|  |  |  |
| --- | --- | --- |
| **1.** | Generics are used for changing data type of any members from one object to another object. | |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | | |
| **Correct Answer: A** | | | |
| **2.** | We can use generics to change field data type from one object to another object. |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | | | |
| **Correct Answer: A** | | | |

|  |  |
| --- | --- |
| **3.** | In which version of JDK, generics are introduced? |
| |  | | --- | | A.  JDK1.8 |  |  | | --- | | B.  JDK1.5 |  |  | | --- | | C.  JDK1.6 | | | |
| **Correct Answer: B** | | |
| **4.** | class A  {  int i;  }  class M1  {  public static void main(String[] args)  {  A a1 = new A();  a1.i = 10;  A a2 = new A();  a2.i = 3;  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: A** | | |

|  |  |
| --- | --- |
| **5.** | class A  {  String obj;  }  class M3  {  public static void main(String[] args)  {  A a1 = new A();  a1.obj = "hello";  A a2 = new A();  a2.obj = "xyz";  A a3 = new A();  a3.obj = "test";  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: A** | | |
| **6.** | class A  {  String obj1;  Integer obj2;  Double obj3;  }  class M4  {  public static void main(String[] args)  {  A a1 = new A();  a1.obj1 = "hello";  a1.obj2 = 20;  a1.obj3 = 4.5;  A a2 = new A();  a2.obj1 = "test";  a2.obj2 = 40;  a2.obj3 = 5.4;  System.out.println("done");  }  }  // will it compiles successfull or not |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
| **Correct Answer: A** | | |

|  |  |
| --- | --- |
| **7.** | class A  {  void test(String arg)  {  }  }  class M5  {  public static void main(String[] args)  {  A a1 = new A();  A a2 = new A();  A a3 = new A();  a1.test("abc");  a2.test("hello");  a3.test("xyz");  System.out.println("Hello World!");  }  }  // Will it compiles successfull are not |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
| **Correct Answer: A** | | |
| **8.** | class A  {  void test(Integer arg)  {  }  }  class M6  {  public static void main(String[] args)  {  A a1 = new A();  A a2 = new A();  A a3 = new A();  a1.test(20);  a2.test(59);  a3.test(1000);  System.out.println("Hello World!");  }  }  // will it compiles successfully or not |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
| **Correct Answer: A** | | |

|  |  |  |
| --- | --- | --- |
| **9.** | class A  {  void test(Integer arg1, String arg2)  {  }  }  class M7  {  public static void main(String[] args)  {  A a1 = new A();  A a2 = new A();  A a3 = new A();  a1.test(20, "abc");  a2.test(59, "hello");  a3.test(1000, "hi");  System.out.println("Hello World!");  }  }  // Will it compiles successful or not? | |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | | |
| **Correct Answer: A** | | | |
| **10.** | | class A  {  String test()  {  return null;  }  }  class M8  {  public static void main(String[] args)  {  A a1 = new A();  A a2 = new A();  String s1 = a1.test();  String s2 = a2.test();  System.out.println("Hello World!");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  Hello World | | | | |
| **Correct Answer: C** | | | |

|  |  |
| --- | --- |
| **11.** | class A  {  Double test()  {  return null;  }  }  class M9  {  public static void main(String[] args)  {  A a1 = new A();  A a2 = new A();  Double s1 = a1.test();  Double s2 = a2.test();  System.out.println("Hello World!");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  Hello World | | | |
| **Correct Answer: C** | | |
| **12.** | class A  {  A(String s1)  {  }  }  class M10  {  public static void main(String[] args)  {  A a1 = new A("abc");  A a2 = new A("xyz");  A a3 = new A("hello");  System.out.println("done");  }  }  // Will it compiles successfull or not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
| **Correct Answer: A** | | |

|  |  |
| --- | --- |
| **13.** | class A < X >  {  X obj;  }  class M11  {  public static void main(String[] args)  {  A<Integer> a1 = new A<Integer>();  A<String> a2 = new A<String>();  A<Double> a3 = new A<Double>();  A a4 = new A();  a1.obj = 40;  a2.obj = "abc";  a3.obj = 4.5;  a4.obj = new Object();  System.out.println("done");  }  }  //Will it compiles successfull or Not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
| **Correct Answer: A** | | |
| **14.** | class A < X >  {  void test(X obj)  {  }  }  class M12  {  public static void main(String[] args)  {  A<Integer> a1 = new A<Integer>();  A<String> a2 = new A<String>();  A<Double> a3 = new A<Double>();  a1.test(10);  a2.test("abc");  a3.test(3.5);  System.out.println("done");  }  }  //Will it compiles successfull or Not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
| **Correct Answer: A** | | |

|  |  |
| --- | --- |
| **15.** | class A < X >  {  X obj;  X test()  {  return obj;  }  }  class M13  {  public static void main(String[] args)  {  A<Integer> a1 = new A<Integer>();  A<String> a2 = new A<String>();  A<Double> a3 = new A<Double>();  Integer obj1 = a1.test();  String obj2 = a2.test();  Double obj3 = a3.test();  System.out.println("done");  }  }  // Will it compiles successfull or Not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
| **Correct Answer: A** | | |
| **16.** | class A < T, U >  {  T obj1;  U obj2;  void test1(T t1)  {  }  U test2()  {  return null;  }  }  class M14  {  public static void main(String[] args)  {  A<Integer, String> a1 = new A<Integer, String>();  A<String, String> a2 = new A<String, String>();  A<Integer, Integer> a3 = new A<Integer, Integer>();  A<String, Integer> a4 = new A<String, Integer>();  a1.obj1 = 10;  a1.obj2 = "abc";  a2.obj1 = "hello";  a2.obj2 = "test";  a3.obj1 = 44;  a3.obj2 = 55;  a4.obj1 = "xyz";  a4.obj2 = 59;  a1.test1(30);  String s1 = a1.test2();  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  done |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |

|  |  |
| --- | --- |
| **17.** | class A < Test >  {  A(Test obj)  {  }  }  class M15  {  public static void main(String[] args)  {  A<String> a1 = new A<String>("abc");  A<Integer> a2 = new A<Integer>(90);  A<Double> a3 = new A<Double>(4.5);  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: A** | | |
| **18.** | interface A < X >  {  X test1();  void test2(X x1);  void test3(X x1, X x2);  }  class B implements A < String >  {  public String test1()  {  return null;  }  public void test2(String s1)  {  }  public void test3(String s1, String s2)  {  }  }  class C implements A < Integer >  {  public Integer test1()  {  return null;  }  public void test2(Integer s1)  {    }  public void test3(Integer s1, Integer s2)  {    }  }  class M16  {  public static void main(String[] args)  {  B b1 = new B();  String s1 = b1.test1();  b1.test2("abc");  b1.test3("abc", "xyz");  C c1 = new C();  Integer s2 = c1.test1();  c1.test2(12);  c1.test3(10, 20);  System.out.println("Hello World!");  }  }  // Will it compilation successfull or not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
| **Correct Answer: A** | | |

|  |  |
| --- | --- |
| **19.** | class P  {  }  class Q extends P  {  }  class R extends Q  {  }  class S extends R  {  }  class T extends S  {  }  class A < X extends R >  {  X obj;  }  class M17  {  public static void main(String[] args)  {  A<R> a1 = new A<R>();  A<S> a2 = new A<S>();  A<T> a3 = new A<T>();  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: A** | | |
| **20.** | class P  {  }  class Q extends P  {  }  class A < X extends R >  {  X obj;  }  class M18  {  public static void main(String[] args)  {  A<P> a1 = new A<P>();  A<Q> a2 = new A<Q>();  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |

|  |  |
| --- | --- |
| **21.** | class A < X extends R>  {  X obj;  }  class M19  {  public static void main(String[] args)  {  A<Integer> a1 = new A<Integer>();  A<String> a2 = new A<String>();  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |
| **22.** | class A < X extends Number>  {  X obj;  }  class M20  {  public static void main(String[] args)  {  A<Number> a1 = new A<Number>();  A<Byte> a2 = new A<Byte>();  A<Short> a3 = new A<Short>();  A<Integer> a4 = new A<Integer>();  A<Long> a5 = new A<Long>();  A<Float> a6 = new A<Float>();  A<Double> a7 = new A<Double>();  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: A** | | |

|  |  |
| --- | --- |
| **23.** | class A < X extends Number>  {  X obj;  }  class M21  {  public static void main(String[] args)  {  A<String> a1 = new A<String>();  A<Thread> a2 = new A<Thread>();  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |
| **24.** | class P  {  }  class Q extends P  {  }  class A < X extends Number>  {  X obj;  }  class M22  {  public static void main(String[] args)  {  A<P> a1 = new A<P>();  A<Q> a2 = new A<Q>();  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |

|  |  |
| --- | --- |
| **25.** | class P  {  }  class A < X >  {  X obj;  }  class M23  {  public static void main(String[] args)  {  A <?> a1 = null;  a1 = new A<Integer>();  a1 = new A<Object>();  a1 = new A<P>();  a1 = new A<String>();  a1 = new A<Thread>();  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: C** | | |
| **26.** | class P  {  }  class A < X >  {  X obj;  }  class M24  {  static void test(A<?> a1)  {  }  public static void main(String[] args)  {  test(new A<Integer>());  test(new A<Object>());  test(new A<P>());  test(new A<String>());  test(new A<Thread>());  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: C** | | |

|  |  |
| --- | --- |
| **27.** | class A < X >  {  X obj;  }  class M25  {  public static void main(String[] args)  {  A <? extends Number> a1 = null;  a1 = new A<Number>();  a1 = new A<Integer>();  a1 = new A<Byte>();  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: A** | | |
| **28.** | class P  {  }  class A < X >  {  X obj;  }  class M26  {  public static void main(String[] args)  {  A <? extends Number> a1 = null;  a1 = new A<P>();  a1 = new A<String>();  a1 = new A<Object>();  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: A** | | |

|  |  |
| --- | --- |
| **29.** | class P  {  }  class Q extends P  {  }  class R extends Q  {  }  class S extends R  {  }  class T extends S  {  }  class A < X >  {  X obj;  }  class M27  {  public static void main(String[] args)  {  A <? extends Q> a1 = null;  a1 = new A<Q>();  a1 = new A<R>();  a1 = new A<S>();  a1 = new A<T>();  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: C** | | |
| **30.** | class P  {  }  class A < X >  {  X obj;  }  class M28  {  public static void main(String[] args)  {  A <? extends Q> a1 = null;  a1 = new A<P>();  a1 = new A<Object>();  a1 = new A<String>();  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: A** | | |

|  |  |
| --- | --- |
| **31.** | class P  {  }  class Q extends P  {  }  class A < X >  {  X obj;  }  class M29  {  public static void main(String[] args)  {  A <? super Q> a1 = null;  a1 = new A<Q>();  a1 = new A<P>();  a1 = new A<Object>();  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: C** | | |
| **32.** | class P  {  }  class Q extends P  {  }  class R extends Q  {  }  class S extends R  {  }  class A < X >  {  X obj;  }  class M30  {  public static void main(String[] args)  {  A <? super Q> a1 = null;  a1 = new A<R>();  a1 = new A<S>();  a1 = new A<String>();  a1 = new A<Number>();  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: A** | | |

|  |  |
| --- | --- |
| **33.** | class A < X >  {  X obj;  }  class M31  {  public static void main(String[] args)  {  A <? super Number> a1 = null;  a1 = new A<Number>();  a1 = new A<Object>();  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: A** | | |
| **34.** | class A < X >  {  X obj;  }  class M32  {  public static void main(String[] args)  {  A <? super Number> a1 = null;  a1 = new A<Integer>();  a1 = new A<String>();  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |

|  |  |
| --- | --- |
| **35.** | class A < X >  {  X obj;  }  class M33  {  public static void main(String[] args)  {  A <Integer> a1 = null;  a1 = new A<Integer>();  //a1 = new A<String>();  a1.obj = 30;  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: A** | | |
| **36.** | class A < X >  {  X obj;  }  class M34  {  public static void main(String[] args)  {  A < ? > a1 = null;  a1 = new A<Integer>();    a1.obj = 30;  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |

|  |  |
| --- | --- |
| **37.** | class P  {  }  class A < X >  {  X obj;  }  class M35  {  public static void main(String[] args)  {  A < ? > a1 = null;  a1 = new A<P>();  a1.obj = new P();  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |
| **38.** | class P  {  }  class A < X >  {  void test(X arg)  {  }  }  class M36  {  public static void main(String[] args)  {  A < ? > a1 = null;  a1 = new A<P>();  a1.test(new P());  A<Integer> a2 = null;  a2 = new A<Integer>();  a2.test(90);  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: A** | | |

|  |  |
| --- | --- |
| **39.** | class A < X >  {  void test(X arg)  {  }  }  class M37  {  public static void main(String[] args)  {  A < ? extends Number > a1 = null;  a1 = new A<Integer>();  a1.test(10);  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: A** | | |
| **40.** | class P  {  }  class A < X >  {  X obj;  }  class M38  {  public static void main(String[] args)  {  A < ? extends P> a1 = null;  a1 = new A<P>();  a1.obj = new P();  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |

|  |  |
| --- | --- |
| **41.** | class P  {  }  class Q extends P  {  }  class R extends Q  {  }  class S extends R  {  }  class T extends S  {  }  class A < X >  {  X obj;  }  class M39  {  public static void main(String[] args)  {  A <? super S> a1 = null;  a1 = new A<S>();  a1 = new A<R>();  a1 = new A<Q>();  a1 = new A<P>();  a1 = new A<Object>();    a1.obj = new S();  a1.obj = new T();  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: C** | | |
| **42.** | class A < T >  {  T obj;  }  class M40  {  public static void main(String[] args)  {  A<?> a1 = null;  a1 = new A<Number>();  a1 = new A<String>();  a1 = new A<Integer>();  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: C** | | |

|  |  |
| --- | --- |
| **43.** | class A < T >  {  T obj;  }  class M41  {  public static void main(String[] args)  {  A<?> a1 = null;  a1 = new A<Integer>();  a1.obj = 20;  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: A** | | |
| **44.** | class A < T >  {  T obj;  }  class M42  {  static void test(A<?> arg)  {  }  public static void main(String[] args)  {  test(new A<Integer>());  test(new A<String>());  test(new A<Object>());  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: C** | | |

|  |  |
| --- | --- |
| **45.** | class A < T >  {  T obj;  }  class M43  {  static void test(A<?> arg)  {  arg.obj = "abc";  }  public static void main(String[] args)  {  test(new A<String>());  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |
| **46.** | class A < T >  {  void test(T obj)  {  }  }  class M44  {  public static void main(String[] args)  {  A<?> a1 = new A<String>();  a1.test("abc");  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |

|  |  |
| --- | --- |
| **47.** | class A < T >  {  T obj;  }  class M45  {  public static void main(String[] args)  {  A<? extends Number> a1 = null;  a1 = new A<String>();  a1 = new A<Number>();  a1 = new A<Integer>();  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: A** | | |
| **48.** | class A < T >  {  T obj;  }  class M46  {  public static void main(String[] args)  {  A<? extends Number> a1 = null;  a1 = new A<Object>();  a1 = new A<Number>();  a1 = new A<Integer>();  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: A** | | |

|  |  |
| --- | --- |
| **49.** | class A < T >  {  T obj;  }  class M47  {  static void test(A<? extends Number> arg)  {  }  public static void main(String[] args)  {  test(new A<Number>());  test(new A<Integer>());  test(new A<Byte>());  test(new A<String>());  test(new A<Object>());  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: A** | | |
| **50.** | class A < T >  {  T obj;  }  class M48  {  public static void main(String[] args)  {  A<? extends Number> a1 = new A<Integer>();  a1.obj = 9000;  System.out.println("done");  }  }  // Will it compilation successfull or Not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
| **Correct Answer: B** | | |

|  |  |
| --- | --- |
| **51.** | class A < T >  {  T obj;  }  class M49  {  static void test(A<? extends Number> arg)  {  arg.obj = 10;  }  }  // Will it compilation successfull or Not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
| **Correct Answer: B** | | |
| **52.** | class A < T >  {  void method(T obj)  {  }  }  class M50  {  static void test(A<? extends Number> arg)  {  arg.method(new Double(1.5));  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |

|  |  |
| --- | --- |
| **53.** | class A < T >  {  T obj;  }  class M51  {  public static void main(String[] args)  {  A<? super Number> a1 = null;  a1 = new A<Number>();  a1 = new A<Object>();  a1 = new A<Integer>();  a1 = new A<String>();  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |
| **54.** | class A < T >  {  T obj;  }  class M52  {  static void test(A<? super Number> arg)  {  }  public static void main(String[] args)  {  test(new A<Number>());  test(new A<Object>());  test(new A<Integer>());  test(new A<String>());  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |

|  |  |
| --- | --- |
| **55.** | class A < T >  {  T obj;  }  class M53  {  public static void main(String[] args)  {  A<? super Number> a1 = null;  a1 = new A<Number>();  a1 = new A<Object>();  a1.obj = 100;  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: A** | | |
| **56.** | class A < T >  {  void test(T obj)  {  }  }  class M54  {  public static void main(String[] args)  {  A<? super Number> a1 = null;  a1 = new A<Number>();  a1 = new A<Object>();  a1.test(45.5);  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: A** | | |

|  |  |
| --- | --- |
| **57.** | class A < T >  {  T var;  }  class M55  {  public static void main(String[] args)  {  A<int> a1 = new A<int>();  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |
| **58.** | class A < T >  {  static T obj;  static void test(T arg)  {  }  }  class M56  {  public static void main(String[] args)  {  System.out.println("Hello World!");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |

|  |  |
| --- | --- |
| **59.** | interface A < T >  {  T obj;  }  class M57  {  public static void main(String[] args)  {  System.out.println("Hello World!");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |
| **60.** | class A < T >  {  T obj;  }  class M58  {  public static void main(String[] args)  {  A<Integer> a1 = new A<Integer>();  a1.obj = 1000;  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: A** | | |

|  |  |
| --- | --- |
| **61.** | class A < T >  {  T obj;  }  class M59  {  public static void main(String[] args)  {  A<Integer> a1 = new A<String>();  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: B** | | |
| **62.** | class A < T >  {  T obj;  }  class M60  {  public static void main(String[] args)  {  A<Integer> a1 = new A<>();  a1.obj = 2000;  A< ? > a2 = new A<>();  A< ? extends Number > a3 = new A<>();  A< ? super Number > a4 = new A<>();  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: A** | | |

|  |  |
| --- | --- |
| **63.** | class A < T >  {  T obj;  }  class M61  {  public static void main(String[] args)  {  A<Integer> a1 = new A<>();  A a2 = new A();  a1 = a2;  a2 = a1;  a1.obj = 1000;  a2.obj = 1000;  System.out.println("done");  }  } |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | | |
| **Correct Answer: A** | | |
| **64.** | class A < T >  {  T obj;  }  class M62  {  public static void main(String[] args)  {  A<Integer> a1 = new A<>();  A a2 = new A();  a1 = a2;  a2 = a1;  a1.obj = 1000;  a2.obj = 1000;  int i = a1.obj;  //int j = a2.obj;  int j = (Integer) a2.obj;  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: C** | | |

|  |  |
| --- | --- |
| **65.** | class A  {  static <T> void test(T obj1, T obj2)  {  }  }  class M63  {  public static void main(String[] args)  {  A.<Integer>test(100, 200);  A.<String>test("abc", "xyz");  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: C** | | |
| **66.** | class A  {  static <T> void test(T obj1, T obj2)  {  }  }  class M64  {  public static void main(String[] args)  {  A.test(100, 200);  A.test("abc", "xyz");  A.test("abc", 5000);  A.test(4.5, "test");  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: C** | | |

|  |  |
| --- | --- |
| **67.** | class A  {  static <T> void test(T obj1, T obj2)  {  }  }  class M65  {  public static void main(String[] args)  {  A.<Integer>test(100, 200);  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: C** | | |
| **68.** | class A  {  static <T> void test(T obj1, T obj2)  {  }  }  class M66  {  public static void main(String[] args)  {  A.<Integer>test(100, 200);  A.test(2000, "xyz");  System.out.println("done");  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **Correct Answer: C** | | |

|  |  |
| --- | --- |
| **69.** | class A  {  static <T, S> S test(T obj1, S obj2)  {  return obj2;  }  }  class M67  {  public static void main(String[] args)  {  String s1 = A.<Integer, String>test(100, "abc");  System.out.println("done with " + s1);  }  } |
| |  | | --- | | A.  done with abc |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error |  |  | | --- | | D.  done with 100 | | | |
| **Correct Answer: A** | | |
| **70.** | class A  {  <T, S> S test(T obj1, S obj2)  {  return obj2;  }  }  class M68  {  public static void main(String[] args)  {  A a1 = new A();  String s1 = a1.<Integer, String>test(100, "abc");  int i = a1.<Integer, Integer>test(100, 500);  System.out.println("done with " + s1);  System.out.println("done with " + i);  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done with abc  done with 500 |  |  | | --- | | D.  done with 500  done with abc | | | |
| **Correct Answer: C** | | |

|  |  |
| --- | --- |
| **71.** | import java.util.ArrayList;  class M69  {  public static void main(String[] args)  {  ArrayList<Integer> list = new ArrayList<>();  list.add(90);  list.add(910);  list.add(190);  list.add(901);  System.out.println(list);  int i = list.get(2);  System.out.println(i);  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  [90, 910, 190, 901]  2 |  |  | | --- | | D.  [90, 910, 190, 901]  190 | | | |
| **Correct Answer: D** | | |
| **72.** | import java.util.ArrayList;  import java.util.Collections;  class A implements Comparable < A >  {  int i;  A(int i)  {  this.i = i;  }  public String toString()  {  return "i = " + i;  }  public int compareTo(A obj)  {  return i - obj.i;  }  }  class M70  {  public static void main(String[] args)  {  ArrayList < A > list = new ArrayList<>();  list.add(new A(90));  list.add(new A(0));  list.add(new A(9));  list.add(new A(910));  list.add(new A(190));  System.out.println(list);  Collections.sort(list);  System.out.println(list);  }  } |
| |  | | --- | | A.  [i = 90, i = 0, i = 9, i = 910, i = 190]  classCastException |  |  | | --- | | B.  [i = 90, i = 0, i = 9, i = 910, i = 190]  [i = 0, i = 9, i = 90, i = 190, i = 910] |  |  | | --- | | C.  Compilation error |  |  | | --- | | D.  Runtime Error | | | |
| **Correct Answer: A** | | |

|  |  |
| --- | --- |
| **73.** | import java.util.ArrayList;  import java.util.Collections;  import java.util.Comparator;  class A  {  int i;  A(int i)  {  this.i = i;  }  public String toString()  {  return "i = " + i;  }  }  class B implements Comparator < A >  {  public int compare(A a1, A a2)  {  return a1.i - a2.i;  }  }  class M71  {  public static void main(String[] args)  {  ArrayList<A> list = new ArrayList<>();  list.add(new A(90));  list.add(new A(0));  list.add(new A(9));  list.add(new A(910));  list.add(new A(190));  System.out.println(list);  Collections.sort(list, new B());  System.out.println(list);  }  } |
| |  | | --- | | A.  [i = 90, i = 0, i = 9, i = 910, i = 190]  classCastException |  |  | | --- | | B.  [i = 90, i = 0, i = 9, i = 910, i = 190]  [i = 0, i = 9, i = 90, i = 190, i = 910] |  |  | | --- | | C.  Compilation error |  |  | | --- | | D.  Runtime Error | | | |
| **Correct Answer: B** | | |
| **74.** | import java.util.ArrayList;  import java.util.Collections;  import java.util.Comparator;  class A  {  int i, j;  A(int i, int j)  {  this.i = i;  this.j = j;  }  public String toString()  {  return "(" + i + "," + j + ")";  }  }  class M72  {  public static void main(String[] args)  {  ArrayList<A> list = new ArrayList<>();  list.add(new A(90, 10));  list.add(new A(0, 910));  list.add(new A(9, 0));  list.add(new A(910, 20));  list.add(new A(190, 500));  System.out.println(list);  Collections.sort(list, new Comparator<A>()  {  public int compare(A a1, A a2)  {  return a1.i - a2.i;  }  });  System.out.println(list);  Collections.sort(list, new Comparator<A>()  {  public int compare(A a1, A a2)  {  return a1.j - a2.j;  }  });  System.out.println(list);  }  } |
| |  | | --- | | A.  [(90,10), (0,910), (9,0), (910,20), (190,500)]  [(0,910), (9,0), (90,10), (190,500), (910,20)]  [(9,0), (90,10), (910,20), (190,500), (0,910)] |  |  | | --- | | B.  [(90,10), (0,910), (9,0), (910,20), (190,500)]  [(9,0), (90,10), (910,20), (190,500), (0,910)]  [(0,910), (9,0), (90,10), (190,500), (910,20)] |  |  | | --- | | C.  Compilation error |  |  | | --- | | D.  Runtime Error | | | |
| **Correct Answer: A** | | |

|  |  |
| --- | --- |
| **75.** | import java.util.ArrayList;  import java.util.Collections;  import java.util.Comparator;  class A  {  int i, j;  A(int i, int j)  {  this.i = i;  this.j = j;  }  public String toString()  {  return "(" + i + "," + j + ")";  }  }  class M73  {  public static void main(String[] args)  {  ArrayList<A> list = new ArrayList<>();  list.add(new A(90, 10));  list.add(new A(0, 910));  list.add(new A(9, 0));  list.add(new A(910, 20));  list.add(new A(190, 500));  System.out.println(list);  Collections.sort(list, (A a1, A a2) -> a1.i - a2.i);  System.out.println(list);  Collections.sort(list, (a1, a2) -> a1.j - a2.j);  System.out.println(list);  }  } |
| |  | | --- | | A.  [(90,10), (0,910), (9,0), (910,20), (190,500)]  [(9,0), (90,10), (910,20), (190,500), (0,910)]  [(0,910), (9,0), (90,10), (190,500), (910,20)] |  |  | | --- | | B.  [(90,10), (0,910), (9,0), (910,20), (190,500)]  [(0,910), (9,0), (90,10), (190,500), (910,20)]  [(9,0), (90,10), (910,20), (190,500), (0,910)] |  |  | | --- | | C.  Compilation error |  |  | | --- | | D.  Runtime Error | | | |
| **Correct Answer: B** | | |
| **76.** | import java.util.HashMap;  class M74  {  public static void main(String[] args)  {  HashMap<String, Integer> map = new HashMap<>();  map.put("hello", 22);  map.put("abc", 12);  map.put("xyz", 42);  System.out.println(map);  int i = map.get("abc");  System.out.println(i);  }  } |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  {abc=12, xyz=42, hello=22}  12 |  |  | | --- | | D.  {abc=12, xyz=42, hello=22}  abc | | | |
| **Correct Answer: C** | | |

|  |  |
| --- | --- |
| **77.** | import java.util.ArrayList;  class M75  {  static void test(ArrayList<?> list)  {  list.add(90);  }  public static void main(String[] args)  {  ArrayList<Integer> list = new ArrayList<>();  list.add(10);  test(list);  list.add(20);  System.out.println(list);  }  }  // Will it compiles successfull or not? |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | | |
| **Correct Answer: B** | | |

Bottom of Form