```
import pygame
from pygame import mixer
import random

# Initialize pygame and mixer
pygame.mixer.pre_init(44100, -16, 2, 512)
mixer.init()
pygame.init()

# Define fps
clock = pygame.time.Clock()
fps = 60

# Set up the screen dimensions
screen_width = 600
screen_height = 800
screen = pygame.display.set_mode((screen_width, screen_height))
pygame.display.set_caption('Space Invaders')

# Define fonts
font30 = pygame.font.SysFont('Constantia', 30)
font40 = pygame.font.SysFont('Constantia', 40)

# Load sounds
explosion_fx = pygame.mixer.Sound("img_explosion.wav")  # Give Path
explosion_fx.set_volume(0.25)

explosion2_fx = pygame.mixer.Sound("C:/Users/eagle/Downloads/explosion2.wav")  # Give Path
```

```
explosion2_fx.set_volume(0.25)

laser_fx = pygame.mixer.Sound("laser.wav")  # Give Path
laser_fx.set_volume(0.25)

# Define game variables
rows = 5
cols = 5
alien_cooldown = 1000  # Alien bullet cooldown in milliseconds
last_alien_shot = pygame.time.get_ticks()
countdown = 3
last_count = pygame.time.get_ticks()
game_over = 0  # 0: no game over, 1: player has won, -1: player has lost

# Define colors
red = (255, 0, 0)
green = (0, 255, 0)
white = (255, 255, 255)

# Load background image
bg = pygame.image.load("bg.png")  # Give Path

def draw_bg():
    screen.blit(bg, (0, 0))

# Function for creating text
def draw_text(text, font, text_col, x, y):
    img = font.render(text, True, text_col)
    screen.blit(img, (x, y))

# Create spaceship class
```

```
class Spaceship(pygame.sprite.Sprite):
    def __init__(self, x, y, health):
        super().__init__()
        self.image = pygame.image.load("spaceship.png")  # Give Path
        self.rect = self.image.get_rect()
        self.rect.center = [x, y]
        self.health_start = health
        self.health_remaining = health
        self.last_shot = pygame.time.get_ticks()

    def update(self):
        global game_over
        if self.health_remaining <= 0:
            game_over = -1
            return

        speed = 8
        cooldown = 500  # milliseconds

        key = pygame.key.get_pressed()
        if key[pygame.K_LEFT] and self.rect.left > 0:
            self.rect.x -= speed
        if key[pygame.K_RIGHT] and self.rect.right < screen_width:
            self.rect.x += speed

        time_now = pygame.time.get_ticks()
        if key[pygame.K_SPACE] and time_now - self.last_shot > cooldown:
            laser_fx.play()
            bullet = Bullets(self.rect.centerx, self.rect.top)
            bullet_group.add(bullet)
            self.last_shot = time_now
```

```
        self.mask = pygame.mask.from_surface(self.image)


        pygame.draw.rect(screen, red, (self.rect.x, (self.rect.bottom + 10), self.rect.width, 15))
        if self.health_remaining > 0:
            pygame.draw.rect(screen, green, (self.rect.x, (self.rect.bottom + 10), int(self.rect.width * (self.health_remaining / self.health_start)), 15))


class Bullets(pygame.sprite.Sprite):
    def __init__(self, x, y):
        super().__init__()
        self.image = pygame.image.load("bullet.png")  # Give Path
        self.rect = self.image.get_rect()
        self.rect.center = [x, y]


    def update(self):
        self.rect.y -= 5
        if self.rect.bottom < 0:
            self.kill()
        if pygame.sprite.spritecollide(self, alien_group, True):
            self.kill()
            explosion_fx.play()
            explosion = Explosion(self.rect.centerx, self.rect.centery, 2)
            explosion_group.add(explosion)


class Aliens(pygame.sprite.Sprite):
    def __init__(self, x, y):
        super().__init__()
        self.image = pygame.image.load("C:/Users/eagle/Downloads/alien" + str(random.randint(1, 5)) + ".png")
        self.rect = self.image.get_rect()
        self.rect.center = [x, y]
```

```
        self.move_counter = 0
        self.move_direction = 1

    def update(self):
        self.rect.x += self.move_direction
        self.move_counter += 1
        if abs(self.move_counter) > 75:
            self.move_direction *= -1
            self.move_counter *= self.move_direction

class Alien_Bullets(pygame.sprite.Sprite):
    def __init__(self, x, y):
        super().__init__()
        self.image = pygame.image.load("alien_bullet.png")  # Give Path
        self.rect = self.image.get_rect()
        self.rect.center = [x, y]

    def update(self):
        global game_over
        self.rect.y += 2
        if self.rect.top > screen_height:
            self.kill()
        if pygame.sprite.spritecollide(self, spaceship_group, False, pygame.sprite.collide_mask):
            self.kill()
            explosion2_fx.play()
            spaceship.health_remaining -= 1
            explosion = Explosion(self.rect.centerx, self.rect.centery, 1)
            explosion_group.add(explosion)

class Explosion(pygame.sprite.Sprite):
    def __init__(self, x, y, size):
```

```
    super().__init__()
    self.images = []
    for num in range(1, 6):
        try:
            img = pygame.image.load(f"C:/Users/eagle/Downloads/exp{num}.png")
        except FileNotFoundError:
            print(f"Error: The file C:/Users/eagle/Downloads/exp{num}.png does not exist.")
            continue
        if size == 1:
            img = pygame.transform.scale(img, (20, 20))
        elif size == 2:
            img = pygame.transform.scale(img, (40, 40))
        elif size == 3:
            img = pygame.transform.scale(img, (160, 160))
        self.images.append(img)
    self.index = 0
    self.image = self.images[self.index]
    self.rect = self.image.get_rect()
    self.rect.center = [x, y]
    self.counter = 0

def update(self):
    explosion_speed = 3
    self.counter += 1
    if self.counter >= explosion_speed and self.index < len(self.images) - 1:
        self.counter = 0
        self.index += 1
        self.image = self.images[self.index]

    if self.index >= len(self.images) - 1 and self.counter >= explosion_speed:
        self.kill()
```

```
spaceship_group = pygame.sprite.Group()
bullet_group = pygame.sprite.Group()
alien_group = pygame.sprite.Group()
alien_bullet_group = pygame.sprite.Group()
explosion_group = pygame.sprite.Group()

def create_aliens():
    for row in range(rows):
        for item in range(cols):
            alien = Aliens(100 + item * 100, 100 + row * 70)
            alien_group.add(alien)

create_aliens()

spaceship = Spaceship(int(screen_width / 2), screen_height - 100, 3)
spaceship_group.add(spaceship)

def reset_game():
    global game_over, countdown, last_count
    spaceship.health_remaining = spaceship.health_start
    bullet_group.empty()
    alien_group.empty()
    alien_bullet_group.empty()
    explosion_group.empty()
    create_aliens()
    game_over = 0
    countdown = 3
    last_count = pygame.time.get_ticks()

run = True
```

```
while run:
    clock.tick(fps)
    draw_bg()

    if game_over == 0:
        if countdown == 0:
            time_now = pygame.time.get_ticks()
            if time_now - last_alien_shot > alien_cooldown and len(alien_bullet_group) < 5 and
len(alien_group) > 0:
                attacking_alien = random.choice(alien_group.sprites())
                alien_bullet = Alien_Bullets(attacking_alien.rect.centerx, attacking_alien.rect.bottom)
                alien_bullet_group.add(alien_bullet)
                last_alien_shot = time_now

            if len(alien_group) == 0:
                game_over = 1

            spaceship.update()
            bullet_group.update()
            alien_group.update()
            alien_bullet_group.update()
        else:
            draw_text('GET READY!', font40, white, int(screen_width / 2 - 110), int(screen_height / 2 +
50))
            draw_text(str(countdown), font40, white, int(screen_width / 2 - 10), int(screen_height / 2 +
100))
            count_timer = pygame.time.get_ticks()
            if count_timer - last_count > 1000:
                countdown -= 1
                last_count = count_timer

    elif game_over == -1:
```

```
        draw_text('GAME OVER!', font40, white, int(screen_width / 2 - 100), int(screen_height / 2))
        draw_text('Press R to Restart', font30, white, int(screen_width / 2 - 100), int(screen_height / 2 + 50))
    elif game_over == 1:
        draw_text('YOU WIN!', font40, white, int(screen_width / 2 - 100), int(screen_height / 2))
        draw_text('Press R to Restart', font30, white, int(screen_width / 2 - 100), int(screen_height / 2 + 50))

    explosion_group.update()
    spaceship_group.draw(screen)
    bullet_group.draw(screen)
    alien_group.draw(screen)
    alien_bullet_group.draw(screen)
    explosion_group.draw(screen)

    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            run = False
        if event.type == pygame.KEYDOWN:
            if event.key == pygame.K_r and game_over != 0:
                reset_game()

    pygame.display.update()

pygame.quit()
```