

NOME:

CPF:

Assinatura: \_\_\_\_\_

**Questão 1**

Valor da questão: 0,33

Em C#, qual das seguintes opções descreve corretamente a finalidade do modificador `private` para membros de uma classe?

- a) Torna os membros acessíveis em todas as classes derivadas.
- b) Torna os membros acessíveis em qualquer lugar dentro do mesmo namespace.
- c) Torna os membros acessíveis apenas dentro da mesma classe.
- d) Torna os membros acessíveis apenas dentro do mesmo assembly.
- e) Torna os membros acessíveis globalmente em todo o código-fonte.

**Questão 2**

Valor da questão: 0,33

Qual das seguintes afirmações é verdadeira em relação a interfaces e classes abstratas em C#?

- a) Uma classe abstrata pode herdar de múltiplas interfaces, enquanto uma interface não pode herdar de outra interface.
- b) Uma interface pode conter campos e construtores, enquanto uma classe abstrata não pode conter esses membros.
- c) Uma classe abstrata pode implementar métodos, enquanto uma interface não pode.
- d) Uma classe abstrata pode ser instanciada diretamente, enquanto uma interface não pode ser instanciada.
- e) Uma classe abstrata pode fornecer implementações para todos os seus membros, enquanto uma interface não pode fornecer implementações.

**Questão 3**

Valor da questão: 0,33

Qual dos seguintes elementos HTML é considerado um elemento de bloco (block-level)?

- a) `<span>`
- b) `<nav>`
- c) `<strong>`
- d) `<a>`
- e) `<img>`

**Questão 4**

Valor da questão: 0,33

Qual das seguintes opções descreve corretamente a função da propriedade `padding` em CSS no modelo de caixa (box model) de um elemento HTML?

- a) Define a área ao redor do conteúdo de um elemento.
- b) Define a espessura da borda de um elemento.
- c) Define o espaço entre as margens de um elemento e seus elementos filhos.
- d) Define o espaço entre a borda de um elemento e seu conteúdo.
- e) Define o espaço entre o conteúdo de um elemento e o conteúdo de seus elementos irmãos.

**Questão 5**

Valor da questão: 0,33

Analise o trecho do programa JS e indique os valores de `w` e `z`, respectivamente

```
const f = (a, b) => a * b;  
let x = 7;  
let y = 6;  
const w = f;  
const z = f(x, y);
```

- a) O valor de `w` é `f`  
O valor de `z` é `42`
- b) O valor de `w` é `(a, b) => a * b`  
O valor de `z` é `42`
- c) O valor de `w` é `const`  
O valor de `z` é `42`
- d) O valor de `w` é `42`  
O valor de `z` é `13`
- e) O valor de `w` é `42`  
O valor de `z` é `(a, b) => a * b`

**Questão 6**

Valor da questão: 0,33

No padrão de projeto Observer, qual é o papel desempenhado pelo objeto "Subject"?

- a) O objeto que representa uma instância de uma classe.
- b) O objeto que fornece uma interface comum para todos os produtos criados.
- c) O objeto que encapsula como os objetos interagem.
- d) O objeto que mantém uma lista de objetos dependente (observers) e notifica automaticamente esses objetos sobre qualquer mudança de estado.
- e) O objeto que define a interface para criar objetos.

**Questão 7**

Valor da questão: 0,33

Quais dos seguintes são benefícios do padrão de projeto Proxy?

- a) Aumento da complexidade do código, pois introduz uma camada adicional entre o cliente e o objeto real.
- b) Diminuição do desempenho, já que o proxy adiciona overhead devido à sua camada intermediária.
- c) Redução da flexibilidade, pois o proxy limita as operações disponíveis no objeto real.
- d) Melhoria da segurança, pois o proxy pode controlar o acesso ao objeto real, impondo restrições adicionais.
- e) Facilitação da implementação de caching, permitindo que o proxy mantenha uma cópia em cache do objeto real para evitar acesso frequente ao mesmo.

**Questão 8**

Valor da questão: 0,33

Quais dos seguintes relacionamentos não são possíveis em um diagrama de casos de uso da UML?

- a) Generalização
- b) Agregação
- c) Inclusão
- d) Associação
- e) Extensão

**Questão 9**

Valor da questão: 0,33

Quais dos seguintes são componentes comuns encontrados em um diagrama de sequência da UML?

- a) Classes, atributos e métodos.
- b) Objetos, mensagens e lifelines (linhas de vida).
- c) Pacotes, interfaces e implementações.
- d) Atividades, estados e transições.
- e) Atores, casos de uso e relacionamentos.

**Questão 10**

Valor da questão: 0,33

Qual dos seguintes é um dos objetivos do diagrama de implantação (deployment) da UML?

- a) Descrever a estrutura estática de um sistema, mostrando com as classes do sistema e suas relações são implementadas.
- b) Organizar e estruturar elementos de modelagem em grupos lógicos para facilitar a compreensão e a manutenção do sistema.
- c) Mostrar as etapas e decisões tomadas durante a execução de um processo dentro de um sistema.
- d) Representar a arquitetura física de um sistema, mostrando como os componentes do sistema são implantados em nós de hardware.
- e) Ilustrar a sequência de mensagens trocadas entre os objetos de um sistema ao longo do tempo.

**Questão 11**

Valor da questão: 0,33

Ao implementar um modelo de dados para uma aplicação .NET Core MVC que gerencia informações de funcionários, qual tipo de relacionamento seria mais apropriado entre as entidades "Funcionário" e "Departamento" se um funcionário pode pertencer a apenas um departamento? E ao implementar os mapeamentos objeto-relacional para essas entidades, qual seria a maneira correta de configurar o relacionamento entre elas?

- a) Relacionamento de um para muitos (one-to-many) - configurar o relacionamento no método OnModelCreating do DbContext usando o método HasMany para a entidade "Funcionário" e o método WithOne para a entidade "Departamento".
- b) Relacionamento semântico - Configurar o relacionamento no método OnModelCreating do DbContext usando o método HasMany para a entidade "Departamento" e o método HasOne para a entidade "Funcionário".
- c) Relacionamento de muitos para um (one-to-many) - Configurar o relacionamento no método OnModelCreating do DbContext usando o método HasMany para a entidade "Departamento" e o método WithOne para a entidade "Funcionário".
- d) Relacionamento de um para um (one-to-one) - utilizar a anotação [Key] na propriedade correspondente ao ID do departamento na entidade "Funcionário" e a anotação [ForeignKey] na propriedade correspondente ao ID do departamento na entidade "Departamento".
- e) Relacionamento de muitos para muitos (many-to-many) - Utilizar a anotação [Key] na propriedade correspondente ao ID do departamento na entidade "Funcionário" e a anotação [ForeignKey] na propriedade correspondente ao ID do departamento na entidade "Departamento".

**Questão 12**

Valor da questão: 0,33

Em uma aplicação .NET Core de blog, você está implementando uma funcionalidade para exibir uma lista de postagens recentes, juntamente com os comentários associados a cada postagem. Esta listagem será exibida de maneira paginada e cada página contém até 100 postagens. Dado que a página inicial contendo as 100 postagens mais recentes deve ser carregada rapidamente e que há milhares de postagens e comentários no banco de dados, qual estratégia de carregamento de dados mais apropriada para recuperar as postagens com seus comentários de maneira a evitar a recuperação de mais dados do que o necessário para a página inicial?

- a) Eager loading
- b) Lazy loading
- c) Intermediate loading
- d) Advanced loading
- e) Deferred loading

**Questão 13**

Valor da questão: 0,33

No Entity Framework Core, qual é o objetivo do DbContext?

- a) Controlar a autenticação e autorização dos usuários.
- b) Definir configurações de roteamento da aplicação.
- c) Gerenciar a conexão com o banco de dados e operações CRUD.
- d) Fornecer acesso direto aos comandos SQL.
- e) Mapear classes de entidade (objetos .NET) para arquivos JSON.

**Questão 14**

Valor da questão: 0,33

Qual é a abordagem comum para conectar a camada controladora de um aplicativo .NET Core com o banco de dados?

- a) Criação de consultas diretas ao banco de dados usando ADO.NET.
- b) Utilização de um ORM (Object-Relational Mapping) como Entity Framework Core.
- c) Implementação direta de consultas SQL em cada método da camada controladora.
- d) Utilização exclusiva de arquivos JSON para armazenar dados.
- e) Uso de protocolos de comunicação como FTP para transferência de dados.

**Questão 15**

Valor da questão: 0,33

Qual é a vantagem principal do uso do Entity Framework Core em relação à conexão direta do aplicativo .NET Core com o banco de dados?

- a) Melhoria significativa na segurança dos dados.
- b) Maior controle sobre a otimização de consultas.
- c) Redução do acoplamento entre o código da aplicação e o esquema do banco de dados.
- d) Facilidade de escrever consultas SQL complexas.
- e) Aumento da eficiência na transferência de dados.

**Questão 16**

Valor da questão: 0,33

No contexto da normalização de banco de dados, qual das seguintes afirmações é verdadeira sobre a Terceira Forma Normal (3FN)?

- a) A Terceira Forma Normal (3FN) permite a eliminação de redundâncias de dados ao garantir que cada relação seja representada por uma única chave estrangeira.
- b) A Terceira Forma Normal (3FN) permite a eliminação de redundâncias de dados ao garantir que cada atributo não chave seja totalmente dependente da chave primária.
- c) A Terceira Forma Normal (3FN) permite a eliminação de redundâncias de dados ao garantir que cada atributo seja funcionalmente dependente de todos os outros atributos da tabela.
- d) A Terceira Forma Normal (3FN) permite a eliminação de redundâncias de dados ao garantir que cada atributo seja funcionalmente dependente apenas da chave primária e não de outros atributos não chave.
- e) A Terceira Forma Normal (3FN) permite a eliminação de redundâncias de dados ao garantir que cada relação seja representada por uma chave primária composta.

**Questão 17**

Valor da questão: 0,33

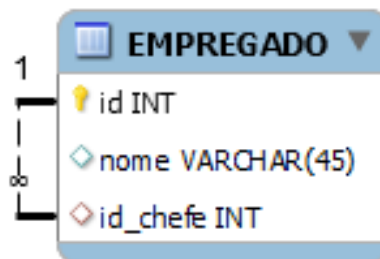
Na criação de tabelas utilizando SQL, quando usamos ON DELETE e ON UPDATE, em geral, que opção não pode ser utilizada?

- a) SET DEFAULT
- b) CASCADE
- c) SET NULL
- d) RESTRICT
- e) ACTION

**Questão 18**

Valor da questão: 0,33

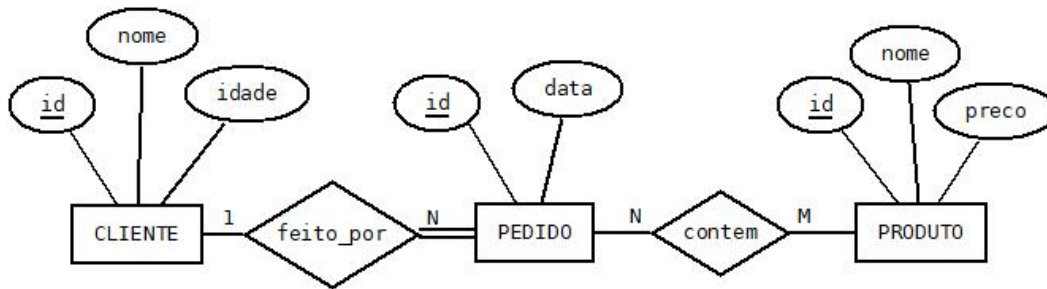
Considere o seguinte modelo relacional:



Qual das seguintes consultas SQL retorna, para todos os empregados, o nome do empregado e o nome de seu chefe imediato, caso o empregado tenha chefe imediato, ou NULL, caso o empregado não tenha chefe imediato.

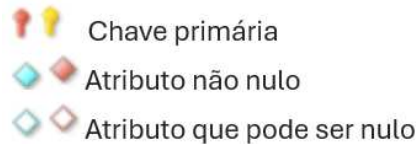
- a) `SELECT E.nome AS nome_empregado, S.nome AS nome_chefe FROM (EMPREGADO AS E JOIN EMPREGADO AS S ON E.id_chefe=S.id);`
- b) `SELECT E.nome AS nome_empregado, S.nome AS nome_chefe FROM (EMPREGADO AS E CROSS JOIN EMPREGADO AS S);`
- c) `SELECT E.nome AS nome_empregado, S.nome AS nome_chefe FROM (EMPREGADO AS E FULL OUTER JOIN EMPREGADO AS S ON E.id_chefe=S.id);`
- d) `SELECT E.nome AS nome_empregado, S.nome AS nome_chefe FROM (EMPREGADO AS E LEFT OUTER JOIN EMPREGADO AS S ON E.id_chefe=S.id);`
- e) `SELECT E.nome AS nome_empregado, S.nome AS nome_chefe FROM (EMPREGADO AS E RIGHT OUTER JOIN EMPREGADO AS S ON E.id_chefe=S.id);`

Considere o seguinte modelo entidade-relacionamento (ER):



Agora, imagine que você precisa mapear esse modelo ER para um modelo relacional. Qual dos seguintes modelos seria o mais apropriado?

Considere a seguinte notação nos modelos abaixo:



- a)
 

<b>CLIENTE</b> id INT (PK) nome VARCHAR(45) idade INT	1		<b>PEDIDO</b> id INT (PK) data DATE feito_por INT	1		<b>contem</b> PEDIDO_id INT (FK) PRODUTO_id INT (FK)	∞		<b>PRODUTO</b> id INT (PK) nome VARCHAR(45) preco DOUBLE
	∞	∞		∞	∞		1		
- b)
 

<b>CLIENTE</b> id INT (PK) nome VARCHAR(45) idade INT	1		<b>PEDIDO</b> id INT (PK) data DATE feito_por INT	1		<b>contem</b> PEDIDO_id INT (FK) PRODUTO_id INT (FK)	∞		<b>PRODUTO</b> id INT (PK) nome VARCHAR(45) preco DOUBLE
	∞	∞		∞	∞		1		
- c)
 

<b>CLIENTE</b> id INT (PK) nome VARCHAR(45) idade INT feito_por INT	1		<b>PEDIDO</b> id INT (PK) data DATE	1		<b>contem</b> PEDIDO_id INT (FK) PRODUTO_id INT (FK)	∞		<b>PRODUTO</b> id INT (PK) nome VARCHAR(45) preco DOUBLE
	∞	∞		∞	∞		1		
- d)
 

<b>CLIENTE</b> id INT (PK) nome VARCHAR(45) idade INT	1		<b>PEDIDO</b> id INT (PK) data DATE feito_por INT	1		<b>contem</b> PEDIDO_id INT (FK) PRODUTO_id INT (FK)	∞		<b>PRODUTO</b> id INT (PK) nome VARCHAR(45) preco DOUBLE
	∞	∞		∞	∞		1		
- e)
 

<b>CLIENTE</b> id INT (PK) nome VARCHAR(45) idade INT	1		<b>PEDIDO</b> id INT (PK) data DATE feito_por INT	1		<b>contem</b> PEDIDO_id INT (FK) PRODUTO_id INT (FK)	∞		<b>PRODUTO</b> id INT (PK) nome VARCHAR(45) preco DOUBLE
	∞	∞		∞	∞		1		

**Questão 20**

Valor da questão: 0,33

Com relação aos conceitos de teste de software, qual das seguintes afirmações é verdadeira?

- a) Os teste de usabilidade identificam vulnerabilidades e falhas de segurança, garantindo que o software possa ser usado apenas por usuários finais autorizados.
- b) Os testes de desempenho avaliam apenas o tempo que o sistema leva para responder a uma determinada solicitação do usuário.
- c) Os testes de regressão verificam se as alterações feitas no software não causam regressões ou quebras de funcionalidades existentes, garantindo que o sistema continue funcionando conforme esperado após modificações
- d) Os testes de aceitação são realizados pelos desenvolvedores para garantir que o software atenda aos requisitos do cliente.
- e) Os testes de unidade verificam se o software funciona corretamente como um todo, integrando todas as partes do sistema.

**Questão 21**

Valor da questão: 0,33

Qual das seguintes afirmações descreve com precisão os conceitos de testes de caixa preta e caixa branca?

- a) O teste de caixa branca não requer conhecimento interno do código, enquanto o teste de caixa preta requer uma compreensão detalhada da lógica interna do software.
- b) No teste de caixa branca, os casos de teste são derivados da análise da lógica interna do código, enquanto no teste de caixa preta os casos de teste são baseados apenas nas especificações externas do software.
- c) O teste de caixa branca é focado na funcionalidade visível do software, enquanto o teste de caixa preta é focado em garantir que o código seja implementado corretamente.
- d) O teste de caixa branca é conduzido pelos desenvolvedores do software, enquanto o teste de caixa preta é conduzido pelos usuários finais.
- e) O teste de caixa preta examina a estrutura interna do código-fonte, enquanto o teste de caixa branca avalia a funcionalidade externa do software.

**Questão 22**

Valor da questão: 0,33

A Activity é o componente do Android responsável por mostrar telas ao usuário e permitir que ele interaja com essas telas através de toques e cliques. A Activity possui um ciclo de vida bem definido que permite ao programador adicionar ações em momentos específicos deste ciclo de vida. Sabendo disso, escolha a opção que contém apenas métodos presentes no ciclo de vida da Activity.

- a) onCreate(), onInit(), onPause(), onDestroy()
- b) onCreate(), onInit(), onResume(), onFinishing()
- c) onCreate(), onStart(), onResume(), onDestroy()
- d) onCreate(), onInit(), onResume, onDestroy ()
- e) onCreate(), onStart(), onPause(), onFinishing()

**Questão 23**

Valor da questão: 0,33

Ao criar aplicativos para dispositivos Android, é crucial definir o tamanho dos componentes na tela para proporcionar uma experiência de usuário eficaz. O Android oferece diversas constantes que auxiliam nesse dimensionamento relativo, adaptando os componentes a diferentes tamanhos e densidades de tela. Sobre tais constantes assinale a alternativa correta:

- a) A constante wrap\_content define um tamanho fixo para o componente, impedindo que ele seja redimensionado de acordo com as mudanças na tela ou no conteúdo.
- b) A constante wrap\_content indica que a View deve se redimensionar de modo a ocupar apenas o tamanho que seus elementos precisam para serem exibidos.
- c) A constante wrap\_content indica que a View deve ocupar a metade da área ocupada pelo seu elemento pai.
- d) A constante wrap\_content indica que a View deve ocupar a mesma área ocupada pelo seu elemento pai.
- e) A constante wrap\_content indica que a View deve ocupar toda área disponível.

**Questão 24**

Valor da questão: 0,33

O sistema operacional Android é uma das plataformas mais populares para dispositivos móveis em todo o mundo, impulsionando milhões de smartphones, tablets e outros dispositivos. Sua arquitetura, cuidadosamente elaborada, é fundamental para oferecer uma experiência de usuário consistente e eficiente, além de proporcionar um ambiente propício para o desenvolvimento de aplicativos inovadores e poderosos. Nesse contexto, e sobre o Android e sua arquitetura, assinale a alternativa incorreta:

- a) Possui quatro grandes componente: Activity, Service, Content Provider e Multicast Receiver
- b) Baseado no sistema Linux
- c) Executa cada aplicativos em uma máquina virtual dedicada
- d) Possui bibliotecas nativas implementadas em C/C++
- e) Utilize Android Runtime (ART) para execução de aplicativos

**Questão 25**

Valor da questão: 0,33

Sobre a arquitetura em camada MVC (Model-View-Controller) analise as sentenças abaixo e assinale a alternativa correta:

- I - A camada *Controller* é responsável por controlar o layout, o estilo e os elementos de design apresentados ao usuário final.
- II - A camada *View* interage diretamente com a camada Model
- III - A camada *Model* é responsável por representar a estrutura dos dados da aplicação, incluindo suas regras de negócio, validações e operações de acesso ao banco de dados.
- IV - A camada *Controller* coordena as interações entre o Modelo e a View, interpretando as requisições do usuário e atualizando a View conforme necessário.

- a) Apenas IV
- b) Apenas III e IV
- c) Apenas I, III
- d) Apenas II e III
- e) Apenas I e II

**Questão 26**

Valor da questão: 0,33

Na arquitetura MVC (*Model-View-Controller*) para sistemas web, a camada de Modelo desempenha um papel crucial na manipulação e organização dos dados da aplicação. Qual das seguintes opções descreve melhor uma função típica da camada de Modelo?

- a) Receber requisições do cliente e retornar respostas HTML.
- b) Controlar o fluxo de navegação entre as diferentes páginas da aplicação.
- c) Interagir diretamente com o banco de dados para recuperar, inserir, atualizar ou deletar informações.
- d) Determinar a aparência visual da interface do usuário.
- e) Validar os dados inseridos pelo usuário em formulários web.

**Questão 27**

Valor da questão: 0,33

No contexto de um framework MVC (*Model-View-Controller*) para desenvolvimento web, qual das seguintes ações **não** é realizada pela camada de Controle?

- a) Validar os dados recebidos do usuário.
- b) Manipular as requisições do cliente.
- c) Realizar operações de consulta no banco de dados.
- d) Atualizar o modelo com base nas interações do usuário.
- e) Decidir qual visão deve ser renderizada.

**Questão 28**

Valor da questão: 0,33

Para a sincronização do repositório local em relação a um repositório remoto, o programador optou por utilizar a operação de `git pull`, em um sistema que utiliza a ferramenta Git como controle de versão. Sabendo disso, assinale a alternativa que indica a sequência de comandos que obtém o mesmo resultado final de um `git pull`.

- a) `git fetch` seguido por `git rebase HEAD`
- b) `git fetch` seguido por `git merge FETCH_HEAD`
- c) `git clone` seguido por `git checkout master`
- d) `git status` seguido por `git diff` e então `git push`
- e) `git fetch` seguido por `git add`.

**Questão 29**

Valor da questão: 0,33

No contexto do sistema de controle de versão Git, assinale a alternativa que indica o conceito de "fast forward" que ocorre durante a mescla de dois *branchs* através do `git merge`?

- a) O "fast forward" ocorre quando o Git não pode mesclar automaticamente as alterações devido a conflitos, exigindo uma intervenção manual do usuário.
- b) O "fast forward" ocorre quando é utilizado o método *rebase na operação de mesclagem de dois branchs*. Nesse caso, o `git` reprocesa todos os *commits* no *branch* de origem no *branch* alvo.
- c) O "fast forward" ocorre quando o histórico de *commits* do *branch* alvo não possui divergências em relação ao *branch* que está sendo mesclado, permitindo que o Git simplesmente avance o ponteiro do *branch* sem criar um novo *commit* de mesclagem.
- d) O "fast forward" ocorre quando o Git reescreve o histórico de *commits* do *branch* que está sendo mesclado para coincidir exatamente com o histórico do *branch* alvo, garantindo uma integração suave e linear das alterações.
- e) O "fast forward" ocorre quando o Git cria um novo *commit* de mesclagem que incorpora automaticamente as alterações do *branch* que está sendo mesclado, sem levar em consideração a linha do tempo do histórico de *commits*.

**Questão 30**

Valor da questão: 0,33

Sobre o sistema de controle de versão Git e o processo de *commit*, assinale a alternativa correta.

- a) O *commit* realiza o registro no repositório remoto de uma ou mais mudanças presentes no *work directory*.
- b) O *commit* realiza o *snapshot* do repositório, copiando todos os arquivos do *work directory* para a pasta `.git`.
- c) O *commit* realiza o registro no repositório local de uma ou mais mudanças presentes no *staging area*.
- d) O *commit* realiza o registro no *staging area* de uma ou mais mudanças realizadas no diretório de trabalho (*work directory*).
- e) O *commit* realiza o registro em repositório remoto de uma ou mais mudanças realizadas no *staging area* de um determinado *branch*.