R: A expressão regular / [^ abc]/ corresponde a uma das strings d, e, f - um personagem que não é *a, b,* ou *c (* opção b).

6 Qual das seguintes expressões regulares corresponde à string Olá?

R: As opções a, be c correspondem. / Olá/ corresponde apenas à string exata Olá. /Olá/ corresponde também Olá ou Olá (o segundo eu é opcional). / Olá/,

depois do primeiro eu, corresponde a qualquer número da letra eu.

7 A expressão regular / (cd) + (de) * / corresponde a qual das seguintes strings? R: As opções a, c, d e f estão corretas. / (cd) + (de) * / corresponde a uma ou mais ocorrências de CD seguido por qualquer número de ocorrências de de.

8 Em expressões regulares, podemos expressar alternativas com qual das seguintes? R: Usamos a barra vertical, |, para expressar alternativas em expressões regulares (opção c).

9 Na expressão regular / ([0-9]) 2 /, podemos fazer referência ao primeiro dígito correspondente com qual dos seguintes?

UMA: \ 1 (opção d)

10 A expressão regular / ([0-5]) 6 \ 1 / corresponderá a qual dos seguintes? R: Na expressão regular / ([0-5]) 6 \ 1 /, o primeiro caractere é um dígito de 0 a 5, o segundo caractere é o dígito 6 e o terceiro caractere é o primeiro dígito correspondente, então ambos 060 e 565 corresponder (opções a e d). A expressão regular / (?: ninja) - (truque)? - \ 1 / corresponderá a qual dos seguintes?

11

R: Na expressão regular / (?: ninja) - (truque)? - \ 1 /, o primeiro grupo
(?: ninja) não é de captura, enquanto o segundo grupo é de captura, o que é opcional (truque)?. Mas se este segundo grupo for encontrado, no final temos uma referência anterior a ele. Portanto, ninja- e truque-truque-ninja corresponder (opções a e c).

12 Qual é o resultado de executar " 012675 ".substituir (/ 0-5 / g," a ")?

R: O código substitui todas as ocorrências de dígitos de 0 a 5 pela letra a, então
aaa67a resultados (opção a).

Capítulo 11. Técnicas de modularização de código

1 Qual mecanismo habilita variáveis de módulo privadas no padrão de módulo?

R: No padrão do módulo, os fechamentos nos permitem ocultar os internos do módulo: Os métodos da API pública do módulo mantêm os internos do módulo vivos (opção b).

2 No código a seguir que usa módulos ES6, quais identificadores podem ser acessados se o módulo for importado?

```
const spy = "Yagyu"; comando de
função () {
    retorno geral + "ordena que você faça guerra!";
}
export const general = "Minamoto";
```

R: De fora do módulo, podemos acessar apenas o em geral identificador, porque é o único identificador que foi explicitamente exportado (opção c).

3 No código a seguir que usa módulos ES6, quais identificadores podem ser acessados quando o módulo é importado?

```
const ninja = "Yagyu"; comando de função
() {
    retorno geral + "ordena que você faça guerra!";
}
const general = "Minamoto";

exportar {ninja como espião};
```

R: De fora do módulo, podemos acessar apenas o espião identificador: este é o único identificador que foi exportado como um alias do ninja variável (opção a).

4 Quais das seguintes importações são permitidas?

```
// Arquivo: staff.js
const ninja = "Yagyu"; comando de função
() {
    retorno geral + "ordena que você faça guerra!";
}
const general = "Minamoto";

exportar {ninja como espião};
```

R: A primeira importação não é permitida porque o pessoal módulo não exporta ninja e em geral identificadores (opção a). A segunda importação é permitida porque importamos todo o módulo, que é acessível através do objeto Pessoal (opção b). A terceira importação também é permitida, pois importamos o exportado espião identificador (opção c).

5 Se tivermos o seguinte código de módulo, qual instrução importará o Ninja aula?

```
//Ninja.js
exportar classe padrão Ninja {
    skulk () {return "skulking"; }
}
```

R: A primeira importação é permitida: Importamos a exportação padrão (opção a). A segunda importação é permitida: Importamos todo o módulo (opção b). A terceira importação não é permitida porque não está sintaticamente correta (após o * deve vir a parte “as Name”) (opção c).

Capítulo 12. Trabalhando o DOM

1 No código a seguir, qual das afirmações a seguir será aprovada?

```
<div id = "samurai"> </div>
<script>
    elemento const = document.querySelector ("# samurai");
```

```
    assert (element.id === "samurai", "id da propriedade é samurai"); assert (element.getAttribute ("id") === "samurai",

                "id do atributo é samurai");

    element.id = "novoSamurai";

    assert (element.id === "newSamurai", "id da propriedade é newSamurai"); assert (element.getAttribute ("id") ===
    "newSamurai",
                "id do atributo é newSamurai");
</script>
```

R: Neste código, todas as afirmações são aprovadas. O eu ia atributo e o eu ia propriedade estão vinculadas; uma mudança em um deles se reflete no outro.

2 Dado o código a seguir, como podemos acessar o elemento largura da borda estilo propriedade?

```
<div id = "element" style = "border-width: 1px;
                                                    estilo de borda: sólido; border-color: red ">
</div>
<script>
    elemento const = document.querySelector ("# elemento"); </script>
```

R: O expressão element.border-width não faz muito sentido. Ele calcula a diferença entre element.border e uma variável largura, o que definitivamente não é algo que queremos. A próxima opção, element.getAttribute ("border-width") ;, busca o atributo do elemento HTML, e não a propriedade de estilo. Finalmente, as duas últimas opções fornecem o valor de 1px (opções ce d).

3 Qual método integrado pode ter todos os estilos aplicados a um determinado elemento (estilos fornecidos pelo navegador, estilos aplicados por meio de folhas de estilo e propriedades definidas por meio do atributo style)?

R: Apenas a última opção, getComputedStyle, é um método integrado que pode ser usado para obter o estilo calculado de um determinado elemento HTML (opção c). Os outros três métodos não estão incluídos na API padrão.

4 Quando ocorre a eliminação do layout?

R: O descarte do layout ocorre quando nosso código executa uma série de leituras e gravações consecutivas no DOM, forçando sempre o navegador a recalcular as informações do layout. Isso leva a aplicativos da web mais lentos e menos responsivos.

Capítulo 13. Eventos de sobrevivência

1 Por que é importante que a adição de tarefas à fila de tarefas aconteça fora do loop de eventos?

R: Se o processo de adicionar tarefas à fila de tarefas fizer parte do loop de eventos, quaisquer eventos que ocorram enquanto o código JavaScript estiver sendo executado serão ignorados. Isso definitivamente seria uma má ideia.

2 Por que é importante que cada iteração do loop de eventos não leve muito mais do que cerca de 16 ms?

R: Para obter aplicativos de execução suave, o navegador tenta realizar a renderização cerca de 60 vezes por segundo. Como a renderização é realizada no final do loop de eventos, cada iteração não deve durar muito mais do que 16 ms, a menos que queiramos criar aplicativos lentos e irregulares.

3 Qual é o resultado da execução do código a seguir por 2 segundos?

```
setTimeout (function () {
    console.log ("Tempo limite");
}, 1000);

setInterval (function () {
    console.log ("Intervalo");
}, 500);
```

UMA: Intervalo Tempo Limite Intervalo Intervalo Intervalo (opção b). O definir-
Intervalo método chama o manipulador com pelo menos o atraso fixo entre cada chamada, até que o intervalo seja explicitamente limpo. O setTimeout método, por outro lado, chama o retorno de chamada apenas uma vez, após o atraso especificado ter decorrido. Neste exemplo, primeiro o setInterval o retorno de chamada é disparado uma vez após 500 ms. Então o setTimeout retorno de chamada é invocado após 1000 ms, e outro setInterval

imediatamente depois. Nosso exame termina com mais dois setInterval invocações de retorno de chamada, uma a 1.500 ms e a outra a 2.000 ms.

4 Qual é o resultado da execução do código a seguir por 2 segundos?

```
const timeoutId = setTimeout (function () {
    console.log ("Tempo limite");
}, 1000);

setInterval (function () {
    console.log ("Intervalo");
}, 500);

clearTimeout (timeoutId);
```

UMA: Interval Interval Interval Interval (opção c). O setTimeout chamar-
volta é liberada antes de ter a chance de disparar, portanto, neste caso, temos apenas quatro execuções do setInterval ligar de volta.

5 Qual é o resultado de executar o código a seguir e clicar no elemento com o ID interno?

```
<body>
    <div id = "outer">
        <div id = "inner"> </div>
    </div>
    <script>
        const innerElement = document.querySelector ("# inner"); const outerElement = document.querySelector
        ("# outer");
```

```
        const bodyElement = document.querySelector ("body"); innerElement.addEventListener ("click",
        function () {
            console.log ("Interno");
        });

        outerElement.addEventListener ("click", function () {
            console.log ("Externo");
        }, verdade);

        bodyElement.addEventListener ("click", function () {
            console.log ("Corpo");
        });
    <script>
</body>
```

UMA: Corpo Interno Externo (opção c). Os manipuladores de clique no innerElement e a bodyElement são registrados no modo bubbling, enquanto o manipulador de cliques no outerElement é registrado no modo de captura. Ao processar o evento, ele primeiro desce do topo e chama todos os manipuladores de eventos no modo de captura. A primeira mensagem será Exterior. Depois que a meta do evento é alcançada, no nosso caso, o elemento com ID interno, o borbulhar do evento acontece, e o evento borbulha. Portanto, a segunda mensagem será Interno, e o terceiro será Corpo.

Capítulo 14. Desenvolvimento de estratégias para vários navegadores

1. O que devemos levar em consideração ao decidir quais navegadores oferecer suporte? R: Ao decidir quais navegadores oferecer suporte, devemos pelo menos levar em consideração o seguinte:

   - As expectativas e necessidades do público-alvo
   - A participação de mercado do navegador
   - A quantidade de esforço necessária para suportar o navegador Explique o

2. problema de IDs gananciosos.

   R: Ao trabalhar com elementos de formulário, o navegador adiciona propriedades ao elemento de formulário para cada elemento descendente com um ID, para que possamos acessar esses elementos facilmente por meio do elemento de formulário. Infelizmente, isso pode substituir algumas das propriedades integradas do formulário, como açao ou enviar.

3. O que é detecção de recurso?

   R: A detecção de recurso funciona determinando se um determinado objeto ou propriedade de objeto existe e, em caso afirmativo, presume que ele fornece a funcionalidade implícita. Em vez de testar se o usuário está usando um navegador específico e, em seguida, implementar soluções alternativas com base nessas informações, testamos se um determinado recurso funciona como deveria.

4. O que é um polyfill do navegador?

   R: Se quisermos usar uma determinada funcionalidade que não é compatível com todos os navegadores-alvo, podemos usar a detecção de recursos. Se um navegador atual não suportar uma determinada funcionalidade, fornecemos nossa própria implementação, e isso é chamado de polyfilling.

índice

Símbolos

UMA

D

E

F

N

O

P

Q

R

S

T

você

V

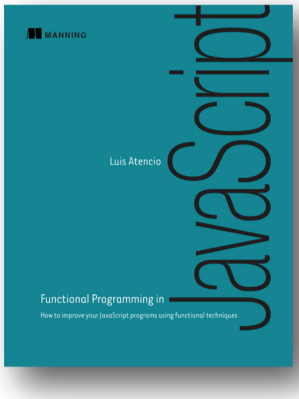
C

X

Y

Z

MAIS TÍTULOS DEMANNING



Programação Funcional em JavaScript

por Luis Atencio

ISBN: 9781617292828

272 páginas

$ 44,99

Junho de 2016

Design de aplicativo JavaScript

Uma abordagem Build First

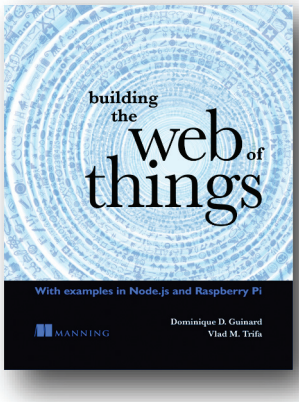
por Nicolas G. Bevacqua

ISBN: 9781617291951

344 páginas

$ 39,99

Janeiro de 2015



Construindo a Teia das Coisas

Com exemplos em Node.js e Raspberry Pi

por Dominique D. Guinard e Vlad M. Trifa

ISBN: 9781617292682

344 páginas

$ 34,99

Junho de 2016

Para obter informações sobre pedidos, acesse www.manning.com

Folha de dicas ES6 (continuação)

Aulas agir como um açúcar sintático em torno dos protótipos do JavaScript:

```
aula Pessoa {
    construtor( nome) {this.name = nome; } dance () {return true; }

}
classe Ninja estende Pessoa {
    construtor (nome, nível) {
        super (nome);
        this.level = level;
    }
    estático compare (ninja1, ninja2) {
        retornar ninja1.level - ninja2.level;
    }
}
```

Proxies controlar o acesso a outros objetos. Ações personalizadas podem ser executadas quando um objeto é interagido (por exemplo, quando uma propriedade é lida ou uma função é chamada):

```
const p = novo Proxy ( alvo, {
    obter: ( target, key) => {/ * Chamado quando propriedade acessada por meio de proxy * /},
    definir: ( destino, chave, valor) => {/ * Chamado quando a propriedade definida por meio de proxy * /}});
```

Mapas são mapeamentos entre uma chave e um valor:

- novo mapa () cria um novo mapa.
- Use o definir método para adicionar um novo mapeamento, o obter método para buscar um mapeamento, o tem método para verificar se existe um mapeamento, e o excluir método para remover um mapeamento.

Jogos são coleções de itens exclusivos:

- novo conjunto() cria um novo conjunto.
- Use o adicionar método para adicionar um novo item, o excluir método para remover um item, e o Tamanho propriedade para verificar o número de itens em um conjunto.

para de os loops iteram sobre coleções e geradores.

Destruição extrai dados de objetos e matrizes:

- **const {nome: ninjaName} =** ninja;
- **const [firstNinja] = ["** Yoshi "];

Módulos são unidades maiores de código de organização que nos permitem dividir programas em clusters:

```
exportar classe Ninja {}; // Exportar um item
exportação padrão class Ninja {} // Exportar exportação padrão {ninja}; // Exportar
exportação de variáveis   existentes {ninja como samurai}; // Renomear uma
exportação

importar Ninja de "Ninja.js"; // Importar uma importação de exportação padrão {ninja} de "Ninja.js"; //
Importar exportações nomeadas
importar * como Ninja de "Ninja.js"; // Importar todas as exportações nomeadas import {ninja como iNinja} de "Ninja.js"; //
Importar com um novo nome
```

Segredos do JavaScript Ninja Segunda edição

Resig . Bibeault . Maras

Junte-se a todo tipo de aplicativo, seja na web, no desktop, na nuvem ou em dispositivos móveis. Quando você se torna um profissional de JavaScript, você tem um conjunto de habilidades poderoso que pode ser usado em todos esses domínios.

Segredos do JavaScript Ninja, segunda edição usa prático exemplos para ilustrar claramente cada conceito e técnica principal. Esta edição completamente revisada mostra como fazer uma chave mestra Conceitos de JavaScript, como funções, closures, objetos, proto tipos e promessas. Abrange APIs como DOM, eventos, e temporizadores. Você descobrirá técnicas de práticas recomendadas, como teste e desenvolvimento em vários navegadores, todas ensinadas da perspectiva de profissionais qualificados de JavaScript.

O que há dentro

- Escrever código mais eficaz com funções, objetos e encerramentos
- Aprendendo a evitar armadilhas de aplicativos JavaScript
- Usando expressões regulares para escrever código de processamento de texto sucinto
- Gerenciando código assíncrono com promessas
- Totalmente revisado para cobrir os conceitos de ES6 e ES7

Você não precisa ser um ninja para ler este livro - apenas esteja disposto a se tornar um. Você está pronto?

John Resig é uma autoridade reconhecida em JavaScript e criadora da biblioteca jQuery. Bear Bibeault é um desenvolvedor web e autor da primeira edição, bem como co-autor de *Ajax na prática, protótipo e Scriptaculous em ação,* e *jQuery em ação.*
Josip Maras é pesquisador pós-doutorado e professor.

> "Excelente e abrangente visão sobre a magia das funções e fechamentos para o uso eficiente de JavaScript."
> - Gerd Klevesaat, Siemens

> "Leitura essencial para ... com técnicas poderosas para melhore o seu JavaScript."
> - Becky Huett, Big Shovel Labs

> "Ajuda você a dominar as técnicas furtivas e ousadas de JavaScript moderno."
> - Christopher Haupt
> New Relic Inc.

> "Habilidades essenciais para desenvolver suas habilidades em JavaScript para o próximo nível."
> - David Starkey, Blu m

Para baixar seu e-book gratuito nos formatos PDF, ePub e Kindle,
os proprietários deste livro devem visitar
manning.com/books/secrets-of-the-javascript-ninja-second-edition

ISBN-13: 978-1-61729-285-9
ISBN-10: 1-61729-285-0
54499
9 781617 292859

TRIPULAÇÃO

$ 44,99 / Can $ 51,99 [INCLUINDO E-LIVRO]