Avaliação Desenvolvedor

# Antes de começar!

Responda as questões conforme solicitado, sendo que as respostas podem ser feitas neste documento ou em arquivo(s) em anexo. Procure enviar as respostas o mais rápido possível.

**1 – Um professor de uma determinada disciplina aplicou uma prova valendo de 0 a 10, e gostaria de obter algumas informações. São elas:**

* Qual a média de nota da sala;
* Qual a média dos alunos do sexo masculino;
* Qual a média dos alunos do sexo feminino;
* Quantos alunos do sexo masculino tiveram nota 10;
* Quantos alunos do sexo feminino tiveram nota 10.

Desenvolva um **fluxograma ou algoritmo ou pseudocódigo** para determinar uma rotina que, após digitar todas as notas dos alunos, seja possível calcular e obter as informações acima, **não é para enviar código C#, Java ou Python ou qualquer outra linguagem famosa,** queremos ver isso em **fluxograma ou algoritmo ou pseudocódigo, importante nada de C# ou Java.**

**2 - Desenvolva um programa que simule a entrega de notas quando um cliente efetuar um saque em um caixa eletrônico.**

* Entregar o menor número de notas;
* É possível sacar o valor solicitado com as notas disponíveis;
* Saldo do cliente infinito;
* Quantidade de notas infinito;
* Notas disponíveis de R$ 100,00; R$ 50,00; R$ 20,00 e R$ 10,00.

Exemplos:

Valor do Saque: R$ 30,00 – Resultado Esperado: Entregar 1 nota de R$ 20,00 e 1 nota de R$ 10,00.

Valor do Saque: R$ 80,00 – Resultado Esperado: Entregar 1 nota de R$ 50,00, 1 nota de R$ 20,00 e 1 nota de R$ 10,00.

Neste exemplo, deverá ser utilizado o método MaquinaDinheiro.Sacar. Este método espera uma lista de objeto da classe Nota, conforme abaixo:

OBS: A classe MaquinaDinheiro é uma classe estática.

**OBS: não é para enviar código C#, Java ou Python ou qualquer outra linguagem famosa,** queremos ver isso em **fluxograma ou algoritmo ou pseudocódigo, importante nada de C# ou Java.**

**3 - Uma determinada empresa precisa desenvolver um módulo de sistema para montar um banco de dados para Pedidos. Essa base de dados deve estar normalizada e conter as seguintes informações:**

* Nome do fornecedor;
* CNPJ do fornecedor;
* Endereço do fornecedor, com bairro, cidade, cep e estado;
* Número e data e valor total do pedido;
* Itens do pedido, com quantidade e valor unitário.

1. Desenhe um diagrama de entidade x relacionamento (DER) a partir dos dados acima;
2. A partir do modelo de banco de dados definido no item anterior, desenhe a estrutura de uma aplicação CRUD utilizando o padrão MVC. (NÃO É NECESSÁRIO ESCREVER OS CÓDIGOS DAS CLASSES, SOMENTE A ESTRUTURA DAS PASTAS, AS CLASSES NECESSÁRIAS E OS MÉTODOS QUE COMPOEM AS CLASSES).

**4 – O que é e para que serve o conceito de “Interface”?**

A interface geralmente é usada como contrato de um método onde exige a implementação e na manutenção quando há mudança na assinatura do método.

**5 – O que está errado com a interface abaixo:**

public interface AlgoEstaErrado

{

void MethodA() {

Console.WriteLine("Hello World.");

}

}

R: Interface não tem um corpo no método, apenas assinatura. O ideial seria: void MetodoA();

**6 – Dado a classe abstrata abaixo:**

public abstract class EraUmGaroto

{

protected abstract string Regiao { get; }

public void Tocar()

{

Console.WriteLine("Era um garoto que como eu");

VersaoBandasFavoritas();

ViagemNoMundo();

Console.WriteLine("As coisas lindas da " + Regiao);

}

protected abstract void VersaoBandasFavoritas();

private void ViagemNoMundo()

{

Console.WriteLine("Girava o mundo sempre a cantar");

}

}

Implemente as seguintes versões:

Caipira:

Bandas Favoritas: “Amava Tonico e Tinoco e os Caipiras do Agreste”;

Região: “Inglaterra”;

Rock:

Bandas Favoritas: “Amava o Sepultura e os Cavalera”;

Região: “Argentina”;

**7 – Faça um programa que instancie as classes especialistas acima e chame o método Tocar. Imprima o resultado obtido.**

**8 – O que é ORM? Cite um exemplo.**

R: Object Relational Mapping, usado para construir um médio de campo entre a programação e o banco de dados, onde possibilita conectar a uma base de dados e trazer os resultados e popular em seus objetos da classe.

**9 - Refatore o método abaixo e explique o porquê das alterações.**

**Observação, neste exercício a parte comentada seria o código de impressão. Imagine que no comentário está a lógica para impressão.**

public class Votacao

{

private MyORM db = new MyORM();

private TRELib tre = new TRELib();

private UrnaLib urnaLib = new UrnaLib();

public static void AdicionarVoto(int numCandidato, string tituloEleitor)

{

var candidato = db.Candidatos.FirstOrDefault(x => x.Numero == numCandidato);

var eleitor = tre.Eleitor.Obter(tituloEleitor);

var resultadoVoto = tre.Voto.AdicionarVoto(candidato, eleitor);

var impressao = "";

/\* ATENÇÃO AQUI FICA A LÓGICA PARA A IMPRESSÃO QUE CONTÉM MAIS DE 300 LINHAS.....

\*

\*/

urnaLib.print(impressao);

}

}

**10 – Há um bug no exercício acima. Quem vota nulo não está tendo seu voto impresso. Faça a correção necessária baseado no código acima.**

**11 - Explique o porquê de uma transação que remove/edita/inclui um registro não pode ser GET.**

R: Porque o método GET ele não altera o estado.

**Dado o código abaixo:**

<p>

Lorem ipsum vitae himenaeos vulputate ut himenaeos vehicula senectus, placerat mi in iaculis inceptos iaculis.

</p>

<p>

Lorem ipsum vitae himenaeos vulputate ut himenaeos vehicula senectus, placerat mi in iaculis inceptos iaculis.

</p>

**12 – No código acima, como faço para atribuir um tamanho de 40px para a primeira letra dos parágrafos?**

**13 – No código acima pinte a fonte do segundo parágrafo de verde.**

**14 – Faça uma declaração de uma media query no CSS.**

**15 – Umas das vantagens do Jquery é a grande extensão de funções que podemos contar de forma gratuita e a integração com vários browsers. Como seria o script em Jquery para pintar as linhas pares de uma tabela de cor azul e as ímpares de cor cinza?**

1. $("tr:even").css("background", "blue");  
   $("tr:odd").css("background", "gray");
2. $("td:even").attr("background", "blue");  
   $("td:odd").attr("background", "gray");
3. $("table td:even").css("background", "blue");  
   $("table td:odd").css("background", " gray");
4. $.each(rows, function(índex, item){ rows[índex].background = índex % 2 === 0 ? “blue” : “gray”; })

**16 – Usando Bootstrap, para alinhar os controles dentro de uma div a direita deverei usar o seguinte código:**

1. ( ) style=”text-align: right”
2. ( ) class=”text-align: right”
3. ( ) style=”margin-right: 0px”
4. ( ) class=”float-right”

**17 - De acordo com o princípio da inversão de dependência, módulos de alto nível não devem depender de módulos de baixo nível, ambos devem depender de abstrações. Abaixo temos um exemplo de violação, com exemplo do Interruptor e da Luminária, onde ambas são classes concretas:**

public class Interruptor

{

private Luminaria\_Luminaria;

public void Acionar()

{ if (condicao)

\_Luminaria.Ligar();

}

}

**18 – Transcreva com suas palavras as instruções SQL abaixo:**

1. UPDATE C SET dataCancelamento = DATEADD(MONTH,1,GETDATE()) FROM contratos C INNER JOIN Fornecedores F ON F.id\_fornecedor = C.id\_fornecedor WHERE EXISTS (SELECT 0 FROM pedidos p WHERE p.id\_fornecedor = C.id\_fornecedor AND P.dataParaEntrega < GETDATE() AND P.dataEntrega IS NULL)
2. SELECT \* FROM Pedidos P INNER JOIN Itens I LEFT OUTER JOIN Estoque E ON E.id\_Item = I.id\_item ON I.id\_pedido = P.id\_pedido WHERE E.id\_item IS NULL
3. DELETE FROM Atletas WHERE NOT EXISTS (SELECT 0 FROM AtletasCampeonatos WHERE Atletas.Codigo\_Atleta = AtletasCampeonatos.Codigo\_Atleta) OR Atletas.Codigo\_Clube IS NULL

**19 – Transcreva as instruções T-SQL abaixo para LINQ TO SQL:**

1. SELECT Nome,Idade,Sexo FROM Atletas WHERE YEAR(DataNascimento) = 1995 AND MONTH(DataNascimento) > 03
2. SELECT Atletas.Nome,Atletas.Idade,Clube.NomeDoClube FROM Atletas INNER JOIN Clubes ON Atletas.codigo\_codigo = Clubes.codigo\_clube WHERE Atletas.Sexo = ‘F’ and Clubes.UF = ‘SP’

**20 – Responda as questões abaixo:**

1. Uma determinada página web possui três elementos html <select>, na seguinte sequência:

Lista de Estados;

Lista de Municípios;

Lista de Bairros.

Ao selecionar o Estado, a Lista de Municípios deve ser carregada com os municípios daquele Estado. Ao selecionar o Município, a Lista de Bairros deve ser carregada com os bairros daquele Município.

Como você faria para desenvolver essa rotina? Quais tecnologias e recursos utilizaria, no lado cliente e no lado servidor?

1. Qual é a sintaxe correta e qual é o valor final de i:

( ) for (I = 0, I <= 5, i++)

( ) for ( i <= 5; i++)

( ) for i = 1 to 5

( x ) for (I = 0; I <= 5; i++)

( ) for (i = 0; <= 5)

( ) Tenho minha própria resposta que é:

1. Crie um Object usando Javascript que contenha:
   1. marca,
   2. modelo,
   3. cor,
   4. ano de fabricação.

const fabrica = {

marca:"Honda",

modelo:"civic",

cor:"branco",

ano: “2022”

};

1. Qual a melhor maneira de garantir a segurança, versionamento e colaboração em um projeto de desenvolvimento de software? De um exemplo das tecnologias a serem adotadas e como funcionariam.

**21 - A função abaixo não possui qualquer documentação. Pela análise do código do método é possível dizer que esta função:**

return (

from p in context.Pagamentos join c in context.Clientes on p.IdCliente equals c.IdCliente where c.Ativo == 1 && p.Vencimento < DateTime.Now.Date.AddDays(-10) && p.ValorPago == 0   
select new { Cliente = c.Nome + $“{c.Status}“, Parcela = p.Parcela, Vencimento = p.Vencimento, Valor = p.Valor })  
.ToList()  
.OrderByDescending(o=>o.Valor)  
.Sum(t=>t.Valor);

1. ( ) Lista todos os clientes que fizeram seus pagamentos até a data de hoje e que possuem ZERO como o valor em aberto;
2. ( ) Ordena todas as parcelas vencidas, em valor crescente, para clientes inativos e que encerraram suas contas com algum valor em aberto;
3. ( ) Lista todas as parcelas que se venceram nos últimos 10 dias, o nome do cliente concatenado ao seu status, o número da parcela, a data do vencimento e o valor ordenando-as de forma decrescente;
4. ( ) Soma o valor das parcelas dos clientes ativos que se venceram nos últmos 10 dias e que ainda não foram pagas.
5. ( ) Tenho minha propria resposta que seria :  
   \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
   \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**22 - A API Produtos contém uma funcionalidade para adicionar comentários a um produto específico. Qual chamada REST melhor representa a inclusão de novos comentários ao produto de código 100?**

1. ( ) ../produtos/{código do produto}/add-comentario/{id do comentário} -> VERB: PUT
2. ( ) ../produtos/add-comentario/{código produto}/{código comentário } -> VERB: GET
3. ( ) ../produtos/{código produto}/comentario -> VERB: POST
4. ( ) ../produtos/{código produto}/comentario/ -> VERB: PATCH
5. ( x ) ../comentario/{código produto}/add -> VERB: PUT

**23 – O Código abaixo muda o status de um cliente para ativo e abre uma ocorrência no livro de registros, O LOG DE ERROS DA APLICAÇÃO contém as entradas abaixo. Aponte as melhorias que se adéquam as ocorrências do log:**

* Null Reference!
* Sql Validation exception! Status must be “A”or “I”
* Entity validation! Field Ocorrencia can not be null

public virtual ChangeClienteStatus(Cliente c, string status, string ocorrencia) {

var cliente = Context.Clientes.Get(c.Id);

cliente.Status = status;

cliente.Ocorrencias.Add(new Ocorrencia{ Text = ocorrencia, Resp = User.Name; });

Context.SaveChanges();

}

1. ( ) A primeira ocorrência, Null Reference, será completamente eliminada se o objeto Context for validado antes de ser usado. Ex: (if Context != null). Também será necessário validar o conteúdo de status usando if (status.lenght > 0).
2. ( ) Uma exception customizada poderia ser gerada nos casos em que (1) o parâmetro “Cliente c ” for nulo (2) a função string.IsNullOrEmpty retornar true para os parâmetros status e ocorrencia. Estas validações são suficientes para todos os erros apontados no LOG.
3. ( x) Null Reference poderá ser resolvida validando: (1) se o parâmetro “c” for diferente de nulo (2) se a variável cliente (resultado da busca pelo ID) for diferente de nulo (3) para o status verificar se a expressão {“A”, “I”}.Contains(status??string.empty) e (4) para ocorrência a função string.isNullOrEmpty será suficiente.
4. ( ) A mensagem Null Reference é uma exceção interna do .NET Framework e não há como ser tratada. Já as mensagens de banco deverão ser tratadas diretamente pelo próprio gerenciador de banco de dados, neste caso o SQL SERVER.
5. ( ) Tenho minha própria resposta que seria:   
   \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**24 – Elabore uma aplicação que faça cadastro de cliente com nome, CPF e endereço e que liste este conteúdo em um grid, esta aplicação deve ser feita em .net core, de preferência .net core 6 com a arquitetura que você achar mais ideal, o front-end deve ser feito em angular também com arquitetura que que você ache mais adequada. Suba essa aplicação no GitHub e nos passes o endereço para que possamos fazer um fork ou download para avaliação!**

**Obs:** Pense bem nas suas escolhas para cada item da arquitetura, pois será avaliado internamente por outros desenvolvedores que podem querer fazer perguntas sobre alguns pontos.