page "Galaga", and Press s to start.
2. When Player hit with either dynamite, meteor or bug enemy, you lost the game:
    - Show 'You Loose' message and able to Restart the game.
3. When Player's score >= 5:
    - Show 'You Win' message and able to Restart the game.

Expected Results

EXAMPLE 7.4 CONT.

SOURCE CODE

```
##To set screen to be in the middle
import os
os.environ['SDL_VIDEO_WINDOW_POS'] = "%d,%d" % (300,50)
import pygame
import pgzrun
import random

HEIGHT = 600
WIDTH = 750
TITLE = 'Game 7'

#actors
galagaSprite = Actor('galaga')
galagaSprite.pos = (WIDTH/2, 550)
meteorSprite = Actor('meteor')
meteorSprite.pos = (50, 50)
bugSprite = Actor('bug')
bugSprite.pos = (WIDTH/2,100)
dynamiteSprite = Actor('dynamite')
dynamiteSprite.pos = (600, 50)

#dynamic
meteorSpeed = 10
dynamiteSpeed = 10

#Group of enemy
enemies = []
bugXdirection = 1
bugMoveDown = False

#Bullets
bullets = []
bulletActive = False

#score
score = 0

#game state
gameState = 'start'

def on_key_down(key):
global bulletActive
if key == keys.SPACE and bulletActive == False:
            bullets.append(Actor('laserred'))
            bullets[-1].pos = galagaSprite.pos
            bulletActive = True
```

EXAMPLE 7.4 CONT.

SOURCE CODE CONT.

```
def bulletMove():
global bulletActive
for bullet in bullets:
            if bullet.y <= 0:
                  bullets.remove(bullet)
                  bulletActive = False
            else :
                  bullet.y = bullet.y - 10

def galagaMove():
global gameState
if keyboard.a == True :
            galagaSprite.x = galagaSprite.x - 10
elif keyboard.d == True :
            galagaSprite.x = galagaSprite.x + 10

if galagaSprite.x >= WIDTH :
            galagaSprite.x = WIDTH
if galagaSprite.x <= 0 :
            galagaSprite.x = 0

if galagaSprite.colliderect(dynamiteSprite):
            gameState = 'gameover'
            dynamiteReset()
elif galagaSprite.colliderect(meteorSprite):
            gameState = 'gameover'
            meteorReset()

def meteorMove():
meteorSprite.y = meteorSprite.y + meteorSpeed
if meteorSprite.y >= HEIGHT:
            meteorReset()

def meteorReset():
global meteorSpeed
meteorSprite.pos = (random.randint(0, WIDTH), 0)
meteorSpeed = random.randint(7,12)

def dynamiteMove():
dynamiteSprite.y = dynamiteSprite.y + dynamiteSpeed
if dynamiteSprite.y >= HEIGHT:
            dynamiteReset()

def dynamiteReset():
global dynamiteSpeed
dynamiteSprite.pos = (random.randint(0, WIDTH), 0)
dynamiteSpeed = random.randint(10,15)
```

EXAMPLE 7.4 CONT.

SOURCE CODE CONT.

```
def update():
global gameState
if gameState == 'play':
            galagaMove()
            meteorMove()
            dynamiteMove()
            enemyMove()
            bulletMove()
            if score >= 5:
                  gameState = 'win’

def enemyMove():
global bugXdirection
global bugMoveDown
global gameState
global bulletActive
global score
if (len(enemies)>0) and (enemies[-1].x >= WIDTH or enemies[0].x <= 0
):
            bugXdirection = bugXdirection*-1#toggle direction
            bugMoveDown = True
for enemy in enemies:
            enemy.x = enemy.x + 5*bugXdirection
            if bugMoveDown == True:
                  enemy.y = enemy.y + 50
            if enemy.colliderect(galagaSprite):
                  gameState = 'gameover'

            for bullet in bullets :
                  if enemy.colliderect(bullet):
                        enemies.remove(enemy)
                        bullets.remove(bullet)
                        bulletActive = False
                        score = score + 1
                        print('score :', score)

bugMoveDown = False

def draw():
screen.blit('galagabg', (0,0))
def restart():
            global gameState
            global score
            global bulletActive
            global bugMoveDown
            screen.draw.text('Press s to play', center = (WIDTH/2,
500), color = 'white', fontsize = 70)
            if keyboard.s == True:
                  gameState = 'play'
                  score = 0
                  bulletActive = False
                  bugMoveDown = False
```

EXAMPLE 7.4 CONT.

SOURCE CODE CONT.

```
                  for row in range(0,2,1):
                        for column in range(0,5,1):
                              enemies.append(Actor('bug'))
                              enemies[-1].x = (100 +
100*column)
                              enemies[-1].y = (50 + 80*row)

      if gameState == 'start':
            screen.draw.text('Galaga', center = (WIDTH/2, 80),
color = 'white', fontsize = 100)
            restart()
      elif gameState == 'win':
            screen.draw.text('You Win', center = (WIDTH/2,
80), color = 'white', fontsize = 100)
            for enemy in enemies :
                  enemies.remove(enemy)
            restart()
      elif gameState == 'gameover':
            screen.draw.text('You Loose', center = (WIDTH/2,
80), color = 'white', fontsize = 100)
            for enemy in enemies :
                  enemies.remove(enemy)
            restart()
      elif gameState == 'play':
            galagaSprite.draw()
            meteorSprite.draw()
            dynamiteSprite.draw()
            for enemy in enemies:
                  enemy.draw()
             for bullet in bullets:
                  bullet.draw()

pgzrun.go()
```

EXAMPLE 7.5 (SONGS/MUSIC TO THE GAME)

FROM EXAMPLE 7.4, PERFORM THE FOLLOWING DETAILS

1. When you play the game
    - Play music 'run.mp3'
    - When bullet is shoot, play 'bullet_pew.wav'
2. If you lose the game or game over
    - Play sounds 'gameover.mp3'
3. if you win the game
    - Play music 'win.mp3'

* Create function for gameState to play music
* Note: music should not be called or being called in the function update(), as it will be recalled in every 1/60 time.

EXAMPLE 7.5 CONT.

SOURCE CODE

```
##To set screen to be in the middle
import os
os.environ['SDL_VIDEO_WINDOW_POS'] = "%d,%d" % (300,50)
import pygame
import pgzrun
import random

HEIGHT = 600
WIDTH = 750
TITLE = 'Game 7'

#actors
galagaSprite = Actor('galaga')
galagaSprite.pos = (WIDTH/2, 550)
meteorSprite = Actor('meteor')
meteorSprite.pos = (50, 50)
bugSprite = Actor('bug')
bugSprite.pos = (WIDTH/2,100)
dynamiteSprite = Actor('dynamite')
dynamiteSprite.pos = (600, 50)

#dynamic
meteorSpeed = 10
dynamiteSpeed = 10

#Group of enemy
enemies = []
bugXdirection = 1
bugMoveDown = False

#Bullets
bullets = []
bulletActive = False

#score
score = 0

#game state
gameState = 'start'

def on_key_down(key):
global bulletActive
if key == keys.SPACE and bulletActive == False:
            bullets.append(Actor('laserred'))
            bullets[-1].pos = galagaSprite.pos
            bulletActive = True
```

EXAMPLE 7.5 CONT.

SOURCE CODE CONT.

```
def bulletMove():
global bulletActive
for bullet in bullets:
            if bullet.y <= 0:
                  bullets.remove(bullet)
                  bulletActive = False
            else :
                  bullet.y = bullet.y - 10

def galagaMove():
global gameState
if keyboard.a == True :
            galagaSprite.x = galagaSprite.x - 10
elif keyboard.d == True :
            galagaSprite.x = galagaSprite.x + 10

if galagaSprite.x >= WIDTH :
            galagaSprite.x = WIDTH
if galagaSprite.x <= 0 :
            galagaSprite.x = 0

if galagaSprite.colliderect(dynamiteSprite):
            gameState = 'gameover'
            gameSongs()
            dynamiteReset()

elif galagaSprite.colliderect(meteorSprite):
            gameState = 'gameover'
            gameSongs()
            meteorReset()

def meteorMove():
meteorSprite.y = meteorSprite.y + meteorSpeed
if meteorSprite.y >= HEIGHT:
            meteorReset()

def meteorReset():
global meteorSpeed
meteorSprite.pos = (random.randint(0, WIDTH), 0)
meteorSpeed = random.randint(7,12)

def dynamiteMove():
dynamiteSprite.y = dynamiteSprite.y + dynamiteSpeed
if dynamiteSprite.y >= HEIGHT:
            dynamiteReset()

def dynamiteReset():
global dynamiteSpeed
dynamiteSprite.pos = (random.randint(0, WIDTH), 0)
dynamiteSpeed = random.randint(10,15)
```

EXAMPLE 7.5 CONT.

SOURCE CODE CONT.

```
def update():
global gameState
if gameState == 'play':
            galagaMove()
            meteorMove()
            dynamiteMove()
            enemyMove()
            bulletMove()
            if score >= 5:
                  gameState = 'win'
                  gameSongs()

def enemyMove():
global bugXdirection
global bugMoveDown
global gameState
global bulletActive
global score
if (len(enemies)>0) and (enemies[-1].x >= WIDTH or enemies[0].x <= 0
):
            bugXdirection = bugXdirection*-1#toggle direction
            bugMoveDown = True
for enemy in enemies:
            enemy.x = enemy.x + 5*bugXdirection
            if bugMoveDown == True:
                  enemy.y = enemy.y + 50
            if enemy.colliderect(galagaSprite):
                  gameState = 'gameover'
                  gameSongs()

            for bullet in bullets :
                  if enemy.colliderect(bullet):
                        enemies.remove(enemy)
                        bullets.remove(bullet)
                        bulletActive = False
                        score = score + 1
                        print('score :', score)
bugMoveDown = False

def draw():
screen.blit('galagabg', (0,0))
def restart():
            global gameState
            global score
            global bulletActive
            global bugMoveDown
            screen.draw.text('Press s to play', center = (WIDTH/2,
500), color = 'white', fontsize = 70)
            if keyboard.s == True:
                  gameState = 'play'
                  score = 0
                  bulletActive = False
                  bugMoveDown = False
```

EXAMPLE 7.5 CONT.

SOURCE CODE CONT.

```
                for row in range(0,2,1):
                      for column in range(0,5,1):
                            enemies.append(Actor('bug'))
                            enemies[-1].x = (100 +
100*column)
                            enemies[-1].y = (50 + 80*row)
                gameSongs()

if gameState == 'start':
          screen.draw.text('Galaga', center = (WIDTH/2, 80),
color = 'white', fontsize = 100)
          restart()
elif gameState == 'win':
          screen.draw.text('You Win', center = (WIDTH/2,
80), color = 'white', fontsize = 100)
          for enemy in enemies :
                enemies.remove(enemy)
          restart()
elif gameState == 'gameover':
          screen.draw.text('You Loose', center = (WIDTH/2,
80), color = 'white', fontsize = 100)
          for enemy in enemies :
                enemies.remove(enemy)
          restart()
elif gameState == 'play':
          galagaSprite.draw()
          meteorSprite.draw()
          dynamiteSprite.draw()
          for enemy in enemies:
                enemy.draw()
           for bullet in bullets:
                bullet.draw()

def gameSongs():
     if gameState == 'play':
          music.play('run')
     elif gameState == 'win':
          music.play('win')
     elif gameState == 'gameover':
          music.play('gameover')

pgzrun.go()
```

EXAMPLE 7.6 [EXTRA]
(ADD OTHER VARIABLE(S) TO THE GAME)

FROM EXAMPLE 7.5, PERFORM THE FOLLOWING DETAILS

1. You may choose to perform extra tasks from the following topics:
    - Add 3 lives to the Player, when life <=0 , game is over.
    - Add count up or count down timer for 10 s the game. If time is over, score must be >= 10 to win the game, if not, you lost the game.
    - Add 2 levels to the game:
        -> Easy : the speed of dynamite and meteor are random and slow
        -> Hard : the speed of dynamite and meteor are random and faster

END OF PYGAME#GAME7 !