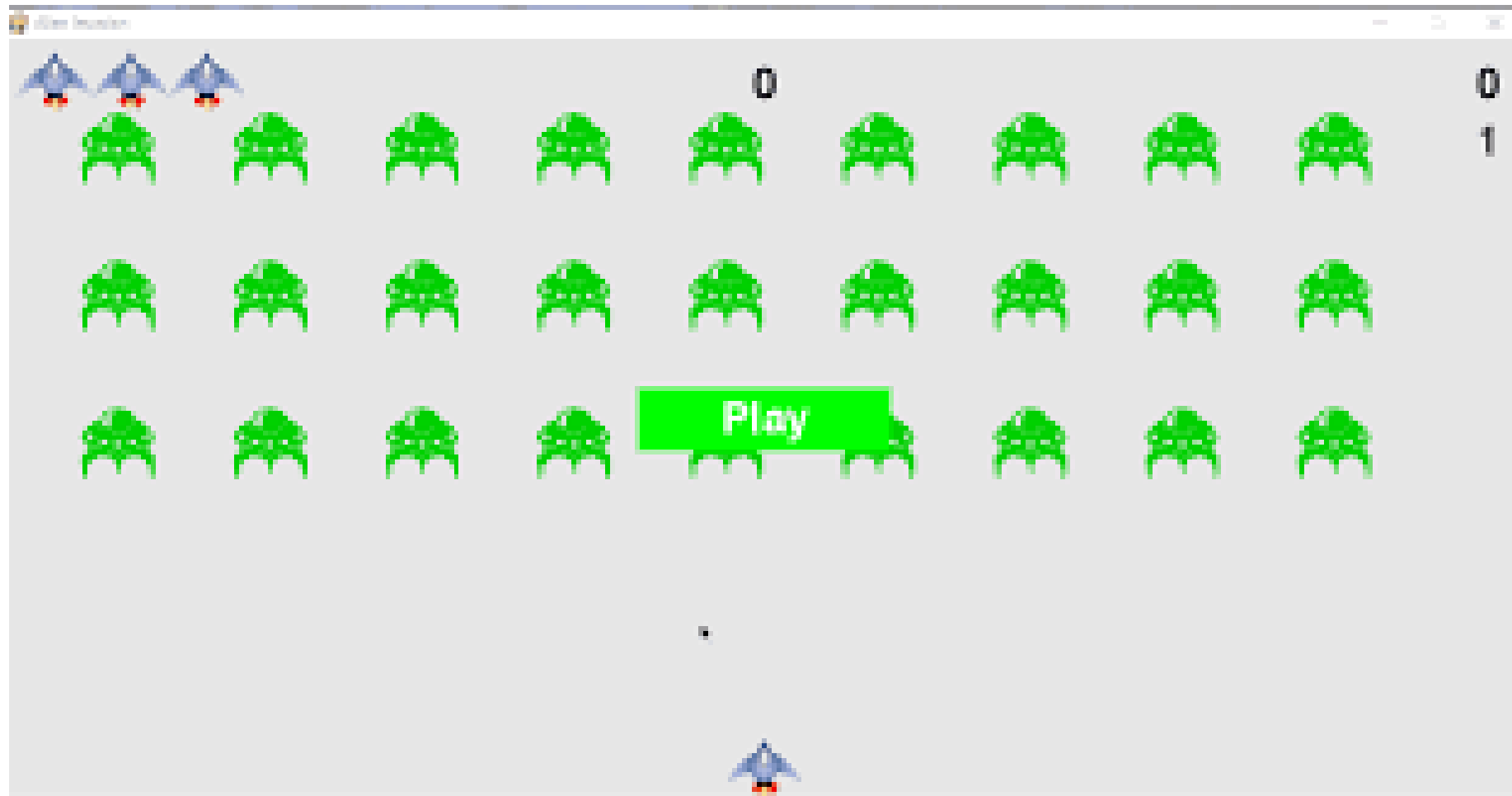


# Alien Invasion

(updated) Project

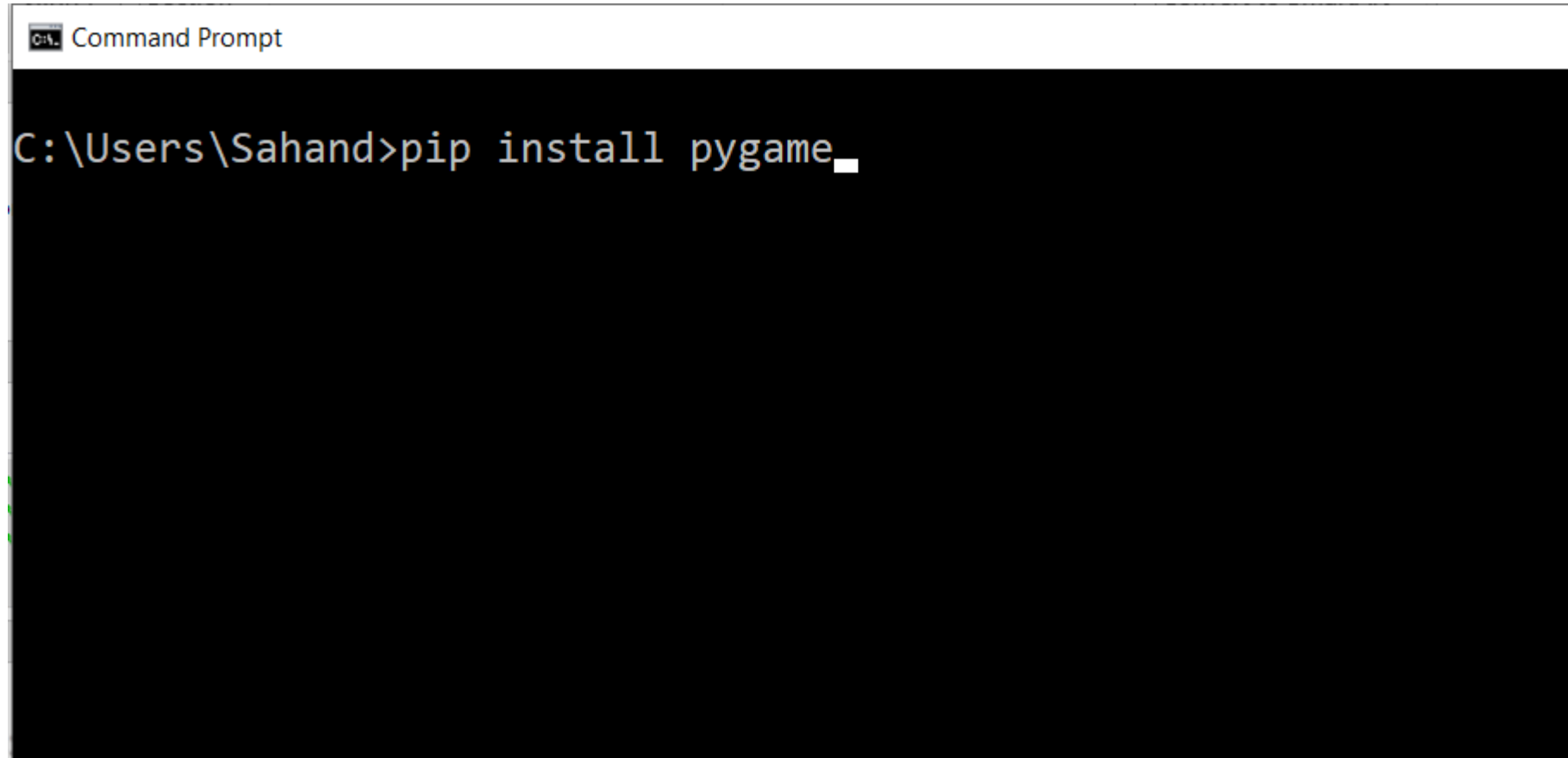
# Introduction



# Introduction

In Alien Invasion, the player controls a ship that appears at the bottom center of the screen. The player can move the ship right and left using the arrow keys and shoot bullets using the spacebar. When the game begins, a fleet of aliens fills the sky and moves across and down the screen. The player shoots and destroys the aliens. If the player shoots all the aliens, a new fleet appears that moves faster than the previous fleet. If any alien hits the player's ship or reaches the bottom of the screen, the player loses a ship. If the player loses three ships, the game ends.

# Preparation

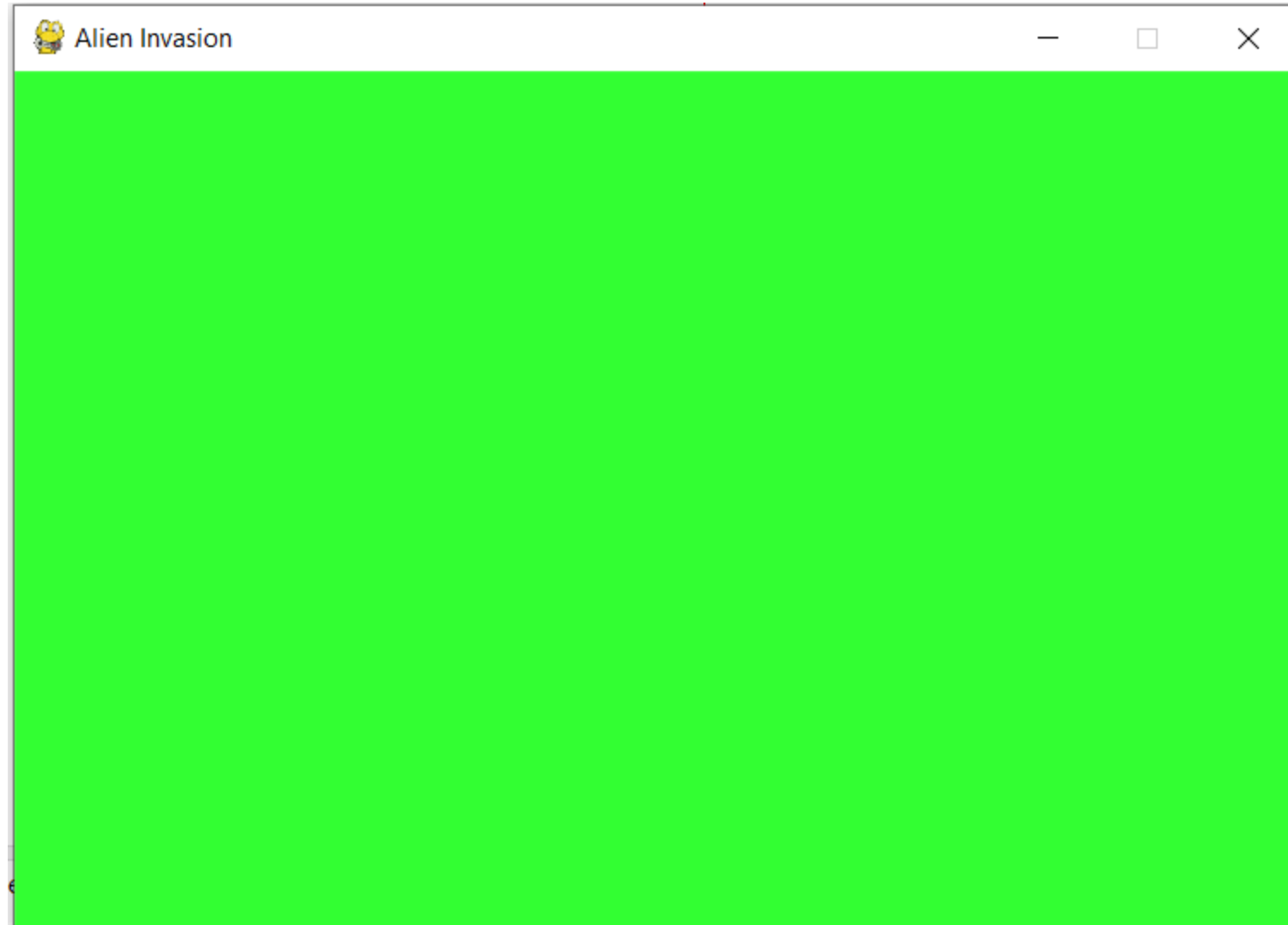


```
Command Prompt
C:\Users\Sahand>pip install pygame_
```

# Version -0

```
1 import sys
2 import pygame
3
4 def run_game():
5     bg_color = (50, 255, 50)
6
7     pygame.init()
8     screen = pygame.display.set_mode((1200, 800))
9     pygame.display.set_caption("Alien Invasion")
10
11     while True:
12         screen.fill(bg_color)
13         for event in pygame.event.get():
14             if event.type == pygame.QUIT:
15                 sys.exit()
16         pygame.display.flip()
17
18 run_game()
```

# Version-0



# Version -1

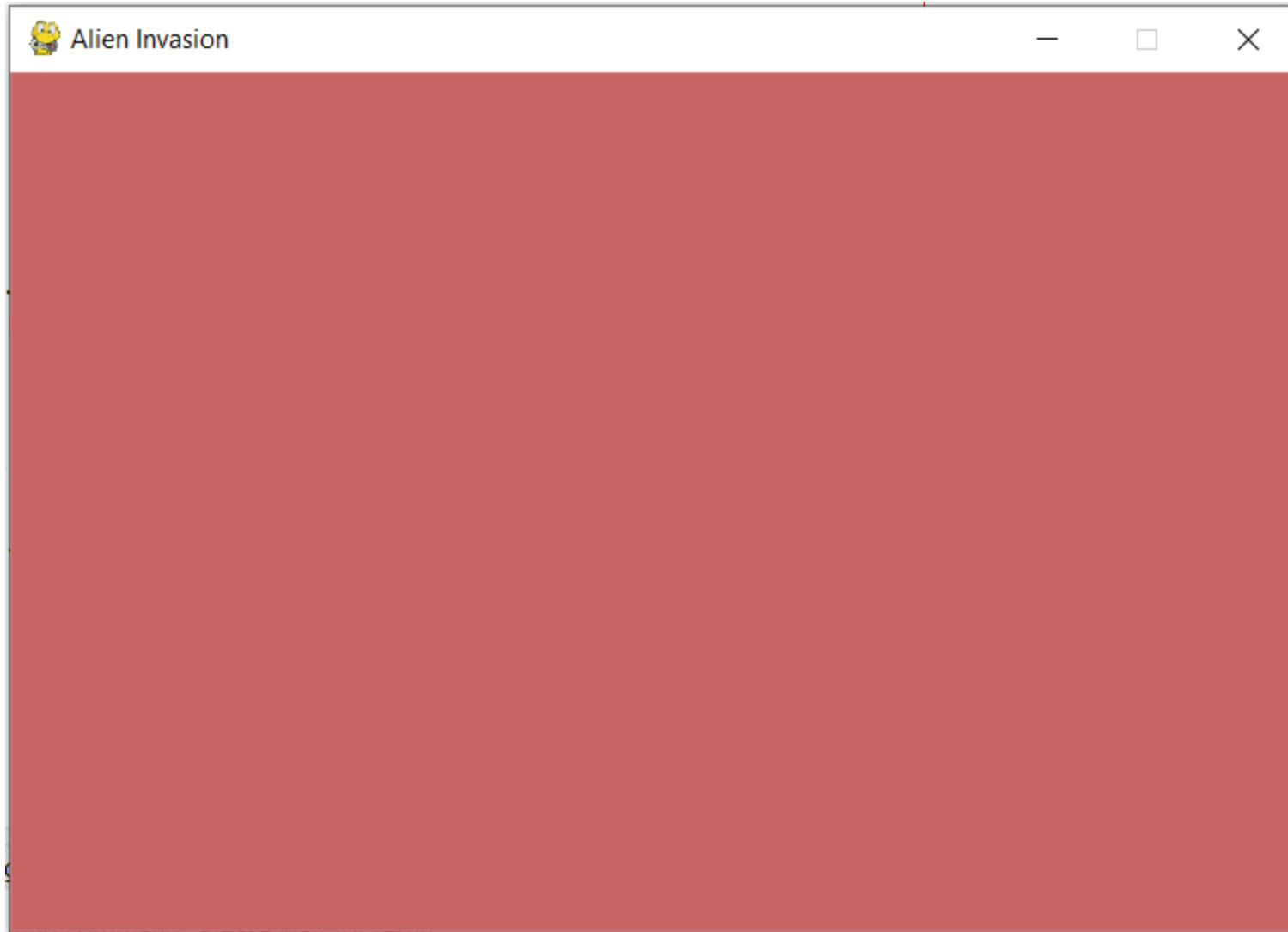
```
1 import sys
2 import pygame
3 from settings import Settings
4
5 def run_game():
6     ai_settings = Settings()
7     pygame.init()
8     screen = pygame.display.set_mode(
9         (ai_settings.screen_width, ai_settings.screen_height))
10    pygame.display.set_caption("Alien Invasion")
11
12    while True:
13        screen.fill(ai_settings.bg_color)
14        for event in pygame.event.get():
15            if event.type == pygame.QUIT:
16                sys.exit()
17        pygame.display.flip()
18
19 run_game()
```

# Version -1

```
1 class Settings():
2     def __init__(self):
3         """Initialize the game's settings."""
4         # Screen settings
5         self.screen_width = 600
6         self.screen_height = 400
7         self.bg_color = (200, 100, 100)
```



# Version -1



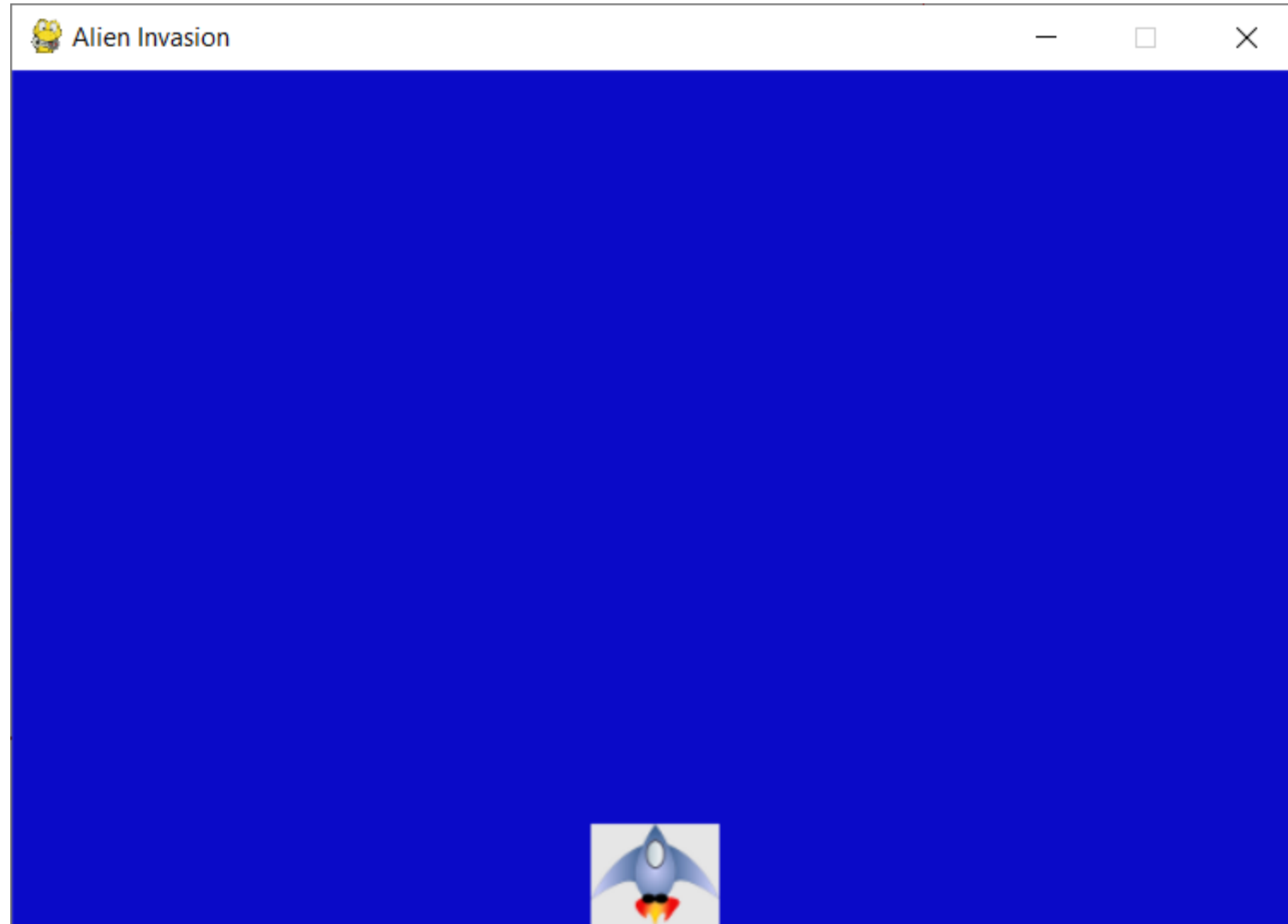
# Version -2

```
1 import sys
2 import pygame
3 from settings import Settings
4 from ship import Ship
5 def run_game():
6     ai_settings = Settings()
7     pygame.init()
8     screen = pygame.display.set_mode((ai_settings.screen_width,
9                                       ai_settings.screen_height))
10    pygame.display.set_caption("Alien Invasion")
11    # Make a ship.
12    ship = Ship(screen)
13
14    while True:
15        screen.fill(ai_settings.bg_color)
16        ship.blitme()
17
18        for event in pygame.event.get():
19            if event.type == pygame.QUIT:
20                sys.exit()
21        pygame.display.flip()
22
23    run_game()
```

## Version -2

```
1 import pygame
2 class Ship():
3     def __init__(self, screen):
4         """Initialize the ship and set its starting position."""
5         self.screen = screen
6         # Load the ship image and get its rect.
7         self.image = pygame.image.load('images/ship.bmp')
8         self.rect = self.image.get_rect()
9         self.screen_rect = screen.get_rect()
10        # Start each new ship at the bottom center of the screen.
11        self.rect.centerx = self.screen_rect.centerx
12        self.rect.bottom = self.screen_rect.bottom
13
14    def blitme(self):
15        """Draw the ship at its current location."""
16        self.screen.blit(self.image, self.rect)
```

# Version -2



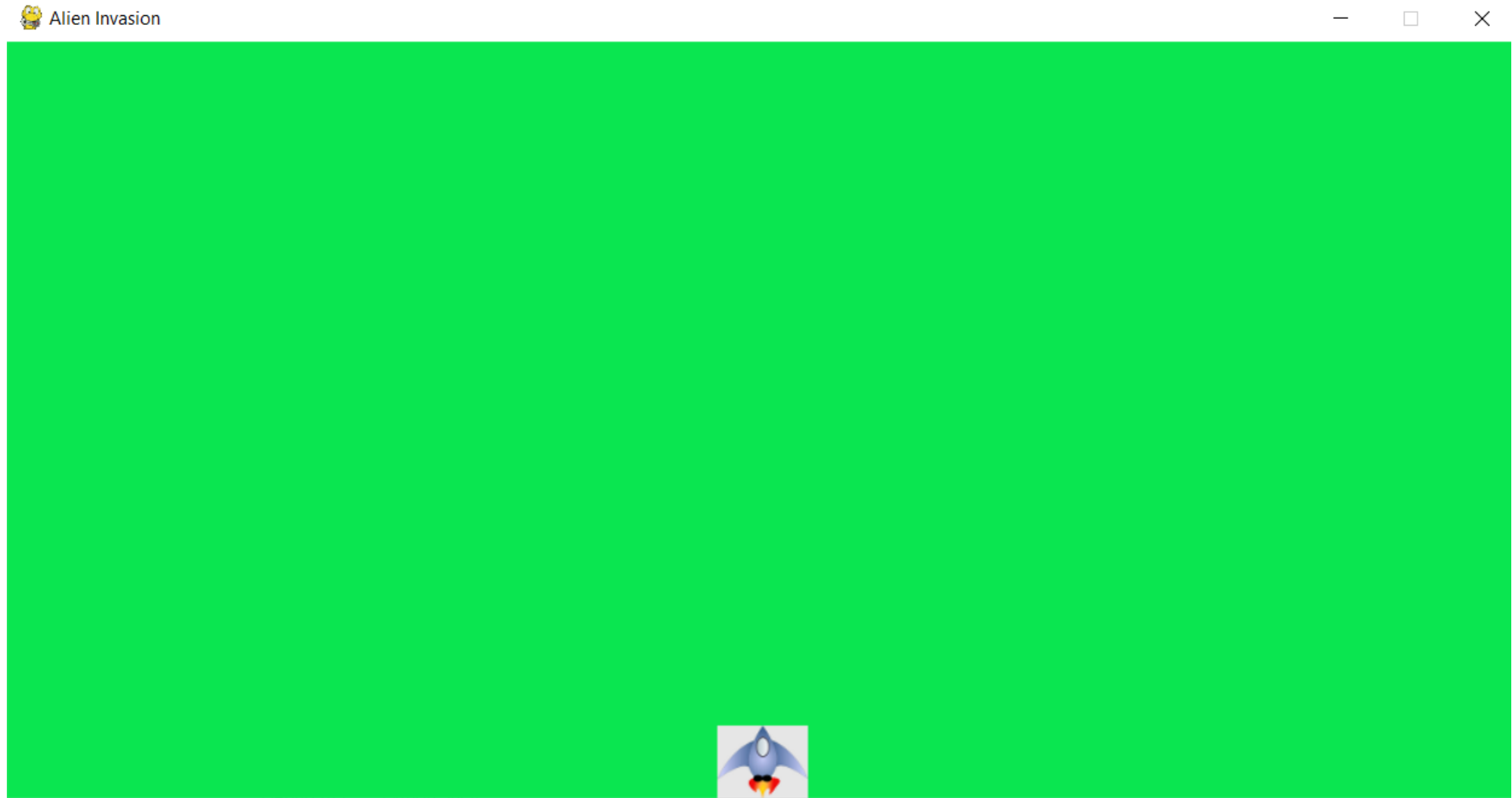
# Version -3

```
1 import pygame
2 from settings import Settings
3 from ship import Ship
4 import game_functions as gf
5
6 def run_game():
7     ai_settings = Settings()
8     pygame.init()
9     screen = pygame.display.set_mode((ai_settings.screen_width,
10                                     ai_settings.screen_height))
11     pygame.display.set_caption("Alien Invasion")
12     # Make a ship.
13     ship = Ship(screen)
14
15     while True:
16         screen.fill(ai_settings.bg_color)
17         ship.blitme()
18         gf.check_events()
19         pygame.display.flip()
20
21 run_game()
```

## Version -3 : game\_functions.py

```
1 import sys
2 import pygame
3
4 def check_events():
5     """Respond to keypresses and mouse events."""
6     for event in pygame.event.get():
7         if event.type == pygame.QUIT:
8             sys.exit()
```

# Version -3



# Version -4

```
1  import pygame
2  from settings import Settings
3  from ship import Ship
4  import game_functions as gf
5
6  def run_game():
7      ai_settings = Settings()
8      pygame.init()
9      screen = pygame.display.set_mode((ai_settings.screen_width,
10                                       ai_settings.screen_height))
11      pygame.display.set_caption("Alien Invasion")
12      # Make a ship.
13      ship = Ship(screen)
14
15      while True:
16          gf.check_events(ship)
17          gf.update_screen(ai_settings, screen, ship)
18
19
20  run_game()
```



# Version -4 : game\_functions.py(improved)

```
4 def check_events():
5     """Respond to keypresses and mouse events."""
6     for event in pygame.event.get():
7         if event.type == pygame.QUIT:
8             sys.exit()
9
10
11
12 def check_events(ship):
13     """Respond to keypresses and mouse events."""
14     for event in pygame.event.get():
15         if event.type == pygame.QUIT:
16             sys.exit()
17         elif event.type == pygame.KEYDOWN:
18             if event.key == pygame.K_RIGHT:
19                 ship.rect.centerx += 1
20
21 def update_screen(ai_settings, screen, ship):
22     """Update images on the screen and flip to the new screen."""
23     # Redraw the screen during each pass through the loop.
24     screen.fill(ai_settings.bg_color)
25     ship.blitme()
26     pygame.display.flip()
```

# Version -5

```
6 class AlienInvasion:
7     """Overall class to manage game assets and behavior."""
8     def __init__(self):
9         pygame.init()
10        self.settings = Settings()
11        self.screen = pygame.display.set_mode((0, 0), pygame.FULLSCREEN)
12        self.settings.screen_width = self.screen.get_rect().width
13        self.settings.screen_height = self.screen.get_rect().height
14        pygame.display.set_caption("Alien Invasion")
15        self.ship = Ship(self)
16
17    def run_game(self):
18        """Start the main loop for the game."""
19        while True:
20            self._check_events()
21            self.ship.update()
22            self._update_screen()
23
24    def _check_events(self):
25        """Respond to keypresses and mouse events."""
26        for event in pygame.event.get():
27            if event.type == pygame.QUIT:
28                sys.exit()
29            elif event.type == pygame.KEYDOWN:
30                self._check_keydown_events(event)
31            elif event.type == pygame.KEYUP:
32                self._check_keyup_events(event)
33
```

## Version -5

```
34 def _check_keydown_events(self, event):
35     """Respond to keypresses."""
36     if event.key == pygame.K_RIGHT:
37         self.ship.moving_right = True
38     elif event.key == pygame.K_LEFT:
39         self.ship.moving_left = True
40     elif event.key == pygame.K_q:
41         sys.exit()
42
43 def _check_keyup_events(self, event):
44     """Respond to key releases."""
45     if event.key == pygame.K_RIGHT:
46         self.ship.moving_right = False
47     elif event.key == pygame.K_LEFT:
48         self.ship.moving_left = False
49
50 def _update_screen(self):
51     """Update images on the screen, and flip to the new screen."""
52     self.screen.fill(self.settings.bg_color)
53     self.ship.blitme()
54
55     pygame.display.flip()
```

# Version -5

```
59     # Make a game instance, and run the game.  
60     ai = AlienInvasion()  
61     ai.run_game()
```

# Version -5

```
1 class Settings:
2     """A class to store all settings for Alien Invasion."""
3
4     def __init__(self):
5         """Initialize the game's settings."""
6         # Screen settings
7         self.screen_width = 1200
8         self.screen_height = 800
9         self.bg_color = (230, 230, 230)
10
11        # Ship settings
12        self.ship_speed = 1.5
```

## Version -5

```
1 import pygame
2
3 class Ship:
4     def __init__(self, ai_game):
5         self.screen = ai_game.screen
6         self.settings = ai_game.settings
7         self.screen_rect = ai_game.screen.get_rect()
8         self.image = pygame.image.load('images/ship.bmp')
9         self.rect = self.image.get_rect()
10        self.rect.midbottom = self.screen_rect.midbottom
11        self.x = float(self.rect.x)
12        self.moving_right = False
13        self.moving_left = False
14
15    def update(self):
16        if self.moving_right and self.rect.right < self.screen_rect.right:
17            self.x += self.settings.ship_speed
18        if self.moving_left and self.rect.left > 0:
19            self.x -= self.settings.ship_speed
20        self.rect.x = self.x
21
22    def blitme(self):
23        self.screen.blit(self.image, self.rect)
24
```