Top of Form

|  |  |
| --- | --- |
| **1.** | final key word is used to the abstract methods |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |
| **2.** | final key word is used to the private methods |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |

|  |  |
| --- | --- |
| **3.** | final key word is used to the static methods |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
|  | | |
| **4.** | static methods can be an abstract |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
| **Correct Answer: `** | | |

|  |  |
| --- | --- |
| **5.** | absstract class can be final |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
| **Correct Answer: `** | | |
| **6.** | constructor can be final |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | |
| **Correct Answer: `** | | |

|  |  |
| --- | --- |
| **7.** | package src;  class A  {  public static void main(String[] args)  {  final int i = 10;  int j = 20;  System.out.println(i);  System.out.println(j);  i = 10;  j = 20;  System.out.println(i);  System.out.println(j);  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  10  20  10  20 |  |  | | --- | | C.  10 20 10 20 |  |  | | --- | | D.  Runtime Error | | | |
|  | | |
| **8.** | package src;  class B  {  public static void main(String[] args)  {  final int i;  i = 10;  int j = 20;  System.out.println(i);  System.out.println(j);  i = 10;  j = 20;  System.out.println(i);  System.out.println(j);  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  10  20  10  20 |  |  | | --- | | C.  10 20 10 20 |  |  | | --- | | D.  Runtime Error | | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| **9.** | package src;  class C  {  public static void main(String[] args)  {  C c1 = new C();  c1 = new C();  final C c2 = new C();  c2 = new C();  System.out.println("done");  }  }  Will it compiles successfully are not? | |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | | |
|  | | | |
| **10.** | | package src;  class D  {  int i;  public static void main(String[] args)  {  D d1 = new D();  d1.i = 10;  final D d2 = new D();  d2.i = 10;  d1.i = 10;  d2.i = 10;  System.out.println("Hello World!");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  RunTime Error |  |  | | --- | | C.  Hello World! |  |  | | --- | | D.  None | | | | |
|  | | | |

|  |  |
| --- | --- |
| **11.** | package src;  class E  {  final int i = 10;  public static void main(String[] args)  {  E e2 = new E();  e2 = new E();  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compilation Error |  |  | | --- | | C.  Runtime Error | | | |
|  | | |
| **12.** | package src;  class F  {  public static void main(String[] args)  {  final int[] x = new int[3];  x[2] = 20;  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compilation Error |  |  | | --- | | C.  Runtime Error | | | |
|  | | |

|  |  |
| --- | --- |
| **13.** | package src;  class G  {  public static void main(String[] args)  {  final int[] x = new int[3];  x = new int[3];  System.out.println("done");  }  }  Will it compiles fine or not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |
| **14.** | package src;  class H  {  public static void main(final String[] args)  {  args[0] = "abc";  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compilation Error |  |  | | --- | | C.  ArrayIndexOutOfBoundsException |  |  | | --- | | D.  None | | | |
|  | | |

|  |  |
| --- | --- |
| **15.** | package src;  class I  {  public static void main(final String[] args)  {  args = null;  System.out.println("done");  }  }  Will it compiles fine are not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |
| **16.** | package src;  class J  {  static void test(int x, final int y)  {  x = 10;  y = 10;  System.out.println("---------");  x = 10;  y = 10;  System.out.println("---------");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  ---------  --------- |  |  | | --- | | C.  Runtime Error |  |  | | --- | | D.  None | | | |
|  | | |

|  |  |
| --- | --- |
| **17.** | package src;  class K  {  final int i;  }  Will it compiles fine are not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |
| **18.** | package src;  class L  {  final int i = 0;  }  Will it compiles fine are not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |

|  |  |
| --- | --- |
| **19.** | package src;  class M  {  final int i;  M()  {  i = 20;  }  }  Will it compiles fine are not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |
| **20.** | package src;  class N  {  final int i = 20;  N()  {  i = 20;  }  }  Will it compiles fine are not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |

|  |  |
| --- | --- |
| **21.** | package src;  class O  {  final int i;  {  i = 20;  }  }  Will it compiles fine are not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |
| **22.** | package src;  class P  {  final int i = 20;  {  i = 20;  }  }  Will it compiles fine are Not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |

|  |  |
| --- | --- |
| **23.** | package src;  class Q  {  final int i;  Q()  {  i = 20;  }  {  i = 20;  }  } |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |
| **24.** | package src;  class R  {  final int i = 10;  public static void main(String[] args)  {  R r1 = new R();  r1.i = 20;  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compilation Error |  |  | | --- | | C.  Runtime Error | | | |
|  | | |

|  |  |
| --- | --- |
| **25.** | package src;  class S  {  final int i;  S()  {  i = 10;  }  S(int x)  {  i = 20;  }  }  Will it compiles fine are not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |
| **26.** | package src;  class T  {  final int i;  T()  {  i = 10;  }  T(int x)  {  this();  i = 20;  }  }  Will it compiles Fine are not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |

|  |  |
| --- | --- |
| **27.** | package src;  class U  {  final int i;  U()  {  i = 10;  }  U(int x)  {  }  }  Will it compiles fine are not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |
| **28.** | package src;  class V  {  final int i;  V()  {  i = 10;  }  V(int x)  {  this();  }  }  Will it compiles fine are not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |

|  |  |
| --- | --- |
| **29.** | package src;  class W  {  static final int x;  public static void main(String[] args)  {  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compilation Error |  |  | | --- | | C.  Runtime Error | | | |
|  | | |
| **30.** | package src;  class X  {  static final int x = 0;  public static void main(String[] args)  {  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compilation Error |  |  | | --- | | C.  Runtime Error |  |  | | --- | | D.  0 | | | |
|  | | |

|  |  |
| --- | --- |
| **31.** | package src;  class Y  {  static final int x = 0;  public static void main(String[] args)  {  x = 0;  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compilation Error |  |  | | --- | | C.  Runtime Error |  |  | | --- | | D.  0 | | | |
|  | | |
| **32.** | package src;  class Z  {  final static int i;    static  {  i = 10;  }  }  Will it compiles fine are not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |

|  |  |
| --- | --- |
| **33.** | package src;  class Z1  {  final static int i = 10;    static  {  i = 10;  }  }  Will it compiles fine are not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |
| **34.** | package src;  class Z2  {  static final int SOME\_VALUE = 100;  public static void main(String[] args)  {  System.out.println(SOME\_VALUE);  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  RunTime Error |  |  | | --- | | C.  100 |  |  | | --- | | D.  0 | | | |
|  | | |

|  |  |
| --- | --- |
| **35.** | package src;  class Z3  {  public static void main(String[] args)  {  System.out.println(Byte.MIN\_VALUE);  System.out.println(Byte.MAX\_VALUE);  System.out.println(Thread.MIN\_PRIORITY);  }  } |
| |  | | --- | | A.  -128  127  1 |  |  | | --- | | B.  128  -127  1 |  |  | | --- | | C.  Compilation Error | | | |
|  | | |
| **36.** | package src;  interface A  {  int i = 10;  }  class Z4  {  public static void main(String[] args)  {  System.out.println(A.i);  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  10 |  |  | | --- | | C.  0 |  |  | | --- | | D.  None | | | |
|  | | |

|  |  |
| --- | --- |
| **37.** | package src;  interface A  {  int i;  }  class Z5  {  public static void main(String[] args)  {  System.out.println(A.i);  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  10 |  |  | | --- | | C.  0 |  |  | | --- | | D.  None | | | |
|  | | |
| **38.** | package src;  interface A3  {  String COUNTRY\_NAME = "Great India";  }  class Z6  {  public static void main(String[] args)  {  System.out.println(A3.COUNTRY\_NAME);  }  } |
| |  | | --- | | A.  Great India |  |  | | --- | | B.  Compilation Error |  |  | | --- | | C.  Running successfully with no output | | | |
|  | | |

|  |  |
| --- | --- |
| **39.** | class A  {  void test1()  {  }  final void test2()  {  }  }  class B extends A  {  void test1()  {  }  void test2()  {  }  }  class M1  {  public static void main(String[] args)  {  System.out.println("Hello World!");  }  } |
| |  | | --- | | A.  Hello World! |  |  | | --- | | B.  Compilation Error |  |  | | --- | | C.  None | | | |
|  | | |
| **40.** | package src;  abstract class A  {  abstract final void test1();  }  class M2  {  public static void main(String[] args)  {  System.out.println("Hello World!");  }  }  Will it compiles fine are not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |

|  |  |
| --- | --- |
| **41.** | package src;  abstract class A  {  static abstract void test1();  }  class M3  {  public static void main(String[] args)  {  System.out.println("Hello World!");  }  } |
| |  | | --- | | A.  Hello World! |  |  | | --- | | B.  RunTime Error |  |  | | --- | | C.  Compilation Error | | | |
|  | | |
| **42.** | package src;  class A  {  static final void test1()  {  }  static void test2()  {  }  }  class B extends A  {  static void test1()  {  }  static void test2()  {  }  }  class M4  {  public static void main(String[] args)  {  System.out.println("Hello World!");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  RunTime Error |  |  | | --- | | C.  Hello World! | | | |
|  | | |

|  |  |
| --- | --- |
| **43.** | class A  {  private final void test1()  {  }  }  class B extends A  {  void test1()  {  }  }  class M5  {  public static void main(String[] args)  {  System.out.println("Hello World!");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  RunTime Error |  |  | | --- | | C.  Hello World! | | | |
|  | | |
| **44.** | package src;  class A  {  final A()  {  }  }  class M6  {  public static void main(String[] args)  {  System.out.println("Hello World!");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  RunTime Error |  |  | | --- | | C.  Hello World! | | | |
|  | | |

|  |  |
| --- | --- |
| **45.** | package src;  final class A  {  }  class B extends A  {  }  class M7  {  public static void main(String[] args)  {  System.out.println("Hello World!");  }  } |
| |  | | --- | | A.  Hello World! |  |  | | --- | | B.  RunTime Error |  |  | | --- | | C.  Compilation Error | | | |
|  | | |
| **46.** | package src;  abstract final class A  {  }  class M8  {  public static void main(String[] args)  {  System.out.println("Hello World!");  }  } |
| |  | | --- | | A.  Hello World! |  |  | | --- | | B.  RunTime Error |  |  | | --- | | C.  Compilation Error | | | |
|  | | |

|  |  |
| --- | --- |
| **47.** | final variab;les can be modified further |
| |  | | --- | | A.  True |  |  | | --- | | B.  False | | | |
|  | | |
| **48.** | Is it possible to declare main mathod arguments as final |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |

|  |  |
| --- | --- |
| **49.** | Final global variable can survive with a default value |
| |  | | --- | | A.  True |  |  | | --- | | B.  False | | | |
|  | | |
| **50.** | Is it possible to intilize final global variable with intilizers |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |

|  |  |
| --- | --- |
| **51.** | static is loading into the memory while class is loading |
| |  | | --- | | A.  True |  |  | | --- | | B.  False | | | |
|  | | |
| **52.** | In java static and final is perfect constant |
| |  | | --- | | A.  True |  |  | | --- | | B.  False | | | |
|  | | |

|  |  |
| --- | --- |
| **53.** | Is it possible to keep method as final |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |
| **54.** | Is it possible to override a final method |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |

|  |  |
| --- | --- |
| **55.** | Is it possible to use both abstract and final together |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |
| **56.** | static method can be qulaified as abstract? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
|  | | |

|  |  |
| --- | --- |
| **57.** | Class can not be final |
| |  | | --- | | A.  True |  |  | | --- | | B.  False | | | |
|  | | |

Bottom of Form