

ブロック崩しのカスタマイズ例

⑤ ブロックの色ごとの得点を変える

■ 初期のプログラム

- ブロックを崩すと、
色に応じて得点が入る

ブロックの色	得点
銀色	200
金色	3000000
その他 (赤、青、緑)	1000

■ 対応するプログラム

- main.js addBlock関数
 - 10行目～12行目
 - 72行目～88行目

10	SETTINGS_POINT = 1000,
11	SETTINGS_POINT_SILVER = 200,
12	SETTINGS_POINT_GOLD = 3000000;

```
72     addBlock: function (x, y, color) {
73         switch (color) {
74             case "red":
75             case "blue":
76                 var point = SETTINGS_POINT;
77                 break;
78             case "silver":
79                 var point =
80                     SETTINGS_POINT_SILVER;
81                 break;
82             case "gold":
83                 var point =
84                     SETTINGS_POINT_GOLD;
85                 break;
86             default:
87                 var point = SETTINGS_POINT;
88                 color = "red";
89                 break;
90         }
91     }
```

■ カスタマイズの例

- 初期の状態では、赤と青で得点と同じ。「赤を500点」「青を1000点」にする

■ 修正箇所

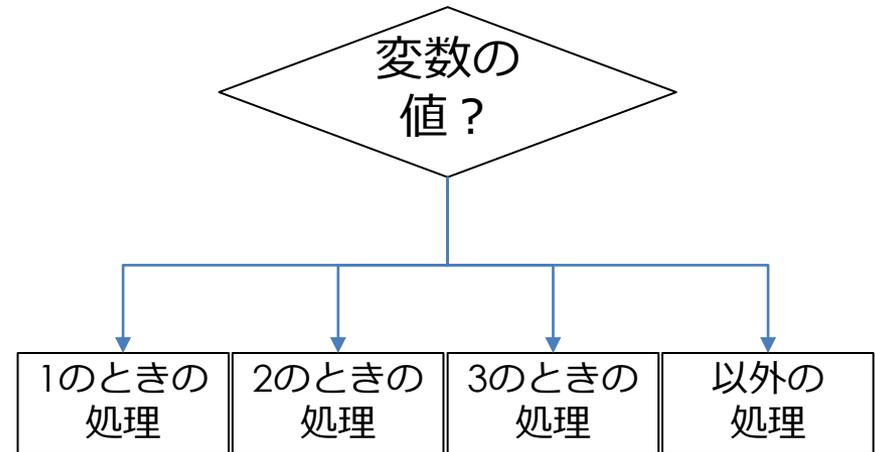
- main.js addBlock関数
 - 10行目 行を追加
 - 74行目前後 行を追加
 - swtich文
 - redだったときの
point指定

```
10 SETTINGS_POINT_RED = 500,  
11 SETTINGS_POINT = 1000,  
12 SETTINGS_POINT_SILVER = 200,  
13 SETTINGS_POINT_GOLD = 3000000;
```

```
73 addBlock: function (x, y, color) {  
74     switch (color) {  
75         case "red":  
76             var point =  
77                 SETTINGS_POINT_RED;  
78             break;  
79         case "blue":  
80             var point = SETTINGS_POINT;  
81             break;  
82         case "silver":
```

■ 補足事項・学習のヒント：switchによる条件分岐

```
switch (変数) {  
  case 値1:  
    (変数の値が1のときの処理)  
    ...  
    break;  
  case 値2:  
    (変数の値が2のときの処理)  
    ...  
    break;  
  case 値3:  
    (変数の値が3のときの処理)  
    ...  
    break;  
  default:  
    (変数がどの値でもないときの処理)  
    ...  
}
```



！：キーワードbreakが無いと、違う動作になるので書き忘れないようにすること