|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ALUNO** |  | | | **MATRÍCULA** |  |
| **DISCIPLINA** |  | | | **DATA DA PROVA** |  |
| **PROFESSOR** |  | | | **NOTA** |  |
| **TURMA** |  | **TEMPO DE PROVA** | 60min |

ATENÇÃO:

Não esquecer de preencher o nome completo na folha das questões;

A avaliação somente poderá ser entregue depois de decorridos 50min de seu início;

A prova deverá ser realizada sem nenhum tipo de consulta e uso de caneta esferográfica azul ou preta. Provas entregues escritas a lápis **NÃO** serão corrigidas;

Será atribuída nota zero ao aluno que devolver sua prova em branco, independentemente de ter assinado a Ata de Prova;

Ao aluno flagrado **utilizando meios ilícitos ou não autorizados pelo professor para responder a avaliação** será atribuída nota zero e, mediante representação do professor, responderá a Procedimento Administrativo Disciplinar, com base no Regimento Interno desta IES; e,

A interpretação das questões faz parte da prova. Leia CUIDADOSAMENTE cada uma das questões. Não se precipite.

**GABARITO**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **QUESTÃO** | **A** | **B** | **C** | **D** | **E** |
| **03.** | O | O | O | O | O |
| **04.** | O | O | O | O | O |
| **05.** | O | O | O | O | O |
| **06.** | O | O | O | O | O |
| **07.** | O | O | O | O | O |
| **08.** | O | O | O | O | O |
| **09.** | O | O | O | O | O |
| **10.** | O | O | O | O | O |

Orientações para questões **DISCURSIVAS:**

* Seu texto deve ser dissertativo-argumentativo (não deve, portanto, ser escrito em forma de poema ou de narração);
* Seu texto deve ser redigido na modalidade escrita padrão da Língua Portuguesa;
* O texto deve ter entre 5 e 10 linhas;
* Serão considerados na correção: Ortografia; Gramática; Linguagem culta.

**QUESTÃO 1 (1,0) –** Considere um cenário em que você está desenvolvendo um sistema de gerenciamento de uma escola. Na modelagem desse sistema, você identificou duas classes principais: **Pessoa** e **Aluno**. A classe **Pessoa** representa todas as pessoas envolvidas na escola, como alunos, professores e funcionários, enquanto a classe **Aluno** representa especificamente os alunos matriculados na escola.

Você decidiu implementar a herança na sua modelagem, de modo que a classe **Aluno** herde os atributos e métodos da classe **Pessoa**. No entanto, durante o desenvolvimento, você percebeu que existem atributos e métodos específicos que se aplicam apenas aos alunos e não a outras pessoas na escola.

Nesse contexto, implemente em **TypeScript** como você utilizaria a herança na modelagem do sistema de gerenciamento da escola e forneça um exemplo de atributo ou método específico da classe Aluno que não se aplicaria à classe Pessoa, demonstrando assim a vantagem da herança nesse cenário. (use o verso da página para responder)

**QUESTÃO 2 (1,0) –** Considere o desenvolvimento de um sistema para uma companhia aérea que gerencia reservas de voos. Nesse sistema, você identificou duas classes principais: **Voo** e **Passageir**o.

A classe **Voo** representa um voo específico, com um número de voo, origem, destino, data e hora de partida, capacidade máxima de passageiros e lista de passageiros reservados. A classe **Passageiro** representa um passageiro que deseja fazer uma reserva em um voo. Cada passageiro tem um nome, número de documento, e-mail e telefone de contato.

Como você modelaria este sistema utilizando o **TypeScript**? (use o verso da página para responder)

**Questão 3 (1,0) -** **Qual é a forma correta de declarar uma classe em TypeScript?**

1. define class MinhaClasse { }
2. Class MinhaClasse { }
3. class = MinhaClasse { }
4. new MinhaClasse { }
5. class MinhaClasse { }

**Questão 4 (1,0) - Como você declararia uma classe chamada Produto para que ela possa ser exportada e utilizada em outros arquivos em TypeScript?**

1. Produto { }
2. export Produto;
3. export class Produto { }
4. Produto = export { };
5. class Produto { export };

**Questão 5 (1,0) - Qual é a função principal de um construtor em uma classe TypeScript?**

1. Definir métodos privados
2. Encapsular os atributos da classe
3. Fornecer uma descrição da classe
4. Exportar a classe para outros arquivos
5. Inicializar os atributos da classe.

**Questão 6 (1,0) - Qual é a finalidade do uso de getters em uma classe TypeScript?**

1. Para criar instâncias de subclasses.
2. Para acessar os atributos privados da classe.
3. Para modificar os valores dos atributos.
4. Para exportar a classe para outros arquivos.
5. Para definir métodos estáticos na classe.

**Questão 7 (1,0) - O que o encapsulamento visa alcançar em programação orientada a objetos?**

1. Facilitar a exportação de classes.
2. Expor todos os membros de uma classe.
3. Permitir que qualquer classe acesse os atributos.
4. Tornar os métodos públicos inacessíveis.
5. Limitar o acesso aos membros de uma classe.

**Questão 8 (1,0) - Qual é uma das principais vantagens da herança em programação orientada a objetos?**

1. Maior complexidade no código.
2. Restrição no acesso aos membros da classe pai.
3. Menor reutilização de código.
4. Facilidade na manutenção do código.
5. Limitação na criação de subclasses.

**Questão 9 (1,0) - O que caracteriza o relacionamento de composição entre duas classes em TypeScript?**

1. Uma classe é associada a outra classe através de uma relação permanente.
2. Uma classe utiliza serviços temporários de outra classe.
3. Uma classe é formada a partir da implementação de uma interface.
4. Uma classe é composta por várias instâncias de outra classe.
5. Uma classe herda os atributos e métodos de outra classe.

**Questão 10 (1,0) - Qual é uma das principais características do relacionamento de agregação em programação orientada a objetos?**

1. Uma classe é associada a outra classe através de uma relação permanente
2. Uma classe é composta por várias instâncias de outra classe.
3. Uma classe é formada a partir da implementação de uma interface.
4. Uma classe utiliza serviços temporários de outra classe.
5. Uma classe herda os atributos e métodos de outra classe.