

Part 10 - Bouncing Bombs

By *Richard Davey* on 20th February 2018 @*photonstorm*

Para deixar nosso jogo mais completo, é hora de adicionar alguns vilões. Isso é um elemento de desafio, algo que antes estava faltando.

A ideia é: Quando você coletar todas as estrelas pela primeira vez, será lançada uma bomba saltitante. A bomba irá saltar aleatoriamente pela fase e se você colidir com ela, todas as estrelas irão reaparecer para que você as colete de novo, e se você for atingido por uma nova bomba será lançada. Isto irá dar ao jogador um desafio: conseguir a pontuação alta sem morrer.

A primeira coisa que precisamos é um Group para as bombas e alguns Collider

```
bombs = this.physics.add.group();  
  
this.physics.add.collider(bombs, platforms);  
  
this.physics.add.collider(player, bombs, hitBomb, null, this);
```

As bombas irão quicar nas plataformas, e se o player encostar nelas nós iremos usar a função `hitBomb`. Ela irá parar o jogo e pintar o player de vermelho.

```
function hitBomb (player, bomb)  
{  
    this.physics.pause();  
  
    player.setTint(0xff0000);  
  
    player.ans.play('turn');  
  
    gameOver = true;  
}
```

Até agora tudo bem, mas nós precisamos soltar as bombas. Para fazer isto precisamos da função `collectStar`:

```

function collectStar (player, star)
{
    star.disableBody(true, true);

    score += 10;
    scoreText.setText('Score: ' + score);

    if (stars.countActive(true) === 0)
    {
        stars.children.iterate(function (child) {

            child.enableBody(true, child.x, 0, true, true);

        });

        var x = (player.x < 400) ? Phaser.Math.Between(400, 800) :
        Phaser.Math.Between(0, 400);

        var bomb = bombs.create(x, 16, 'bomb');
        bomb.setBounce(1);
        bomb.setCollideWorldBounds(true);
        bomb.setVelocity(Phaser.Math.Between(-200, 200), 20);

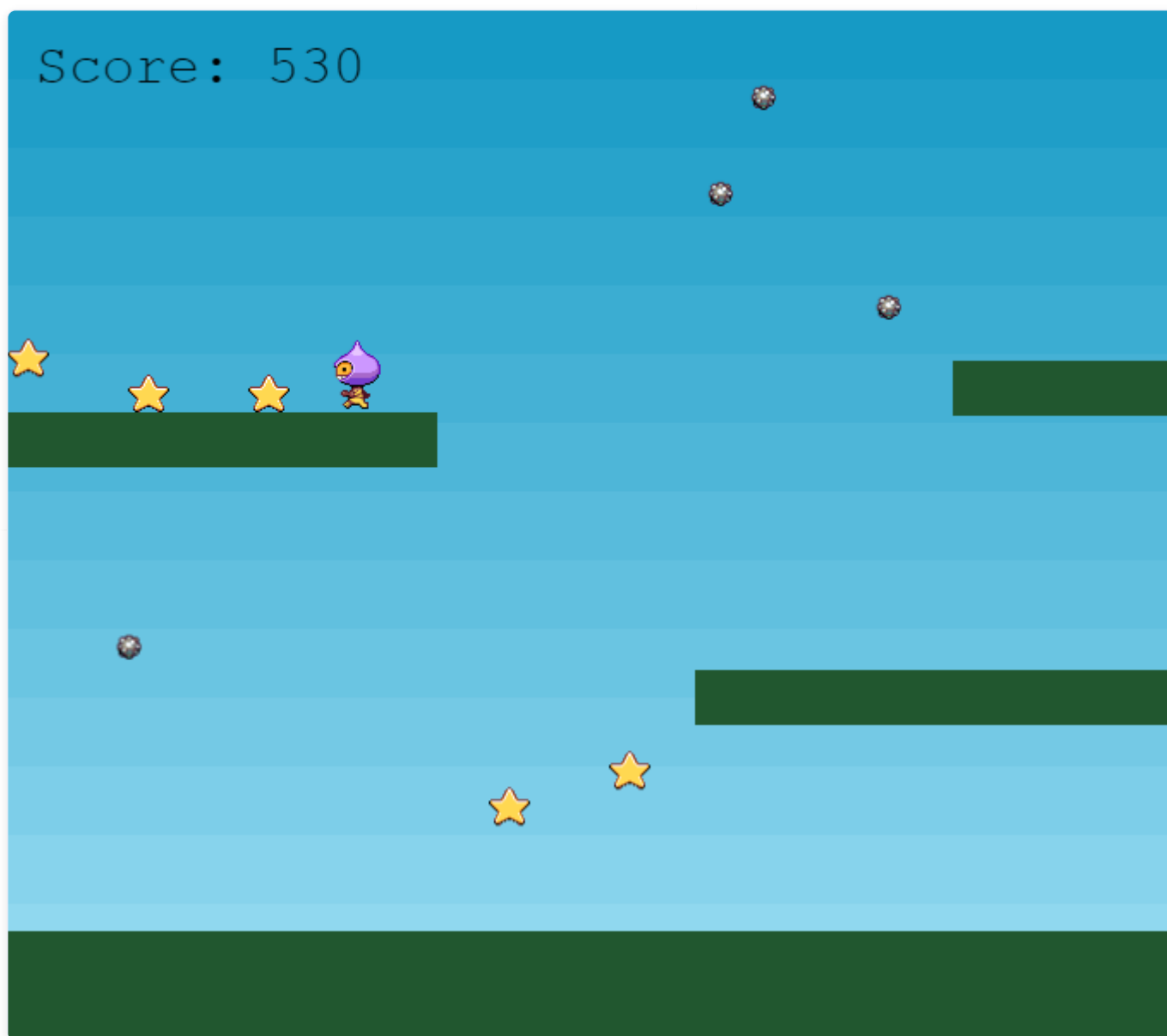
    }
}

```

Nós utilizamos um método do Grupo chamado `countActive` para ver quantas sobraram vivas. Se não houver nenhuma, então o jogador coletou todas, com isso utilizamos a função de `iterate` para reativar todas as estrelas e resetar suas posições para zero. Isto fará com que todas as estrelas caiam do topo da tela de novo;

A próxima parte do código cria a bomba. Primeiro nós escolhemos uma coordenada aleatória, sempre no lado oposto ao player, apenas para dar uma chance a ele. Quando a bomba é criada, ela é configurada para colidir com o world, quicar e ter uma velocidade aleatória.

O resultado final é este pequeno sprite de bomba que salta pela tela. pequena para ser facilmente evitada, no começo, mas assim que os números aumentam fica bem difícil!



E o nosso jogo está feito :)

Conclusão

Você agora aprendeu como criar um sprite com propriedades físicas, controlar movimento e torná-lo interativo com outros objetos em um pequeno game. Você pode fazer muitas outras coisas para aperfeiçoar este jogo. Por que não aumentar o tamanho da fase e permitir que a câmera role? Talvez adicionar vilões de diferentes valores, ou dar ao player uma barra de vida.

Ou para um jogo não violento, você pode torná-lo um speed-run e desafiar o jogador a coletar as estrelas o mais rápido possível.

Com o que você aprendeu neste tutorial, e as centenas de exemplos disponíveis, você tem agora uma fundação sólida para futuros projetos. Mas como sempre, se você tiver alguma dúvida, precisar de um conselho, ou quer compartilhar o que você está trabalhando, sinta-se à vontade para pedir ajuda no fórum do Phaser.

Facebook Instant Games

Phaser 3 agora tem suporte total para criar Facebook Instant Games. Agora você pode ver como criar um jogo utilizando o Phaser, por quê não ver como é fácil converter um Instant Game no nosso [Guia de primeiros passos](#).