

06

Jogo - PARTE 3

Transcrição

Nosso jogo ainda não sabe dizer se jogador ganhou ou perdeu, muito menos reiniciar a partida. Vamos adicionar o esqueleto de três novas funções: `ganhou` , `perdeu` , `ganhouOuPerdeu` , `reinicia` :

```
var criaJogo = function (sprite) {

    var etapa = 1;
    var lacunas = [];
    var palavraSecreta = '';

    var criaLacunas = function () {

        for (let i = 0; i < palavraSecreta.length; i++) {
            lacunas.push('_');
        }
    };

    var proximaEtapa = function () {

        etapa = 2;
    };

    var setPalavraSecreta = function (palavra) {

        palavraSecreta = palavra;
        criaLacunas();
        proximaEtapa();
    };

    var getLacunas = function () {

        return lacunas;
    };

    var getEtapa = function () {

        return etapa;
    };

    var processaChute = function (chute) {

        var exp = new RegExp(chute, 'gi')
            , resultado
            , acertou = false;

        while (resultado = exp.exec(palavraSecreta))
            acertou = lacunas[resultado.index] = chute;

        if (!acertou) sprite.nextFrame();
    };
}
```

```
// novas funcionalidades que precisamos implementar

var ganhou = function () {
    console.log('falta implementar');
};

var perdeu = function () {
    console.log('falta implementar');
};

var ganhouOuPerdeu = function () {
    console.log('falta implementar');
};

var reinicia = function () {
    console.log('falta implementar');
};

// adicionou novas propriedades
return {
    setPalavraSecreta: setPalavraSecreta,
    getLacunas: getLacunas,
    getEtapa: getEtapa,
    processaChute: processaChute,
    ganhou: ganhou,
    perdeu: perdeu,
    ganhouOuPerdeu: ganhouOuPerdeu,
    reinicia: reinicia
}
};
```

Como você deve implementar a lógica dessas funções? Dessa vez não haverá dicas e você precisará usar o bom senso para resolver essa questão. Como de costume, vou demonstrar como essas funções devem se comportar realizando testes através do console com esta parte já implementada. Inclusive, você pode usar a mesma lógica de teste aqui para verificar seu código:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width">
    <link rel="stylesheet" href="css/bootstrap.min.css">
    <link rel="stylesheet" href="css/sprite.css">
    <link rel="stylesheet" href="css/lacuna.css">
    <title>Jogo da Forca</title>
</head>
<body>
    <div class="jumbotron">
        <h2 class="text-center">Jogo da forca</h2>
    </div>
    <div class="container">
        <div class="sprite"></div>
        <ul class="lacunas text-center"></ul>
    <div class="text-center">
```

```
<input id="entrada"
       class="entrada"
       placeholder="Palavra secreta" autofocus>
</div>
</div>

<script src="js/jquery.min.js"></script>
<script src="js/sprite.js"></script>
<script src="js/jogo.js"></script>
<script>
    var jogo = criaJogo(createSprite('.sprite'));
    console.log(jogo.getEtapa()); // 1
    jogo.setPalavraSecreta('calopsita');
    console.log(jogo.ganhouOuPerdeu()); // false
    jogo.processaChute('c');
    jogo.processaChute('a');
    jogo.processaChute('l');
    jogo.processaChute('o');
    jogo.processaChute('p');
    jogo.processaChute('s');
    jogo.processaChute('i');
    jogo.processaChute('t');
    console.log(jogo.ganhouOuPerdeu()); // true
    console.log(jogo.ganhou()); // true
    console.log(jogo.perdeu()); // false
    jogo.reinicia();
    console.log(jogo.getLacunas());
    console.log(jogo.getEtapa());
    jogo.setPalavraSecreta('calopsita');
    jogo.processaChute('z');
    jogo.processaChute('z');
    jogo.processaChute('z');
    jogo.processaChute('z');
    jogo.processaChute('z');
    jogo.processaChute('z');
    jogo.processaChute('z');
    jogo.processaChute('z');
    console.log(jogo.ganhouOuPerdeu()); // true
    console.log(jogo.ganhou()); // false
    console.log(jogo.perdeu()); // true

</script>
</body>
</html>
```