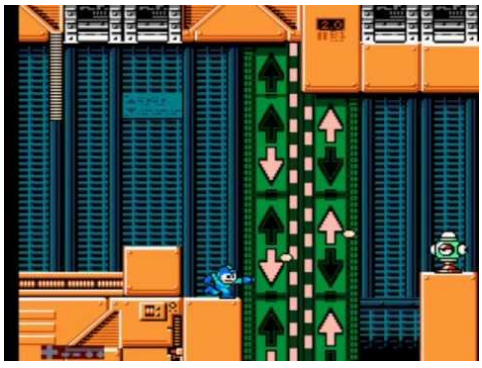
The Warrior

The Warrior เป็นเกมแนวผจญภัยโจมตีมอนสเตอร์และผ่านด่านเพิ่มความท้าทายเกมไปเรื่อย ๆ ที่มาของชื่อเกม The Warrior มาจากที่ตัวเอกเราจะรับบทเป็นนักรบในการปราบมอนสเตอร์และผ่าฟันอุปสรรคต่าง ๆ และบอสของเกมนี้เป็นมิโนทอร์ ตามตำนานกรีซ มิโนทอร์จะอยู่ในวงกตและธีสซูสเป็นนักรบผู้ที่ผ่าฟันอุปสรรคในวงกตและสามารถพิชิตมิโนทอร์ลงได้ จึงตั้งชื่อเกมว่า The Warrior ที่แปลว่า นักรบ เพราะอยากให้ผู้เล่นรับบทเป็นนักรบผู้กล้าหาญผ่าฟันอุปสรรคและเอาชนะเกมได้

ซึ่งเกมผมได้รับแรงบันดาลใจมาจากเกม Mario กับ Rockman ซึ่งจะเป็นเกมแนวผจญภัยที่ดังในยุค 90 แล้วผมก็เคยเล่นและเคยเห็นผ่านตามาบ้าง การเล่นไม่เข้าใจยากและสนุกมาก

เกม Mario



เกม Rockman

วิธีการเล่น

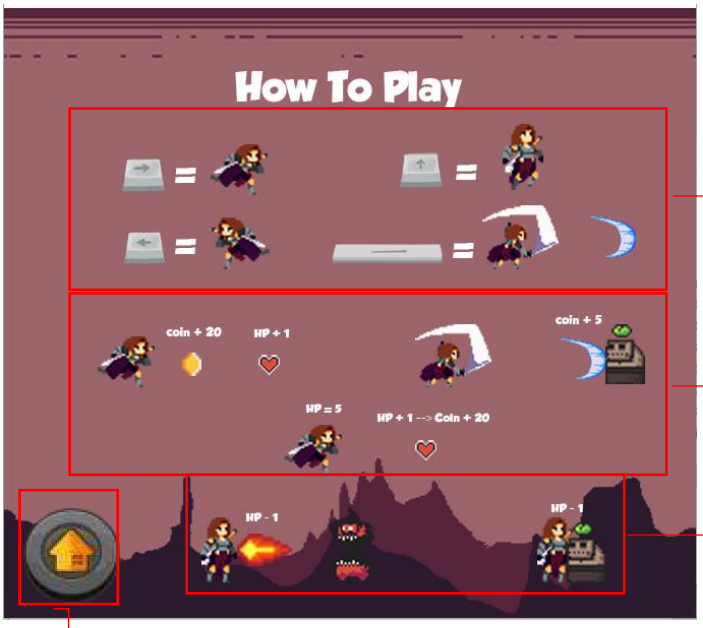
หน้าMenu

ปุ่มเริ่มเกม

ปุ่มออกเกม

ปุ่ม How To Play

หน้า How To Play



ปุ่มคำสั่งต่าง ๆ ใน

วิธีการเก็บคะแนน

HP ลด

ปุ่มกลับหน้าเมนู

หน้าเกม

เกมนี้จะจบเมื่อ

- หัวใจหมด เกมจะ Over
- ตกลาวา เกมจะ Over
- ฆ่าบอสเสร็จ เกมจะ Clear

การเก็บคะแนน จะขึ้นอยู่กับจำนวน Coin ที่เก็บได้โดยถ้า Score ปัจจุบันมากกว่า High Score จะทำการเขียนข้อมูล High Score ขึ้นมาใหม่

อธิบายโค้ด

EMPTY WINDOW

```python
import pygame
from pygame import mixer
import os
import random
import csv

mixer.init()
pygame.init()

#set หน้าเกม
WINDOW_WIDTH = 680
WINDOW_HEIGHT = 600
screen = pygame.display.set_mode((WINDOW_WIDTH, WINDOW_HEIGHT))
pygame.display.set_caption("The Warrior")
```

Set ค่าต่าง ๆ ที่ใช้

```python
#set FPS
clock = pygame.time.Clock()
FPS = 0

#set ค่าต่างๆ
GRAVITY = 0.75
TILE_SIZE = 40
ROWS = 15
COLS = 17
TILE_SIZE = WINDOW_HEIGHT // ROWS
TILE_TYPES = 10
MAX_LEVELS = 6
level = 1
start_game = False
score = 0
moving_left = False
moving_right = False
shoot = False
```

Load Images

```python
#โหลดรูป
img_list = []
for x in range(TILE_TYPES):
    img = pygame.image.load(f'img/tile/{x}.png')
    if x == 7 :
        img = pygame.transform.scale(img, (50, 80))
    else:
        img = pygame.transform.scale(img, (TILE_SIZE, TILE_SIZE))
    img_list.append(img)

#โหลดกระสุน
bullet_img = pygame.image.load('img/Fire/Fire.png').convert_alpha()
bullet_img = pygame.transform.scale(bullet_img, (59, 42))

#โหลดกระสุนศัตรู
bullet_demon = pygame.image.load('img/Fire/demon.png').convert_alpha()
bullet_demon = pygame.transform.scale(bullet_demon, (59, 42))

#โหลด Item
heal_box_img = pygame.image.load('img/Icons/heart.png').convert_alpha()
heal_box_img = pygame.transform.scale(heal_box_img, (30, 30))
heal_bar = pygame.image.load('img/Icons/heart_bar.png').convert_alpha()
heal_bar = pygame.transform.scale(heal_bar, (125, 25))
coin_box_img = pygame.image.load('img/Icons/coin.png').convert_alpha()
coin_box_img = pygame.transform.scale(coin_box_img, (20, 20))
item_boxes = {
    'Health' : heal_box_img,
    'Coin' : coin_box_img
}
#โหลดหน้าแรก
menu_img = pygame.image.load('img/GUI/main.png')
menu_img = pygame.transform.scale(menu_img, (680, 600))
over_img = pygame.image.load('img/GUI/Over.png')
over_img = pygame.transform.scale(over_img, (680, 600))
clear_img = pygame.image.load('img/GUI/Clear.png')
clear_img = pygame.transform.scale(clear_img, (680, 600))
how2play_img = pygame.image.load('img/GUI/how2play.png')
how2play_img = pygame.transform.scale(how2play_img, (680, 600))

#โหลดปุ่มกด
start_img = pygame.image.load('img/GUI/Start.png').convert_alpha()
exit_img = pygame.image.load('img/GUI/out.png').convert_alpha()
howplay_img = pygame.image.load('img/GUI/how.png').convert_alpha()
home_img = pygame.image.load('img/GUI/home.png').convert_alpha()

#พื้นหลัง
bg_img_1 = pygame.image.load('img/background/parallax-mountain-bg.png')
bg_img_1 = pygame.transform.scale(bg_img_1, (1000, 600))
bg_img_2 = pygame.image.load('img/background/parallax-mountain-foreground-trees.png')
bg_img_2 = pygame.transform.scale(bg_img_2, (1000, 600))
bg_img_3 = pygame.image.load('img/background/parallax-mountain-montain-far.png')
bg_img_3 = pygame.transform.scale(bg_img_3, (1000, 600))
bg_img_4 = pygame.image.load('img/background/parallax-mountain-mountains.png')
bg_img_4 = pygame.transform.scale(bg_img_4, (1000, 600))
bg_img_5 = pygame.image.load('img/background/parallax-mountain-trees.png')
bg_img_5 = pygame.transform.scale(bg_img_5, (1000, 600))
```

Sounds and Music

```python
#โหลดเพลง
pygame.mixer.music.load('sound/Sound_BG.wav')
pygame.mixer.music.set_volume(0.3)
pygame.mixer.music.play(-1, 0.0, 5000)
coin_fx = pygame.mixer.Sound('sound/coin.wav')
coin_fx.set_volume(2)
health_fx = pygame.mixer.Sound('sound/alive.wav')
health_fx.set_volume(2)
hit_fx = pygame.mixer.Sound('sound/hit.wav')
hit_fx.set_volume(0.3)
completed_fx = pygame.mixer.Sound('sound/completed.wav')
completed_fx.set_volume(0.3)
over_fx = pygame.mixer.Sound('sound/game_over.wav')
over_fx.set_volume(0.3)
fire_fx = pygame.mixer.Sound('sound/fire.wav')
fire_fx.set_volume(0.3)
hit_player_fx = pygame.mixer.Sound('sound/hit_player.wav')
hit_player_fx.set_volume(0.3)

```

Load Font

```python
#โหลด font
font_score = pygame.font.Font('GROBOLD.ttf', 50)
font = pygame.font.Font('Noah-Bold.otf', 20)
```

ฟังก์ชันโชว์ Score บนเกม

```python
#Score
def draw_text(text, font, text_col, x, y):
    coin_img = pygame.image.load('img/Icons/coin.png').convert_alpha()
    coin_img = pygame.transform.scale(coin_img, (20, 20))
    img = font.render(text, True, text_col)
    screen.blit(coin_img, (x, y-2))
    screen.blit(img, (x + 20, y-5))
```

สร้างไฟล์ txt เก็บ High Score

```python
#High Score
if os.path.exists('score.txt'):
    with open('score.txt', 'r') as file:
        high_score = int(file.read())
else:
    high_score = 0
```

ฟังก์ชัน Reset ด่าน

```python
#Resetด่าน
def reset_level():
    bullet_group.empty()
    item_box_group.empty()
    zombie_group.empty()
    demon_group.empty()
    boss_group.empty()
    exit_group.empty()
    rava_group.empty()
    property_group.empty()

    data = []
    for row in range(ROWS):
        r = [-1] * COLS
        data.append(r)
    return data
```

Class Player

```
class Player(pygame.sprite.Sprite):
    def __init__(self, x, y, speed):
        pygame.sprite.Sprite.__init__(self)
        self.alive = True
        self.speed = speed
        self.shoot_cooldown = 0
        self.health = 5
        self.max_health = self.health
        self.coin = 0
        self.direction = 1
        self.vel_y = 0
        self.jump = False
        self.in_air = True
        self.flip = False
        self.animation_list = []
        self.frame_index = 0
        self.action = 0
        self.update_time = pygame.time.get_ticks()
        self.immortal = False
        self.complete = False

        #โหลดรูปต่างๆ
        animation_types = ['Idle', 'run', 'Jump', 'Attack', 'Death']
        for animation in animation_types:
            temp_list = []
            num_of_frames = len(os.listdir(f'img/player/{animation}'))
            for num in range(num_of_frames):
                img = pygame.image.load(f'img/player/{animation}/{num}.png').convert_alpha()
                img = pygame.transform.scale(img, (120, 110))
                temp_list.append(img)
            self.animation_list.append(temp_list)

        self.image = self.animation_list[self.action][self.frame_index]
        self.rect = self.image.get_rect()
        self.rect.center = (x, y)
        self.rect.x = x
        self.rect.y = y
        self.width = self.image.get_width()
        self.height = self.image.get_height()
        self.hitbox = (self.rect.x + 45, self.rect.y +35, 20, 75)
```

การเคลื่อนไหวของ Class Player

```
    #update ตัวละคร
    def update(self):
        self.update_animation()
        self.check_alive()
        if self.shoot_cooldown > 0 :
            self.shoot_cooldown -= 1

    def move(self, moving_left, moving_right):
            dx = 0
            dy = 0
            #ซ้าย/ขวา
            if moving_left:
                dx = -self.speed
                self.flip = True
                self.direction = -1
            if moving_right:
                dx = self.speed
                self.flip = False
                self.direction = 1
            #กระโดด
            if self.jump == True and self.in_air == False :
                self.vel_y = -12
                self.jump = False
                self.in_air = True

            self.vel_y += GRAVITY
            if self.vel_y > 10:
                self.vel_y
            dy += self.vel_y
            #ที่ชนล็อค
            for tile in world.obstacle_list:
                if tile[1].colliderect(self.rect.x + 45 + dx , self.rect.y +35, 20, 75):
                    dx = 0
                if tile[1].colliderect(self.rect.x + 45, self.rect.y +35 + dy, 20, 75):
                    if self.vel_y < 0:
                        self.vel_y = 0
                        dy = tile[1].bottom - self.rect.top
                    elif self.vel_y >= 0:
                        self.vel_y = 0
                        self.in_air = False
                        dy = tile[1].top - self.rect.bottom
```

ตรวจสอบสถานะของ Class Player

```
            #ตกลาวา
            for rava in rava_group:
                if rava.rect.colliderect(self.hitbox) :
                    self.health = 0

            #เปลี่ยนตำแหน่ง
            self.rect.x += dx
            self.rect.y += dy
            self.hitbox = (self.rect.x + 45, self.rect.y +35, 20, 75)

            #เปลี่ยนด่าน
            level_complete = False
            for next in exit_group:
                if next.rect.colliderect(self.hitbox) :
                    level_complete = True
                return level_complete

            #ชนะ
            for property in property_group:
                if property.rect.colliderect(self.hitbox) and clear_num == 1:
                    self.complete = True
                    completed_fx.play()
```

ฟังก์ชันการโจมตีของ Class Player

```
    #ยิง
    def shoot(self):
        if self.shoot_cooldown == 0 :
            hit_fx.play()
            self.shoot_cooldown = 20
            bullet = Bullet(self.rect.centerx + 25 * self.direction, self.rect.centery + 5, self.direction, 1)
            bullet_group.add(bullet)
```

Animation ของ Class Player

```
    #เปลี่ยนรูป
    def update_action(self, new_action):
        if new_action != self.action :
            self.action = new_action
            self.frame_index = 0
            self.update_time = pygame.time.get_ticks()

```

เช็คสถานะ HP ของ Class Player

```
    #เช็คเลือด
    def check_alive(self):
        if self.health <= 0 :
            self.health = 0
            self.speed = 0
            self.alive = False
            self.update_action(4)

```

ฟังก์ชันตำแหน่ง และ Draw Class Player

```
    #set ตำแหน่ง
    def setlocation(self, x, y):
        self.rect.center = (x, y)
        self.rect.x = x
        self.rect.y = y
        self.hitbox = (self.rect.x + 45, self.rect.y +35, 20, 75)

    def draw(self):
        screen.blit(pygame.transform.flip(self.image, self.flip, False), self.rect)
        # pygame.draw.rect(screen, (255,255,255), self.hitbox, 2)
```

Class Enemy

```
class Enemy(pygame.sprite.Sprite):
    def __init__(self, char_type, x, y, speed):
        pygame.sprite.Sprite.__init__(self)
        self.alive = True
        self.char_type = char_type
        self.speed = speed
        self.shoot_cooldown = 0
        if self.char_type == 'zombie' :
            self.health = 20
        if self.char_type == 'demon' :
            self.health = 10
        if self.char_type == 'boss' :
            self.health = 100
        self.direction = 1
        self.flip = False
        self.animation_list = []
        self.frame_index = 0
        self.action = 0
        self.update_time = pygame.time.get_ticks()
        self.attack_time = 0

        #โหลดรูปต่างๆ
        animation_types = ['Idle', 'run', 'Death', 'Attack']
        for animation in animation_types:
            temp_list = []
            num_of_frames = len(os.listdir(f'img/enemy/{self.char_type}/{animation}'))
            for num in range(num_of_frames):
                img = pygame.image.load(f'img/enemy/{self.char_type}/{animation}/{num}.png').convert_alpha()

                if self.char_type == 'boss' :
                    img = pygame.transform.scale(img, (150, 150))
                    temp_list.append(img)

                elif animation == 'Death' and self.char_type != 'boss' :
                    img = pygame.transform.scale(img, (80, 80))
                    temp_list.append(img)
                elif self.char_type != 'boss':
                    img = pygame.transform.scale(img, (80, 80))
                    temp_list.append(img)
            self.animation_list.append(temp_list)

        self.image = self.animation_list[self.action][self.frame_index]
        self.rect = self.image.get_rect()
        self.rect.center = (x, y)
        self.rect.x = x
        self.rect.y = y
        self.move_counter = 0
        self.idling = False
        self.idling_counter = 0
        self.vision = pygame.Rect(0, 0, 350, 20)
        if self.char_type == 'boss' :
            self.hitbox = (self.rect.x + 40, self.rect.y +35, 60, 100)
        elif self.char_type != 'boss' :
            self.hitbox = (self.rect.x + 25, self.rect.y +10, 40, 70)
```

การเคลื่อนไหวของ Class Enemy

```python
    #update ตัวละคร
    def update(self):
        self.update_animation()
        self.check_alive()
        if self.shoot_cooldown > 0 :
            self.shoot_cooldown -= 1

    def move(self, moving_left, moving_right):
        dx = 0
        dy = 0
        #วิ่งซ้าย/ขวา
        if moving_left:
            dx = -self.speed
            self.flip = True
            self.direction = -1
        if moving_right:
            dx = self.speed
            self.flip = False
            self.direction = 1

        #เปลี่ยนตำแหน่ง
        self.rect.x += dx
        self.rect.y += dy
        if self.char_type == 'boss' :
            self.hitbox = (self.rect.x + 40, self.rect.y +35, 60, 100)
        elif self.char_type != 'boss' :
            self.hitbox = (self.rect.x + 25, self.rect.y +10, 40, 70)
    #ศัตรูเคลื่อนไหว
    def ai(self):
        #Zombie
        if self.alive and self.char_type == 'zombie' :
            if self.direction == 1:
                ai_moving_right = True
            else:
                ai_moving_right = False
            ai_moving_left = not ai_moving_right
            self.move(ai_moving_left, ai_moving_right)
            self.update_action(1)
            self.move_counter += 1

            if self.move_counter > 10 :
                self.direction *= -1
                self.move_counter *= -1
        #Demon
        if self.alive and self.char_type == 'demon' :
            self.vision.center = (self.rect.centerx + 150 * self.direction, self.rect.centery)
            if self.vision.colliderect(player.hitbox) and player.alive :
                self.shoot()
            else:
                if random.randint(1, 20) == 1:
                    self.direction = -1
                    self.flip = True
                elif random.randint(1, 20) == 5:
                    self.direction = 1
                    self.flip = False
        #Boss
        if self.alive and self.char_type == 'boss'  :
            if self.idling == False and random.randint(1, 200) == 1:
                self.update_action(0)
                self.idling = True
                self.idling_counter = 50
            if self.vision.colliderect(player.hitbox) and player.alive :
                self.update_action(3)
                self.shoot()
            else:
                if self.idling == False:
                    if self.direction == 1:
                        ai_moving_right = True
                    else:
                        ai_moving_right = False
                    ai_moving_left = not ai_moving_right
                    self.move(ai_moving_left, ai_moving_right)
                    self.update_action(1)
                    self.move_counter += 1
                    self.vision.center = (self.rect.centerx + 150 * self.direction, self.rect.centery)
                    # pygame.draw.rect(screen,(255,255,255),self.vision)

                    if self.move_counter > 10:
                        self.direction *= -1
                        self.move_counter *= -1
                else:
                    self.idling_counter -= 1
                    if self.idling_counter <= 0:
                        self.idling = False
```

ฟังก์ชันการโจมตีของ Class Enemy

```python
    #ยิง
    def shoot(self):
        if self.shoot_cooldown == 0 :
            fire_fx.play()
            self.shoot_cooldown = 30
            bullet = Bullet(self.rect.centerx + 25 * self.direction, self.rect.centery + 5, self.direction, 2)
            bullet_group.add(bullet)
```

Animation และ เช็คสถานะ HP ของ Class Enemy

```python
    #รูปเคลื่อนไหว
    def update_animation(self):
        ANIMATION_COOLDOWN = 100
        self.image = self.animation_list[self.action][self.frame_index]
        if pygame.time.get_ticks() - self.update_time > ANIMATION_COOLDOWN:
            self.update_time = pygame.time.get_ticks()
            self.frame_index += 1
        if self.frame_index >= len(self.animation_list[self.action]):
            if self.action == 2 :
                self.frame_index = len(self.animation_list[self.action]) - 1
            else:
                self.frame_index = 0

    #เปลี่ยนรูป
    def update_action(self, new_action):
        if new_action != self.action :
            self.action = new_action
            self.frame_index = 0
            self.update_time = pygame.time.get_ticks()

    #เช็คชีวิต
    def check_alive(self):
        if self.health <= 0 :
            self.health = 0
            self.speed = 0
            self.alive = False
            self.update_action(2)
```

ฟังก์ชันโจมตี Player และ Draw Class Enemy

```python
    #ชนผู้เล่น
    def attack(self):
        if self.alive and player.alive :
                if self.rect.colliderect(player.hitbox):
                    self.attack_time += 0.2
                    if player.immortal == False:
                        hit_player_fx.play()
                        player.health -= 1
                    if int(self.attack_time) == 3 :
                        player.immortal = False
                        self.attack_time = 0
                    else:
                        player.immortal = True

    def draw(self):
        screen.blit(pygame.transform.flip(self.image, self.flip, False), self.rect)
        # pygame.draw.rect(screen, (255,255,255), self.hitbox, 2)
```

ฟังก์ชันสร้าง Coin กับ HP

```python
#สร้างเลือดกับเหรียญ
class ItemBox(pygame.sprite.Sprite):
    def __init__(self, item_type, x, y):
        pygame.sprite.Sprite.__init__(self)
        self.item_type = item_type
        self.image = item_boxes[self.item_type]
        self.rect = self.image.get_rect()
        self.rect.midtop = (x + TILE_SIZE // 2, y + (TILE_SIZE - self.image.get_height()))

    def update(self):
        if self.rect.colliderect(player.hitbox):
            if self.item_type == 'Health':
                health_fx.play()
                player.health += 1
                if player.health > player.max_health:
                    player.health = player.max_health
                    player.coin += 20
            elif self.item_type == 'Coin':
                coin_fx.play()
                player.coin += 20
            self.kill()
```

ฟังก์ชันสร้างประตูเปลี่ยนด่าน

```python
#สร้างประตู
class Exit(pygame.sprite.Sprite):
    def __init__(self, img, x, y):
        pygame.sprite.Sprite.__init__(self)
        self.image = img
        self.rect = self.image.get_rect()
        self.rect.midtop = (x + TILE_SIZE // 2, y + (TILE_SIZE - self.image.get_height()))
```

ฟังก์ชันสร้างลาวา

```python
#สร้างลาวา
class Rava(pygame.sprite.Sprite):
    def __init__(self, img, x, y):
        pygame.sprite.Sprite.__init__(self)
        self.image = img
        self.rect = self.image.get_rect()
        self.rect.midtop = (x + TILE_SIZE // 2, y + (TILE_SIZE - self.image.get_height()))
```

ฟังก์ชันสร้างสมบัติ

```python
#สร้างสมบัติ
class Property(pygame.sprite.Sprite):
    def __init__(self, img, x, y):
        pygame.sprite.Sprite.__init__(self)
        self.image = img
        self.rect = self.image.get_rect()
        self.rect.midtop = (x + TILE_SIZE // 2, y + (TILE_SIZE - self.image.get_height()))
```

ฟังก์ชันสร้างกระสุน

```python
#สร้างกระสุน
class Bullet(pygame.sprite.Sprite):
    def __init__(self, x, y, direction, types):
        pygame.sprite.Sprite.__init__(self)
        self.types = types
        if self.types == 1 :
            type_img = bullet_img
            self.speed = 20
        elif self.types == 2 :
            type_img = bullet_demon
            self.speed = 10
        if direction == 1 :
            self.image = type_img
        elif direction == -1 :
            self.image = pygame.transform.flip(type_img, True, False)
        self.rect = self.image.get_rect()
        self.rect.center = (x, y)
        self.direction = direction

    def update(self):
        self.rect.x += (self.direction * self.speed)
        if self.rect.right < 0 or self.rect.left > WINDOW_WIDTH:
            self.kill()

        #กระสุนโดนกำแพง
        for tile in world.obstacle_list:
            if tile[1].colliderect(self.rect):
                self.kill()
        #playerโดนกระสุน
        if self.rect.colliderect(player.hitbox) and self.types == 2 :
            if player.alive:
                player.health -= 1
                self.kill()

        #zombieโดนกระสุน
        for zombie in zombie_group:
            if pygame.sprite.spritecollide(zombie, bullet_group, False):
                if self.rect.colliderect(zombie.hitbox) and self.types == 1:
                    if zombie.alive:
                        zombie.health -= 5
                        player.coin += 5
                        self.kill()

        #demonโดนกระสุน
        for demon in demon_group:
            if pygame.sprite.spritecollide(demon, bullet_group, False):
                if self.rect.colliderect(demon.hitbox) and self.types == 1 :
                    if demon.alive:
                        demon.health -= 5
                        player.coin += 5
                        self.kill()

        #bossโดนกระสุน
        for boss in boss_group:
            if pygame.sprite.spritecollide(boss, bullet_group, False):
                if self.rect.colliderect(boss.hitbox) and self.types == 1 :
                    if boss.alive:
                        boss.health -= 5
                        player.coin += 5
                        self.kill()
```

โค้ดหน้า Menu

```python
#สร้างแมพ
class World():
    def __init__(self):
        self.obstacle_list = []

    def process_data(self, data, player_data = None):
        for y, row in enumerate(data):
            for x, tile in enumerate(row):
                if tile >= 0 :
                    img = img_list[tile]
                    img_rect = img.get_rect()
                    img_rect.x = x * TILE_SIZE
                    img_rect.y = y * TILE_SIZE
                    tile_data = (img, img_rect)
                    if tile == 0 :
                        self.obstacle_list.append(tile_data)
                    elif tile == 1 :
                        if player_data == None :
                            player = Player(x * TILE_SIZE, y * TILE_SIZE,15)
                        else:
                            player = player_data
                            player.setlocation(x * TILE_SIZE, y * TILE_SIZE)
                    elif tile == 2 :
                        zombie = Enemy('zombie',x * TILE_SIZE, y * TILE_SIZE - 40 ,3)
                        zombie_group.add(zombie)
                    elif tile == 3 :
                        demon = Enemy('demon',x * TILE_SIZE, y * TILE_SIZE -40 ,15)
                        demon_group.add(demon)
                    elif tile == 4 :
                        boss = Enemy('boss',x * TILE_SIZE, y * TILE_SIZE - 90,3)
                        boss_group.add(boss)
                    elif tile == 5 :
                        item_box = ItemBox('Coin', x * TILE_SIZE, y * TILE_SIZE)
                        item_box_group.add(item_box)
                    elif tile == 6 :
                        item_box = ItemBox('Health', x * TILE_SIZE, y * TILE_SIZE)
                        item_box_group.add(item_box)
                    elif tile == 7 :
                        exit = Exit(img, x * TILE_SIZE, y * TILE_SIZE)
                        exit_group.add(exit)
                    elif tile == 8 :
                        rava = Rava(img, x * TILE_SIZE, y * TILE_SIZE)
                        rava_group.add(rava)
                    elif tile == 9 :
                        property = Property(img, x * TILE_SIZE, y * TILE_SIZE)
                        property_group.add(property)
        return player

    def draw(self):
        for tile in self.obstacle_list:
            screen.blit(tile[0], tile[1])
```

ฟังก์ชันสร้างปุ่ม Button

```python
#สร้างปุ่ม
class Button():
    def __init__(self,x, y, image, scale):
        width = image.get_width()
        height = image.get_height()
        self.image = pygame.transform.scale(image, (int(width * scale), int(height * scale)))
        self.rect = self.image.get_rect()
        self.rect.topleft = (x, y)
        self.clicked = False

    def draw(self, screen):
        action = False
        pos = pygame.mouse.get_pos()

        #เช็คว่ากดไหม
        if self.rect.collidepoint(pos):
            if pygame.mouse.get_pressed()[0] == 1 and self.clicked == False:
                action = True
                self.clicked = True
        if pygame.mouse.get_pressed()[0] == 0:
            self.clicked = False

        #วาดปุ่ม
        screen.blit(self.image, (self.rect.x, self.rect.y))
        return action
#ทำปุ่มกด
start_button = Button(155, 320, start_img, 2)
howplay_button = Button(280, 320, howplay_img, 2)
exit_button = Button(401, 320, exit_img, 2)
home_button = Button(10, 480, home_img, 2)
```

ฟังก์ชันรวมกลุ่มของแต่ละฟังก์ชัน

```python
#group
bullet_group = pygame.sprite.Group()
item_box_group = pygame.sprite.Group()
zombie_group = pygame.sprite.Group()
demon_group = pygame.sprite.Group()
boss_group = pygame.sprite.Group()
exit_group = pygame.sprite.Group()
rava_group = pygame.sprite.Group()
property_group = pygame.sprite.Group()
```

โค้ดอ่านข้อมูลจากไฟล์อื่น

```python
#อ่านแมพจากไฟล์อื่น
world_data = []
for row in range(ROWS):
    r = [-1] * COLS
    world_data.append(r)
with open(f'level{level}_data.csv', newline='') as csvfile:
    reader = csv.reader(csvfile, delimiter=',')
    for x, row in enumerate(reader):
        for y, tile in enumerate(row):
            world_data[x][y] = int(tile)
world = World()
player = world.process_data(world_data)
```

โค้ดหน้า Menu

```python
run = True
while run:
        clock.tick(FPS)
        #หน้าเมนู
        if start_game == False :
            screen.blit(menu_img,(0, 0))
            #add high score
            high_score_surface = font.render(f'High Score : {high_score}',True,(255,255,255))
            high_score_rect = high_score_surface.get_rect()
            high_score_rect.center = (330, 300)
            screen.blit(high_score_surface, high_score_rect)
            if start_button.draw(screen) :
                start_game = True
            if exit_button.draw(screen) :
                run = False
            if howplay_button.draw(screen):
                how2play = True
```

โค้ดหลังกดเริ่มเกม ให้สร้างรูปต่าง ๆ และ Player

```python
        else:
            #วาดพื้นหลัง
            screen.blit(bg_img_1,(-100, 0))
            screen.blit(bg_img_2,(-100, 0))
            screen.blit(bg_img_3,(-100, 0))
            screen.blit(bg_img_4,(-100, 0))
            screen.blit(bg_img_5,(-100, 0))
            screen.blit(heal_bar, (52,12))

            score += player.coin
            player.coin = 0
            draw_text(f': {score}', font, (255,255,255), 50, 50) #วาด score
            for x in range(player.health):
                screen.blit(heal_box_img , (50 + (25 * x) , 10)) #วาด health

            player.draw()
            bullet_group.draw(screen)
            item_box_group.draw(screen)
            exit_group.draw(screen)
            rava_group.draw(screen)
            world.draw()
            player.update()
            bullet_group.update()
            item_box_group.update()
            exit_group.update()
            rava_group.update()
```

โค้ดสร้าง Enemy

```
        for zombie in zombie_group:
            zombie.ai()
            zombie.update()
            zombie.draw()
            zombie.attack()

        for demon in demon_group:
            demon.ai()
            demon.update()
            demon.draw()
            demon.attack()

        for boss in boss_group:
            boss.ai()
            boss.update()
            boss.draw()
            boss.attack()
            #ผ่านด่าน
            if boss.health <= 0 :
                property_group.draw(screen)
                property_group.update()
                clear_num = 1
```

โค้ดเปลี่ยน Animation ของ Player และ เปลี่ยนด่าน

```
        #player มีชีวิต
        if player.alive:
            if shoot :
                player.update_action(3) # 3:ยิง
                player.shoot()
            elif moving_left or moving_right :
                player.update_action(1) # 1:วิ่ง
            elif player.in_air :
                player.update_action(2) # 2:กระโดด
            else:
                player.update_action(0) # 0:ยืน
            level_complete = player.move(moving_left, moving_right)
            #ตรวจว่าจบประตู
            if level_complete :
                level += 1
                world_data = reset_level()
                if level <= MAX_LEVELS :
                    with open(f'level{level}_data.csv', newline='') as csvfile:
                        reader = csv.reader(csvfile, delimiter=',')
                        for x, row in enumerate(reader):
                            for y, tile in enumerate(row):
                                world_data[x][y] = int(tile)
                    world = World()
                    player = world.process_data(world_data, player)
```

โค้ดตรวจสอบ Game Clear

```
            #ตรวจชนะ
            if player.complete :
                screen.blit(clear_img,(0, 0))
                #แสดง score
                score_surface = font_score.render(f'Score : {score}',True,(255,255,255))
                score_rect = score_surface.get_rect()
                score_rect.center = (320, 350)
                screen.blit(score_surface, score_rect)
                #แสดง high score
                high_score_surface = font_score.render(f'High Score : {high_score}',True,(255,255,255))
                high_score_rect = high_score_surface.get_rect()
                high_score_rect.center = (320, 430)
                screen.blit(high_score_surface, high_score_rect)
                #บันทึก high score
                if score > high_score:
                    high_score = score
                    with open('score.txt', 'w') as file:
                        file.write(str(high_score))
                #ปุ่มhome
                if home_button.draw(screen) :
                    start_game = False
                    world_data = reset_level()
                    level = 1
                    score = 0
                    with open(f'level{level}_data.csv', newline='') as csvfile:
                        reader = csv.reader(csvfile, delimiter=',')
                        for x, row in enumerate(reader):
                            for y, tile in enumerate(row):
                                world_data[x][y] = int(tile)
                    world = World()
                    player = world.process_data(world_data)
```

โค้ดตรวจสอบ Game Over

```
        #player ไม่มีชีวิต
        else:
            over_fx.play()
            screen.blit(over_img,(0, 0))
            #แสดง score
            score_surface = font_score.render(f'Score : {score}',True,(255,255,255))
            score_rect = score_surface.get_rect()
            score_rect.center = (320, 350)
            screen.blit(score_surface, score_rect)
            #แสดง high score
            high_score_surface = font_score.render(f'High Score : {high_score}',True,(255,255,255))
            high_score_rect = high_score_surface.get_rect()
            high_score_rect.center = (320, 410)
            screen.blit(high_score_surface, high_score_rect)
            #บันทึก high score
            if score > high_score:
                high_score = score
                with open('score.txt', 'w') as file:
                    file.write(str(high_score))
            #ปุ่มhome
            if home_button.draw(screen) :
                start_game = False
                world_data = reset_level()
                level = 1
                score = 0
                with open(f'level{level}_data.csv', newline='') as csvfile:
                    reader = csv.reader(csvfile, delimiter=',')
                    for x, row in enumerate(reader):
                        for y, tile in enumerate(row):
                            world_data[x][y] = int(tile)
                world = World()
```

โค้ดสร้างหน้า How To Play

```
        #หน้าคู่มือ
        if how2play :
            screen.blit(how2play_img,(0, 0))
            if home_button.draw(screen) :
                how2play = False
                world_data = reset_level()
                level = 1
                score = 0
                with open(f'level{level}_data.csv', newline='') as csvfile:
                    reader = csv.reader(csvfile, delimiter=',')
                    for x, row in enumerate(reader):
                        for y, tile in enumerate(row):
                            world_data[x][y] = int(tile)
                world = World()
                player = world.process_data(world_data)
```

Event Loop

```

        for event in pygame.event.get():
            if event.type == pygame.QUIT:
                run = False

```

Keyboard Events & Quit Event

```
            #กดคีย์บอร์ด
            if event.type == pygame.KEYDOWN:
                if event.key == pygame.K_LEFT:
                    moving_left = True
                if event.key == pygame.K_RIGHT:
                    moving_right = True
                if event.key == pygame.K_SPACE:
                    shoot = True
                if event.key == pygame.K_UP and player.alive :
                    player.jump = True
                if event.key == pygame.K_ESCAPE:
                    run = False

            #ปล่อยคีย์บอร์ด
            if event.type == pygame.KEYUP:
                if event.key == pygame.K_LEFT:
                    moving_left = False
                if event.key == pygame.K_RIGHT:
                    moving_right = False
                if event.key == pygame.K_SPACE:
                    shoot = False

        pygame.display.update()

    pygame.quit()
```