### Crear una clase de configuración

Cada vez que introduzcamos una nueva funcionalidad en el juego, normalmente crearemos también algunos ajustes nuevos. En lugar de añadir ajustes por todo el código, escribamos un módulo llamado settings que contenga una clase llamada Settings para almacenar todos estos valores en un solo lugar. Este enfoque nos permite trabajar con un solo objeto settings cada vez que necesitemos acceder a un ajuste individual. Esto también facilita la modificación de la apariencia y el comportamiento del juego a medida que crece nuestro proyecto. Para modificar el juego, cambiaremos los valores relevantes en *settings.py*, que crearemos a continuación, en lugar de buscar diferentes ajustes por todo el proyecto.

Crea un nuevo archivo llamado *settings.py* dentro de tu carpeta *alien\_invasion* , y añade esta clase inicial Settings:

**settings.py**

class Settings:  
 """A class to store all settings for Alien Invasion."""  
  
 def \_\_init\_\_(self):  
 """Initialize the game's settings."""  
 # Screen settings  
 self.screen\_width = 1200  
 self.screen\_height = 800  
 self.bg\_color = (230, 230, 230)

Para crear una instancia de Settings en el proyecto y utilizarla para acceder a nuestra configuración, tenemos que modificar *alien\_invasion.py* del siguiente modo:

**alien\_invasion.py**

--snip--  
import pygame  
  
from settings import Settings  
  
class AlienInvasion:  
 """Overall class to manage game assets and behavior."""  
  
 def \_\_init\_\_(self):  
 """Initialize the game, and create game resources."""  
 pygame.init()  
 self.clock = pygame.time.Clock()  
❶ self.settings = Settings()  
  
❷ self.screen = pygame.display.set\_mode(  
 (self.settings.screen\_width, self.settings.screen\_height))  
 pygame.display.set\_caption("Alien Invasion")  
  
 def run\_game(self):  
 --snip--  
 # Redraw the screen during each pass through the loop.  
❸ self.screen.fill(self.settings.bg\_color)  
  
 # Make the most recently drawn screen visible.  
 pygame.display.flip()  
 self.clock.tick(60)  
--snip--

Importamos Settings al archivo principal del programa. Luego creamos una instancia de Settings y la asignamos a self.settings ❶, después de hacer la llamada a pygame.init(). Cuando creamos una pantalla ❷, utilizamos los atributos screen\_width y screen\_height de self.settings, y luego utilizamos self.settings para acceder también al color de fondo al rellenar la pantalla ❸.

Cuando ejecutes ahora *alien\_invasion.py* aún no verás ningún cambio, porque lo único que hemos hecho es trasladar a otro sitio los ajustes que ya estábamos utilizando. Ahora estamos listos para empezar a añadir nuevos elementos a la pantalla.

[anterior](c12_9.html)[Subtema 10 de 43: (Ver todo)](c12.html)[siguiente](c12_11.html)