### Puntuaciones altas

Todos los jugadores quieren superar la puntuación máxima de un juego, así que vamos a registrar e informar de las puntuaciones máximas para dar a los jugadores algo por lo que esforzarse. Almacenaremos las puntuaciones altas en GameStats:

**game\_stats.py**

def \_\_init\_\_(self, ai\_game):  
 --snip--  
 # High score should never be reset.  
 self.high\_score = 0

Como la puntuación máxima no debe restablecerse nunca, inicializamos high\_score en \_\_init\_\_() en lugar de en reset\_stats().

A continuación, modificaremos Scoreboard para que muestre la puntuación máxima. Empecemos con el método \_\_init\_\_():

**scoreboard.py**

def \_\_init\_\_(self, ai\_game):  
 --snip--  
 # Prepare the initial score images.  
 self.prep\_score()  
❶ self.prep\_high\_score()

La puntuación alta se mostrará separada de la puntuación, por lo que necesitamos un nuevo método, prep\_high\_score(), para preparar la imagen de la puntuación alta ❶.

Aquí tienes el método prep\_high\_score():

**scoreboard.py**

def prep\_high\_score(self):  
 """Turn the high score into a rendered image."""  
❶ high\_score = round(self.stats.high\_score, -1)  
 high\_score\_str = f"{high\_score:,}"  
❷ self.high\_score\_image = self.font.render(high\_score\_str, True,  
 self.text\_color, self.settings.bg\_color)  
  
 # Center the high score at the top of the screen.  
 self.high\_score\_rect = self.high\_score\_image.get\_rect()  
❸ self.high\_score\_rect.centerx = self.screen\_rect.centerx  
❹ self.high\_score\_rect.top = self.score\_rect.top

Redondeamos la puntuación máxima a la decena más próxima y le damos formato con comas ❶. A continuación, generamos una imagen a partir de la puntuación máxima ❷, centramos horizontalmente la puntuación máxima rect ❸, y establecemos su atributo top para que coincida con la parte superior de la imagen de la puntuación ❹.

El método show\_score() dibuja ahora la puntuación actual en la parte superior derecha y la puntuación alta en la parte superior central de la pantalla:

**scoreboard.py**

def show\_score(self):  
 """Draw score to the screen."""  
 self.screen.blit(self.score\_image, self.score\_rect)  
 self.screen.blit(self.high\_score\_image, self.high\_score\_rect)

Para comprobar las puntuaciones altas, escribiremos un nuevo método, check\_high\_score(), en Scoreboard:

**scoreboard.py**

def check\_high\_score(self):  
 """Check to see if there's a new high score."""  
 if self.stats.score > self.stats.high\_score:  
 self.stats.high\_score = self.stats.score  
 self.prep\_high\_score()

El método check\_high\_score() comprueba la puntuación actual comparándola con la puntuación alta. Si la puntuación actual es mayor, actualizamos el valor de high\_score y llamamos a prep\_high\_score() para actualizar la imagen de la puntuación alta.

Tenemos que llamar a check\_high\_score() cada vez que se golpea a un alienígena después de actualizar la puntuación en \_check\_bullet\_alien\_collisions():

**alien\_invasion.py**

def \_check\_bullet\_alien\_collisions(self):  
 --snip--  
 if collisions:  
 for aliens in collisions.values():  
 self.stats.score += self.settings.alien\_points \* len(aliens)  
 self.sb.prep\_score()  
 self.sb.check\_high\_score()  
 --snip--

Llamamos a check\_high\_score() cuando el diccionario collisions está presente, y lo hacemos después de actualizar la puntuación de todos los alienígenas que han sido golpeados.

La primera vez que juegues a *Alien Invasion*, tu puntuación será la puntuación máxima, por lo que se mostrará como la puntuación actual y la puntuación máxima. Pero cuando empieces una segunda partida, tu puntuación alta debería aparecer en el centro y tu puntuación actual debería aparecer a la derecha, como se muestra en la [Figura 14-4](#figure14-4).

|  |
| --- |
| Figura |



14-4: La puntuación máxima aparece en la parte superior central de la pantalla.

[anterior](c14_22.html)[Subtema 23 de 29: (Ver todo)](c14.html)[siguiente](c14_24.html)