



# WFDD Week 7

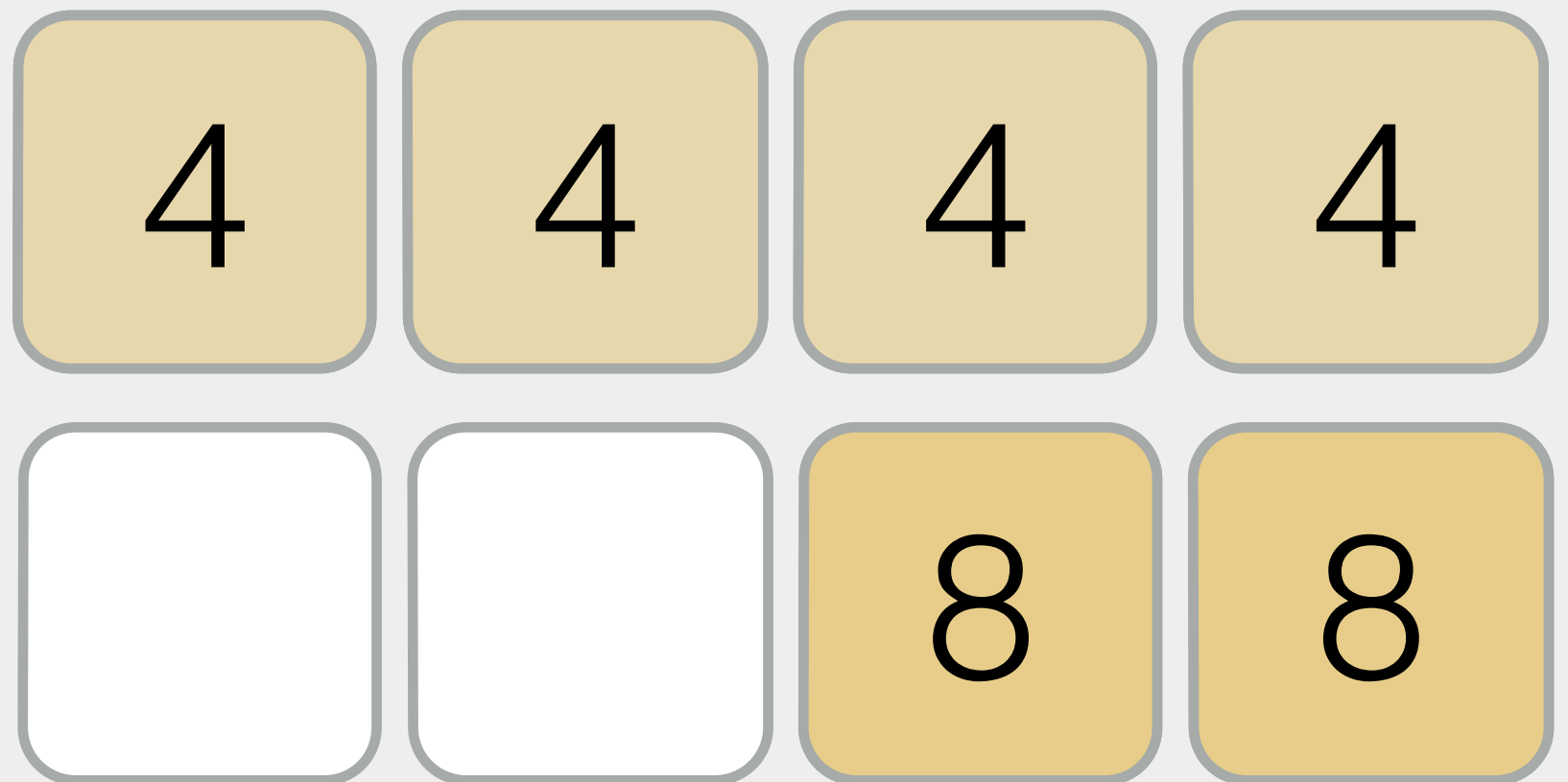
By Breezewish

|     |      |    |   |
|-----|------|----|---|
| 256 | 1024 | 16 | 4 |
| 32  | 64   | 8  |   |
|     | 32   | 16 |   |
|     |      |    |   |

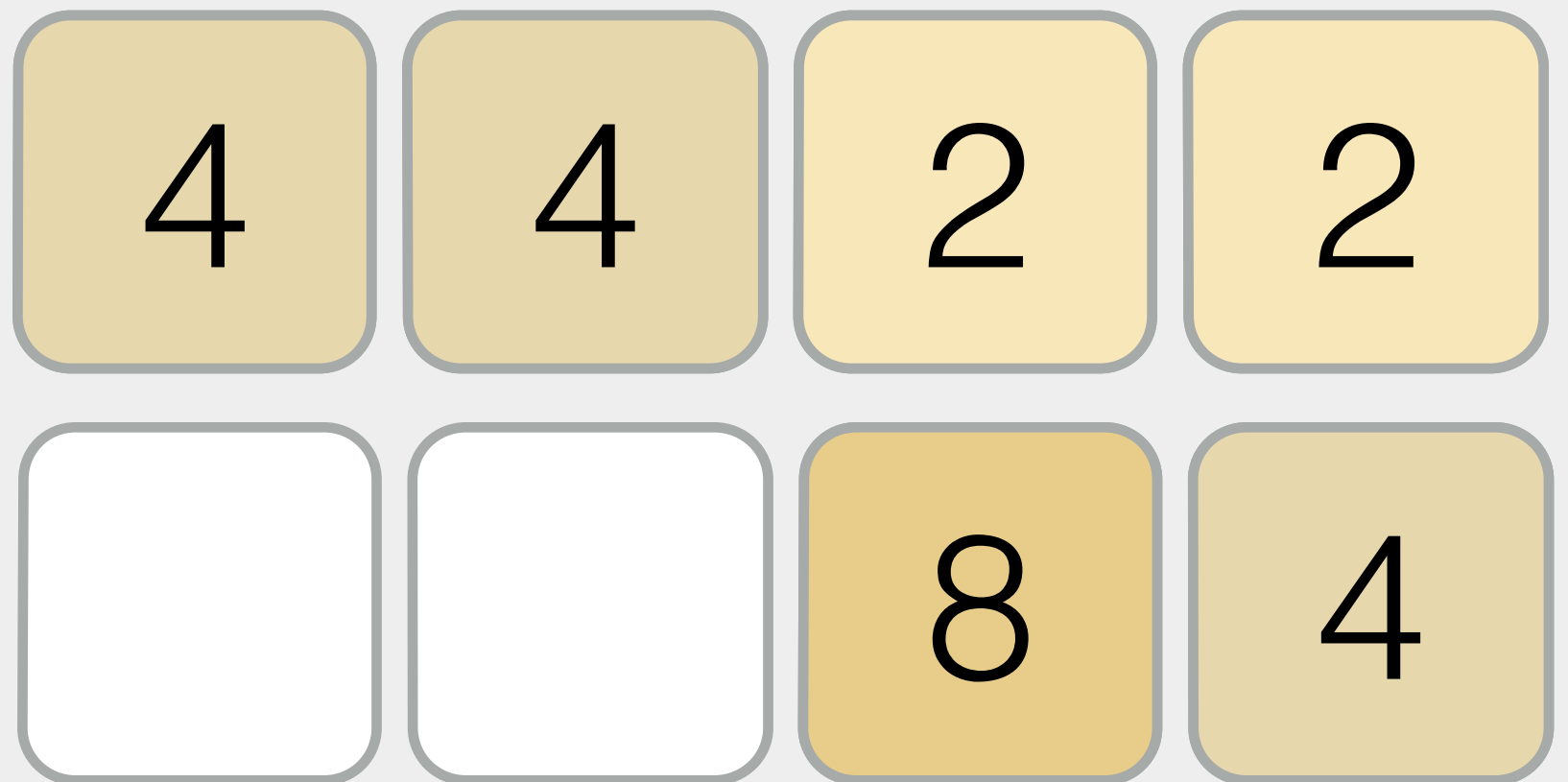
# 2048 RULES



# 2048 RULES



# 2048 RULES



# MOVE RIGHT

for each line in the grid:

for each tile in the line from **right to left**:

move right until touches border/tile

merge to right tile if

the right tile exists &&

their values are equal &&

the right tile is not previously merged

# BUILD STEPS

iteration 1:

- 2D-array based map storage
- rendering map via HTML
- placing random tile
- receiving keyboard events
- moving tile
- animating via CSS3

# BUILD STEPS

updateGrid()

```
<div ....></div>
<div ....></div>
<div ....></div>
<div ....>4</div>
<div ....></div>
<div ....></div>
<div ....>8</div>
<div ....></div>
```

```
var map = [
  [0, 0, 0, 4],
  [0, 0, 8, 0],
  [0, 0, 0, 0],
  [0, 0, 0, 0],
];
```

|  |  |   |   |
|--|--|---|---|
|  |  |   | 4 |
|  |  | 8 |   |
|  |  |   |   |
|  |  |   |   |



# BUILD STEPS

iteration 2:

- 2D-array based map storage
- rendering map via HTML
- placing random tile
- receiving keyboard events
- moving tile
- animating via CSS3

# BUILD STEPS

iteration 3:

- 2D-array based map storage
- rendering map via HTML
- placing random tile
- receiving keyboard events
- moving tile
- animating via CSS3

## How to render the grid?

Solution1:

```
display: inline-block
```

Solution2:

```
position: absolute  
class="cell-row-1-col-1"
```

Solution3:

```
position: absolute  
style="left:..., top:..."
```

## How to render tiles of different values?

### Different

- background-color
- color
- font-size
- [text]

### Same

- border-radius: 3px
- position: absolute
- text-align: center

|      |      |     |     |     |   |
|------|------|-----|-----|-----|---|
| 2048 | 1024 | 512 | 256 | 128 |   |
| 64   | 32   | 16  | 8   | 4   | 2 |

Inline style directly

```
<div  
  class="tile"  
  style="background:#e2c164;font-  
    size:45px;color:#FFF;"  
>512</div>
```

# SOL2

Use class

```
<div  
    class="tile tile-512"  
>512</div>
```

```
.tile-512 {  
    ...  
}
```

# SOL3

Use attribute

```
<div  
  class="tile"  
  data-point="512"  
>512</div>  
  
.tile[data-point="512"] {  
  ...  
}
```

# SOL3

## V2

Use *::before*, *::after* pseudo-element

```
<div  
  class="tile"  
  data-point="512"  
>512</div>
```

```
.tile[data-point="512"] {  
  ...  
}  
.tile[data-point="512"]::after {  
  content: "512";  
}
```



# SOL3

## V3

Use *content:attr()* to cut down styles

```
<div
```

```
  class="tile"
```

```
  data-point="512"
```

```
></div>
```

```
.tile[data-point="512"] {
```

```
  ...
```

```
}
```

```
.tile::after {
```

```
  content: attr(data-point);
```

```
}
```

# SOL3

## V4

Use *:not()* pseudo class to exclude zero

```
<div  
  class="tile"  
  data-point="512"  
></div>
```

```
.tile[data-point="512"] {  
  ...  
}  
.tile:not([data-point="0"]):after {  
  content: attr(data-point);  
}
```

## Download project template

From CDN mirror:

<http://breeswish-org.qiniudn.com/wfdd-week-7-template.zip>

From original:

<https://breeswish.org/file/wfdd-week-7-template.zip>

# TEMPLATE

## ITERATION 1

```
function createMap() {}  
// (根据 size) 初始化 map 为一个二维数组  
  
function createGrid() {}  
// 初始化网格 DOM  
  
function updateGrid() {}  
// 将 map 内容更新到网格 DOM  
  
function createRandomTile() {}  
// 在地图上放置一块新格子
```

# TEMPLATE ITERATION 2

```
function move(direction) {}  
// 根据 direction 移动格子
```