

## ブロック崩しのカスタマイズ例

⑧ 緑色のブロックを崩したら、ボールを3つ増やす

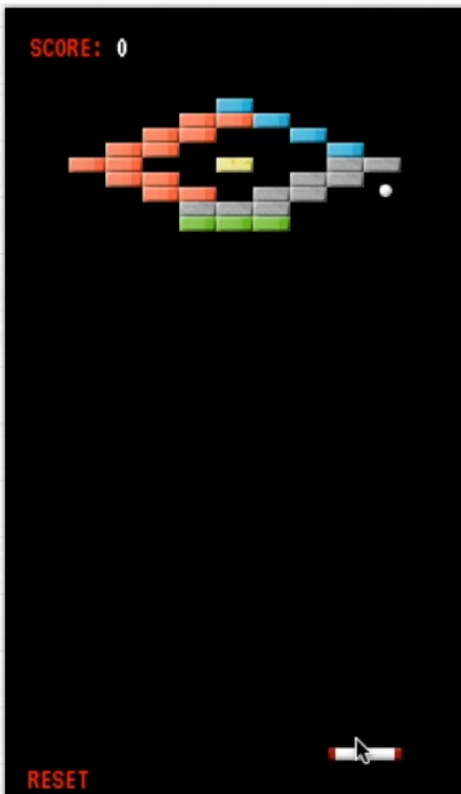
---

緑色のブロックを崩したら、ボールを3つ増やす

※このカスタマイズは、「ブロックの色を増やす」の後に行えます

## ■ 緑色のブロックを追加し、それを崩したらボールを3つ増やす

```
100
101     block.width = 30;
102     block.height = 12;
103
104     block.point = point;
105     block.color = color;
106
107     BB.stage.addChild(block);
108     BB.blocks.push(block);
109
110     return block;
111 },
112
113 // Create a ball and add it to PIXI.Stage
114 addBall: function () {
115     var texture = PIXI.Texture.fromImage(imgPath
116     ["ball"], false);
117     var ball = new PIXI.Sprite(texture);
118
119     ball.anchor.x = 0.5;
120     ball.anchor.y = 0.5;
121
122     ball.position.x = parseInt(BB.renderer.
123     width * 0.5);
124     ball.position.y = 200;
125
126     ball.width = 10;
127     ball.height = 10;
```

A screenshot of a game interface. At the top left, it says "SCORE: 0" in red. The main area is a black grid with a pattern of colored blocks (orange, blue, yellow, grey, green) forming a shape. A small white ball is on the right side of the grid. At the bottom left, it says "RESET" in red. A mouse cursor is pointing at the bottom right of the grid.

## ■ 修正箇所(1)

- ・ ブロックを配置する関数addBlockの中に、以下の行を追加する

103			block.point = point;
104			block.color = color;
105			

- ・ ブロックに、得点の情報を加えている
- ・ 同様に、色の情報を加えている

## ■ 修正箇所(2)

- ボールがブロックに当たったら
  - ブロックの色を調べ、"green"なら、

BB.addBall()を3回実行する

```
391         if (BB.isBallHit(ball, block)) {
392             if ( block.color == "green" ){
393                 BB.addBall();
394                 BB.addBall();
395                 BB.addBall();
396             }
397         BB.addScore(block.point);
```