1a GUIA DE POO

(Roberto Tecla)

Nombre: Bocanegra Heziquio Yestlanezi Grupo:

Fecha: 6 de mayo de 2021

1.-Diga a que concepto corresponde la definición siguiente: denota las **características esenciales** **que distinguen a un objeto de otra clase de objetos** y provee así una frontera conceptual definida, relativa a la perspectiva del observador.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a) Identidad de un objeto | b) Encapsulamiento | c) Operación | d) Abstracción |  |

2.- Diga a que concepto corresponde la definición siguiente: **engloba todas las** (usualmente estáticas) **propiedades del objeto** mas los valores actuales (usualmente dinámicos) de cada una de sus propiedades.

|  |  |  |
| --- | --- | --- |
| A) Estado de un objeto | B) Comportamiento de un objeto |  |
| C) Identidad de un objeto | D) Interfaz de un objeto |  |

3.-Diga a que concepto corresponde la definición siguiente: Es como **actúa** y reacciona **un objeto en términos** de **cambio de estado**.

|  |  |  |
| --- | --- | --- |
| A) Encapsulamiento | B) Comportamiento de un objeto |  |
| C) Identidad de un objeto | D) Interfaz de un objeto |  |

4.-Diga a que concepto corresponde la definición siguiente: es la **propiedad** de un objeto que lo **distingue de todos los otros objetos**

|  |  |  |
| --- | --- | --- |
| A) Estado de un objeto | B) Comportamiento de un objeto |  |
| C) Identidad de un objeto | D) Interfaz de un objeto |  |

5.- Es una **colección de objetos que comparten estructura y comportamiento**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a) herencia | b) clase | c) paquete | d) ) interfaz |  |

6.- Diga a que concepto corresponde la definición siguiente: es una **función** o transformación q**ue se puede aplicar o que puede ser aplicada** por los objetos de una clase.

|  |  |  |
| --- | --- | --- |
| A) Identidad de un objeto | B) Encapsulamiento |  |
| C) Operación | C) Abstracción |  |

7.- Diga a que concepto corresponde la definición siguiente: Es una **relación entre clases** en la que **una clase comparte la estructura y/o el comportamiento definido en una o más clases**.

|  |  |  |
| --- | --- | --- |
| A) Herencia | B) Interfaz |  |
| C) Una clase abstracta | C) Encapsulamiento |  |

8.- Es la utilización de **interfaces idénticas** con **diferentes implementaciones**.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A) metodo | B) polimorfismo | C) mensaje | D) operacion | E) Abstraccion |  |

9.-De acuerdo con su **ámbito** que **variables** se pueden usar en mas de un método de la misma clase

|  |  |  |  |
| --- | --- | --- | --- |
| a) Las de instancia | b) Las locales | c) Ninguna de las anteriores |  |

10.-De acuerdo con su **ámbito** que **variables** se declaran dentro de un método

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a) Las de instancia | b) Las de clase | c) Las locales | d) Ninguna de las anteriores |  |

11.-Cual de las siguientes no es una **palabra reservada** en Java?

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| a) byte | c) catch | c) real | d) implements | e) static |  |

12.-Los tipos básicos en Java son:

|  |
| --- |
| A) String, float, double, integer, short, long, char |
| B) integer, byte, String, float, short, long |
| C) int, byte, double, float, short, long, char, boolean |
| D) int, short, long, char, double, float |

13.-Cual de las siguientes afirmaciones no es cierta para un constructor ?

|  |  |  |
| --- | --- | --- |
| A) Su tipo de retorno es void | B) Esta sobrecargado |  |
| C) Sirve para inicializar los objetos de la clase | D) Se llama como la clase |  |

14.-En Java\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**solo contiene constantes y métodos abstractos**

|  |  |  |
| --- | --- | --- |
| A) Una clase abstracta | B) Una abstracción |  |
| C) Una interfaz | D) Una operación |  |

15.-En Java cual de las siguientes definiciones del metodo suma de la clase Complejo es la correcta.

Nota: los … representan codigo no mostrado.

|  |  |  |
| --- | --- | --- |
| void suma(){...} | Complejo suma(Complejo c) {...} |  |
| Complejo suma() {...} | Complejo suma(Complejo c1, Complejo c2){...} |  |

Diga si las afirmaciones siguientes son **verdaderas(**V) **o falsas** (F):

|  |  |
| --- | --- |
| 1.-En POO los datos deben ir junto al codigo que los manipula | (V) |
| 2-En POO los datos no deben ir junto al codigo que los manipula | (F) |
| 3.-La representacion de un TAD solo debe ser manipulada por las operaciones definidas para ese TAD | (V) |
| 4.-El **encapsulamiento** sirve para ocultar la **complejidad** | (V) |
| 5.-El **encapsulamiento** sirve para proteger los **datos** | (V) |
| 6.- Si un programa en Java tiene un método **main** podemos asegurar que es una **aplicación** | (F) |
| 7.-Si un programa en Java tiene un método **main** podemos asegurar que es un **applet** | (F) |
| 8.-Si un programa en Java tiene como superclase a la clase **Applet** podemos asegurar que es un **applet** | (V) |
| 9.-El **estado** es lo que permite distinguir a un objeto de otro | (V) |
| 10.-La **herencia** en Java es múltiple | (F) |
| 11.-La **herencia** en Java es simple | (V) |
| 12.-Es la  **herencia** la que permite usar métodos y atributos definidos en una superclase | (V) |
| 13.-La **herencia** sirve para reutilizar codigo | (V) |
| 14.-La **herencia** nosirve para reutilizar codigo | (F) |
| 15.-La **superclase** generaliza a las **subclases** | (V) |
| 16.-La **superclase** no generaliza a las **subclases** | (F) |
| 17.-Las **subclases** no son casos particulares de la **superclase** | (F) |
| 18.-Las **subclases** son casos particulares de la **superclase** | (F) |
| 19.-Si usamos **herencia** estamos haciendo uso del concepto de **abstraccion** | (V) |
| 20.-La POO surge para manejar el cambio y la complejidad | ( F) |
| 21.-La POO surge para manejar el cambio y la abstraccion | (F) |
| 22.-El **constructor** de los objetos de una clase es un metodo | (V) |
| 23.-El **constructor** de los objetos de una clase no es un metodo | (F) |
| 24.-El nombre del **constructor** de los objetos de una clase no tiene que ser el mismo que el nombre de dicha clase | (F) |
| 25.-El **constructor** sirve para inicializar los objetos de una clase | (V) |
| 26.-El **constructor** no sirve para inicializar los objetos de una clase | (F) |
| 27.-El **constructor** de los objetos de una clase tiene como tipo de retorno **void** | (V) |
| 28.-La palabra reservada **this** se usa para **resolver conflictos de nombres** | (V) |
| 29.-Si 2 o mas métodos tienen el mismo nombre pero distinta implementación entonces el método esta **sobrecargado** | (V) |
| 30.-Si hay mas de un constructor en una clase entonces el **constructor** esta **sobrecargado** | (F) |
| 31.-Una **clase abstracta** no puede tener instancias directas | (V) |
| 32.-Una **clase abstracta**, en Java, no puede tener variables de instancia | (F) |
| 33.-En Java si una clase tiene un **método abstracto** entonces es una clase abstracta | (V) |
| 34.-Un método **abstracto** no tiene implementación | (V) |
| 35.-Un método **abstracto** no se define es decir solo se declara | (F) |
| 36.-Una **interfaz**, en Java, solo contiene **constantes y métodos abstractos** | (F) |
| 37.-Todo lo que esta dentro de una **interfaz** es **publico** | (V) |
| 38.-Todo lo que esta dentro de una **interfaz** es **privado** | (F) |
| 39.-Una **interfaz** puede contener métodos abstractos y variables de instancia | (V) |
| 40.-Una clase no puede **implementar** más de una **interfaz** | (V) |
| 41.-La clase que **implementa** una interfaz no tiene que implementar todos los métodos contenidos en dicha **interfaz** | (F) |
| 42.-Una **variable de clase** es compartida por todos los objetos de la clase | (V) |
| 43.-En Java se pueden **concatenar cadenas** usando el operador + | (V) |
| 44.-Se usa nombreObjeto.metodo(listaParametrosReales) para llamar un **método de instancia** | (V) |
| 45.-Se usa nombreClase.metodo(listaParametrosReales) para llamar un **método de instancia** | (V) |