Top of Form

|  |  |
| --- | --- |
| **1.** | overloaded methods should have same name. |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |
| **2.** | overloaded methods should have different name. |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| **3.** | overloaded methods should have same name and difference in the number of arguments or difference in any one argument data type. | |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | | |
|  | | | |
| **4.** | overloaded methods should have same return type. |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | | |
|  | | | |

|  |  |
| --- | --- |
| **5.** | overloaded methods should have different return type. |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |
| **6.** | overloaded methods can have different return type. |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |

|  |  |
| --- | --- |
| **7.** | overloaded methods can have same return type. |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |
| **8.** | overloaded methods can be both static and not staic also. |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| **9.** | overloaded methods can be only static | |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | | |
|  | | | |
| **10.** | | overloaded methods can be only non static |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | | |
|  | | | |

|  |  |
| --- | --- |
| **11.** | overloaded methods can be only abstract |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |
| **12.** | defined methd cant be overloaded with an abstract one |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |

|  |  |
| --- | --- |
| **13.** | all overloaded methods should have same access specifier |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |
| **14.** | all overloaded methods can have same access specifier |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |

|  |  |
| --- | --- |
| **15.** | all overloaded methods can have different access specifier |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |
| **16.** | inherited method can overload with an incorporated method in the subclass |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |

|  |  |
| --- | --- |
| **17.** | super class method can overload with a subclass method in the subclass |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |
| **18.** | abstract method of super class can overload in the subclass |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |

|  |  |
| --- | --- |
| **19.** | while overriding a method, signature should be same |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |
| **20.** | while overriding a method, signature should differ |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |

|  |  |
| --- | --- |
| **21.** | while overriding a method, return type should differ |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |
| **22.** | while overriding a method, return type shold be same except in the derived datatype. |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |

|  |  |
| --- | --- |
| **23.** | what could be the return type while overring super class method in the sub class, if super class method return type is void. |
| |  | | --- | | A.  void only |  |  | | --- | | B.  any return type | | | |
|  | | |
| **24.** | what could be the return type while overring super class method in the sub class, if super class method return type is int. |
| |  | | --- | | A.  byte or int |  |  | | --- | | B.  int only | | | |
|  | | |

|  |  |
| --- | --- |
| **25.** | what could be the return type while overring super class method in the sub class, if super class method return type is A type. Assume B is a subclass to A. |
| |  | | --- | | A.  A |  |  | | --- | | B.  B | | | |
| **,** | | |
| **26.** | what could be the return type while overring super class method in the sub class, if super class method return type is A type. Assume A is a subclass to B. |
| |  | | --- | | A.  A |  |  | | --- | | B.  B | | | |
|  | | |

|  |  |
| --- | --- |
| **27.** | what could be the the access specifier while overring super class method in the sub class, if super class method access specifier is protected. |
| |  | | --- | | A.  private |  |  | | --- | | B.  protected |  |  | | --- | | C.  public | | | |
| **,** | | |
| **28.** | what could be the the access specifier while overring super class method in the sub class, if super class method access specifier is package scope. |
| |  | | --- | | A.  private |  |  | | --- | | B.  protected |  |  | | --- | | C.  public | | | |
| **,** | | |

|  |  |
| --- | --- |
| **29.** | is it possible to override static method of super class with non static in the sub class. |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |
| **30.** | which key word is used to call super class method while its got overrided in the subclass |
| |  | | --- | | A.  super |  |  | | --- | | B.  extends | | | |
|  | | |

|  |  |
| --- | --- |
| **31.** | method overriding is required to achieve polimorphism. |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |
| **32.** | constructor overriding is required to achieve polimorphism. |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |

|  |  |
| --- | --- |
| **33.** | upcasting is required to achieve polimorphism. |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |
| **34.** | downcasting is required to achieve polimorphism. |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |

|  |  |
| --- | --- |
| **35.** | which two are required to achieve polimorphism. |
| |  | | --- | | A.  down casting |  |  | | --- | | B.  up casting |  |  | | --- | | C.  constructor overriding |  |  | | --- | | D.  method overriding | | | |
| **,D** | | |
| **36.** | static methods cant be overrided |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |

|  |  |
| --- | --- |
| **37.** | static methods can be overrided |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |
| **38.** | static methods are involving in the polymorphism |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |

|  |  |
| --- | --- |
| **39.** | subclass doesn’t required to load while accessing super class static member with a subclass name. |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |

Bottom of Form