

ATIVIDADE 3 - ESOFT - PROGRAMAÇÃO ORIENTADA A OBJETOS - 54_2024

Período: 21/10/2024 08:00 a 24/11/2024 23:59 (Horário de Brasília)

Status: ABERTO

Nota máxima: 0,50

Gabarito: Gabarito será liberado no dia 25/11/2024 00:00 (Horário de Brasília)

Nota obtida:

1ª QUESTÃO

Um programa escrito em Java começa sempre a ser executado a partir do método main. O método main possui uma assinatura padrão que utilizamos. Com base nesta informação, assinale a alternativa que contém a assinatura padrão correta.

ALTERNATIVAS

public void main static(args :String)
 public static void main(String args)
 private static main void(String args)
 static public main void(String args)
 static public void main(String args)
 args)

2ª QUESTÃO

Em Java é possível definir um elemento de uma classe (atributo, método ou variável de classe) ou até a mesma classe, com a característica de que não podem ser alterados, herdados ou sofrer polimorfismo em tempo de execução. Assinale a alternativa com o modificador responsável por este comportamento:

ALTERNATIVAS

- Final.
- Static.
- Void.
- Abstract.
- Protected.

3ª QUESTÃO

Alguns modificadores são responsáveis por proibir certos recursos do Java, tais como o "final". De acordo com o funcionamento dos modificadores "static", "final" e "abstract", analise as afirmações abaixo:

- I. Static pode ser atribuído a classes, atributos e métodos.
- II. Métodos assinados com static não necessitam de instância da classe onde estão para serem invocados.
- III. Final pode ser atribuído a classes, atributos e métodos.
- IV. Classes abstratas não permitem a criação de objetos.

É correto o que se afirma em:

ALTERNATIVAS

- I, apenas.
- I e II, apenas.
- III e IV, apenas.
- II, III e IV, apenas.
- I, II, III e IV.

4ª QUESTÃO

Uma aplicação java deve ter pelo menos uma classe que contenha o método main(), o qual é responsável por iniciar toda a execução do programa. Todo método deve possuir uma assinatura que compreende um conjunto de informações que definem o método.

Sobre a assinatura do método main(), analise as informações abaixo:

- I. Todo método main() deve ser público (public).
- II. Todo método main() deve ser estático (static).
- III. Todo método main() deve receber um array de objetos String como parâmetro.
- IV. Todo método main() recebe a palavra reservada void, pois retorna valores desse tipo ao final de sua execução.

É correto o que afirma em:

ALTERNATIVAS

- I, apenas.
- I e III, apenas.
- I, II e III, apenas.
- II, III e IV, apenas.
- I, II, III e IV.

5ª QUESTÃO

A declaração de variáveis no Java precisa ser feita com muita atenção. Por ser uma linguagem de programação orientada a objetos, precisamos nos atentar ao escopo das variáveis que desejamos declarar, pois isso faz toda a diferença em sua utilização. Pautando-se nessas informações assinale a alternativa correta:

```
package unicesumar.ead.programacao;

public class Pessoa {

    private String nome;
    private int idade;

    private String falar(){

        String frase = "Olá";
        return frase;
    }

    public void andar(){
        System.out.println(nome + frase);
    }
}
```

ALTERNATIVAS

- A variável "idade" foi declarada e referenciada.
- A variável "frase" não pode ser declarada no método "falar", pois ele é privado.
- A variável "nome" é uma variável de classe e não pode ser referenciada dentro dos métodos.
- A variável "frase", não pode ser utilizada no método "andar", pois ela foi criada apenas no método "falar".
- O sinal de "+" não pode ser utilizado para somar duas Strings no Java, como está sendo feito dentro do método "andar".

6ª QUESTÃO

Os modificadores de acesso em Java, vão restringir o acesso ao membro da classe ou à própria classe. Dada uma classe chamada "Animal", onde exista um atributo do tipo double chamado "tamanho" definido com o modificador de acesso "protected" e ausente de métodos assessores, assinale a alternativa que expressa corretamente a forma de acesso a esse atributo por outras classes.

ALTERNATIVAS

- O atributo não será acessível.
- O atributo será acessível somente dentro da própria classe.
- O atributo será acessível por qualquer classe do projeto mesmo que não esteja no mesmo pacote que a classe Animal.
- O atributo somente será acessível por classes que estejam no mesmo pacote que a classe Animal ou por classes que tenham herdado as características dessa classe.
- O atributo somente será acessível por classes que estejam no mesmo pacote que a classe Animal e não será acessível às classes que tenham herdado as características dessa.

7ª QUESTÃO

A Herança em Orientação a Objetos é um dos recursos mais utilizados pelos programadores, pois propõe inúmeras vantagens tais como reaproveitamento de código, menos linhas de código escritas, maior estruturação do projeto, melhora na extensão do código quando se necessita criar novas funcionalidades, facilita a correção de erros etc. Toda linguagem Orientada a Objetos possui esse recurso, sendo que em algumas delas é possível fazer Herança Múltipla.

Sobre Herança Múltipla em JAVA, analise as alternativas abaixo e assinale a correta.

ALTERNATIVAS

- O JAVA não possibilita os recursos de Herança Múltipla.
- Em JAVA esse recurso é melhor empregado pelos modificadores de acesso.
- A Herança Múltipla permite que uma classe herde métodos e atributos de mais de uma classe.
- O JAVA possibilita os recursos de Herança Múltipla por meio da utilização de classes abstratas.
- O interpretador JAVA se encarrega de realizar a Herança Múltipla sem o programador se preocupar com a implementação da mesma.

8ª QUESTÃO

Quando uma classe pai define um método, ele pode ser sobreescrito, na maioria dos casos, em suas classes filhas, realizando, assim, o conceito de polimorfismo. Nesse sentido analise o trecho de código abaixo, assinale a alternativa correta.

```
public final void calcular() {  
    //Código de claculo...  
}  
Classe Pai  
  
public final void calcular() {  
    //Código sobreescrito da classe pai...  
}  
Classe Filha
```

ALTERNATIVAS

- O Polimorfismo será executado normalmente.
- A diretiva "final" significa que o método "calcular" é o último método da classe.
- Não será possível executar a sobrecarga do método "calcular", pois, para isso, ele deveria ser "protected" e não "public".
- Será possível a aplicação do "Poliformismo", pois o método "calcular" está assinado corretamente nas classes pai e filha.
- Não será possível a aplicação do "Polimorfismo", pois o método "calcular", na classe pai, está assinado com a palavra "final".

9ª QUESTÃO

Um dos princípios da Programação Orientada a Objetos é a possibilidade de se criar novas classes a partir de outras já criadas. Como estudamos, uma subclasse herda as características da superclasse.

Assinale a alternativa que contém a palavra que usamos na linha de declaração de uma classe para defini-la como subclasse.

ALTERNATIVAS

- Subclass.
 - Inherits.
 - Abstract.
 - Extends.
 - Static.
-

10ª QUESTÃO

Já vimos que toda linguagem de programação orientada a objetos, trabalha com o conceito de classes, objetos, abstrações e muitos outros. Esses conceitos são a base para um bom desenvolvimento em linguagens como Java.

Sobre os conceitos de orientação a objetos, analise as afirmações abaixo:

- I. Classes, são considerados moldes para criação de objetos. Naselas definimos o que um objeto tem e faz.
- II. Polimorfismo compreende a técnica de não expor detalhes internos para o usuário, tornando as partes do sistema mais independentes possível.
- III. Herança compreende a possibilidade de uma classe herdar as características e métodos de outra classe.

É correto o que se afirma em:

ALTERNATIVAS

- I, apenas.
 - I e II, apenas.
 - I e III, apenas.
 - II e III, apenas.
 - I, II e III.
-