

INSTITUTO INFNET



TP 1

Javascript

Bloco: Desenvolvimento Front-End

Evan Imbeloni Rispoli Neto



Evan Imbeloni Rispoli Neto

TP1
JavaScript

Projeto realizado como requisito parcial avaliativo
- Teste de Performance - TP1 da disciplina de
JavaScript; do curso de Análise e
desenvolvimento de softwares pelo Instituto
INFNET da ESTI - Escola Superior de Tecnologia
da Informação.

Professor: Daniel de Oliveira



Verificação de conceitos.

05/02/2022

—

Evan Imbeloni Rispoli Neto -
Infnet - ESTI - Escola Superior de Tecnologia da Informação

1- Qual a diferença de uma linguagem script server-side e uma linguagem client-side? Cite um exemplo de cada.

As linguagens server-side são linguagens que são processadas pelo servidor e após o processamento, este enviará a resposta para o navegador. Podemos citar como exemplo de linguagem server-side o ASP, PHP e JSP(Java)

As linguagens cliente-side são aquelas que são processadas pelo browser e não o servidor. Como exemplo de linguagem client-side, podemos citar o Javascript, Typescript e Coffeescript.

2- Como funciona a arquitetura web de 3 camadas? Explique o funcionamento de cada uma das camadas. Cite, pelo menos, um exemplo de tecnologia ou aplicação que se utiliza em cada camada.

O servidor web é a camada de apresentação e fornece a interface do usuário. Geralmente, é uma página da Web ou um site da Web, como um site de comércio eletrônico em que o usuário adiciona produtos ao carrinho de compras, adiciona detalhes de pagamento ou cria uma conta. O conteúdo pode ser estático ou dinâmico, e geralmente é desenvolvido usando HTML, CSS e Javascript.

O servidor de aplicativos corresponde à camada intermediária, abrigando a lógica de negócios usada para processar as entradas do usuário. Para continuar o exemplo de comércio eletrônico, essa é a camada que consulta o banco de dados de inventário para retornar à disponibilidade do produto ou adiciona detalhes ao perfil de um cliente. Essa camada geralmente é desenvolvida usando Python, Ruby ou PHP e executa um framework como o Django, Rails, Symphony ou ASP.NET, por exemplo.

O servidor de banco de dados é a camada de dados ou backend de um aplicativo da web. Ele roda em softwares de gerenciamento de banco de dados, como MySQL, Oracle, DB2 ou PostgreSQL, por exemplo.

3- Por que estudar lógica de programação? Qual sua influência no desenvolvimento de software?

A lógica de programação é importante porque é ela quem nos dá as ferramentas necessárias para executar o processo mais básico no desenvolvimento de alguma aplicação: a criação de algoritmos.

4- Explique usando 2 exemplos diferentes de pseudocódigo o conceito de repetições e de condicionais. Qual o propósito de cada uma?

Muitas vezes, quando se escreve código, é desejável executar ações diferentes para decisões diferentes. Pode-se usar instruções condicionais em seu código para fazer isso. As instruções condicionais são usadas para executar diferentes ações com base em diferentes condições

Ex:

```
se (hora < 12){  
    saudacao= bom dia;  
}
```

Dentro da lógica de programação, a estrutura de repetição é uma estrutura que permite executar mais de uma vez o mesmo comando ou conjunto de comandos, de acordo com uma condição ou com um contador. São utilizadas, por exemplo, para repetir ações semelhantes que são executadas para todos os elementos de uma lista de dados, ou simplesmente para repetir um mesmo processamento até que a condição seja satisfeita.

Ex:

```
Enquanto (hora >=8 e hora <= 17){  
    Atividade= trabalho;  
}
```

5- Cite e explique os 4 passos para resolução de problemas computáveis.

Os 4 passos são:

- **Decomposição:** Processo que divide o problema em partes menores para facilitar a resolução.
- **Reconhecimento de padrões:** Identificar padrões é o processo que nos permite aplicar a mesma solução para problemas parecidos.
- **Abstração:** Trata-se da filtragem e classificação dos dados de maneira que nos permita separar somente os elementos essenciais em determinado problema
- **Algoritmo:** É um conjunto de instruções ordenadas que nos permite solucionar um problema.

6- Explique a importância das funções para a lógica de programação e a escrita de algoritmos

Funções são instruções bem definidas para executar uma tarefa específica a qualquer momento que seja necessário. Sendo assim as funções são de extrema importância para a lógica de programação, pois é o que nos permite a criação de algoritmos mais eficientes sem repetição de código.

Prática de lógica

Neste teste, você deverá concluir os exercícios executados em sala:

O Labirinto (The Maze)

O Artista #1 (The Artist)

O Artista #2 (The Artist)

O Artista #3 (The Artist)

O Artista #4 (The Artist)

E a sequência A Fazendeira deixada para o leitor:

A Fazendeira #1 (The Farmer)

A Fazendeira #2 (The Farmer)

A Fazendeira #3 (The Farmer)

Utilize seu cadastro no code.org, certifique-se que esteja logado e cadastrado na turma criada pelo professor e que tenha colocado um nome legível, conforme orientações da etapa 1. Tire também screenshots (prints de tela) e coloque no arquivo nome-do-aluno.tp1.zip.




Seções de Sala de aula

Junte-se a sala de aula do seu professor digitando o código de sessão do professor abaixo. Os professores serão capazes de ver o progresso do seu curso, e redefinir a sua senha caso você a esqueça.

Seção	Curso	Professor	Código da seção
Lógica, Computação e Algoritmos - Noites	Introdução acelerada para o curso de CC	Prof.	RDVDNP
18GRPADS01BFE201	Introdução acelerada para o curso de CC	Prof.	VNNGZF

Introdução acelerada para o curso de CC

Este curso de 20 horas apresenta o informática de núcleo e conceitos de programação. O curso é projetado para uso em salas de aula para os graus K-8, mas é divertido de aprender em todos os tempos.

Inscrito certificado	Obter apêlo	✓ Atribuído	
Nome da Lição	Progresso		
1. Introdução à Ciência Comput...	Atividade off-line		
2. O Labirinto	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20		
3. Pensamento computacional	Atividade off-line		
4. Programação de papel milim...	Atividade off-line		
5. O Artista	1 2 3 4 5 6 7 8 9 10		
6. Algoritmos	Atividade off-line		
7. O Artista 2	1 2 3 4 5 6 7 8 9 10 11		
8. Funções	Atividade off-line		
9. O Agricultor	1 2 3 4 5 6 7 8 9 10 11		
10. Condicionais	Atividade off-line		
11. O Artista 3	1 2 3 4 5 6 7 8 9 10 11		
12. escrita de canção	Atividade off-line		
13. O Agricultor 2	1 2 3 4 5 6 7 8 9 10		
14. Abstração	Atividade off-line		
15. O Artista 4	1 2 3 4 5 6 7 8 9 10		
16. Programação por revezamen...	Atividade off-line		
17. O Agricultor 3	1 2 3 4 5 6 7 8 9		
18. A Internet	Atividade off-line		
19. O Artista 5	1 2 3 4 5 6		
20. Resumo	Atividade off-line		