**Core Java Advanced Part – 4**

  Generics

  basics (Total available questions : 23)

  advanced (Total available questions : 54)

  File Handling

  basics (Total available questions : 37)

  readers and writers (Total available questions : 9)

  streams

  serialization (Total available questions : 5)

  adv serialization and externalization

  PrintStream

  serialVersionUID

  JDK8.0 Features

  forEach() method in Iterable interface.

  default and static methods in Interfaces.

  Functional Interfaces and Lambda Expressions. (Total available questions : 3)

  double colon (::) Operator

  forEach in detailed

  Java Streams creation

  Java Streams Usage

  Java IO improvements.

  Java Date and Time API Improvement

  Collection API improvements

  Concurrency API improvements

|  |  |
| --- | --- |
| **1.** | package com.lara;  import java.io.File;  public class M1 {  public static void main(String[] args) throws Exception {  File f1 = new File("hello1.txt");  f1.createNewFile();  System.out.println("done");  }  }  Q765 |
| |  | | --- | | A.  done |  |  | | --- | | B.  RuntimeException |  |  | | --- | | C.  None | | |
| **2.** | package com.lara;  import java.io.File;  public class M2 {  public static void main(String[] args) throws Exception {  File f1 = new File("hello2.html");  f1.createNewFile();  System.out.println("done");  }  }  Q766 |
| |  | | --- | | A.  done |  |  | | --- | | B.  RuntimeException |  |  | | --- | | C.  compile Time Error |  |  | | --- | | D.  None | | |

|  |  |
| --- | --- |
| **3.** | package com.lara;  import java.io.File;  public class M3 {  public static void main(String[] args) throws Exception {  File f1 = new File("hello3.doc");  f1.createNewFile();  System.out.println("done");  }  }  Q767 |
| |  | | --- | | A.  done |  |  | | --- | | B.  RuntimeException |  |  | | --- | | C.  compile Time Error |  |  | | --- | | D.  None | | |
| **4.** | package com.lara;  import java.io.File;  public class M4 {  public static void main(String[] args) throws Exception {  File f1 = new File("hello4.java");  boolean flag = f1.createNewFile();  System.out.println("done:" + flag);  }  }  // assume you are running first time the what could be the outputQ768 |
| |  | | --- | | A.  done: true |  |  | | --- | | B.  done: false |  |  | | --- | | C.  compile Time Error |  |  | | --- | | D.  None | | |

|  |  |
| --- | --- |
| **5.** | package com.lara;  import java.io.File;  public class M4 {  public static void main(String[] args) throws Exception {  File f1 = new File("hello4.java");  boolean flag = f1.createNewFile();  System.out.println("done:" + flag);  }  }  // assume you are running second time the what could be the outputQ769 |
| |  | | --- | | A.  done: true |  |  | | --- | | B.  done: false |  |  | | --- | | C.  compile Time Error |  |  | | --- | | D.  None | | |
| **6.** | package com.lara;  import java.io.File;  public class M5 {  public static void main(String[] args) throws Exception {  File f1 = new File("hello5.txt");  System.out.println("a:" + f1.exists());  System.out.println("b:" + f1.createNewFile());  System.out.println("c:" + f1.exists());  System.out.println("done:");  }  }  // assume that you are running first time what could be the outputQ770 |
| |  | | --- | | A.  a:true  b:false  c:true  done: |  |  | | --- | | B.  a: false  b:true  c: true  done: |  |  | | --- | | C.  compile Time Error |  |  | | --- | | D.  None | | |

|  |  |
| --- | --- |
| **7.** | package com.lara;  import java.io.File;  public class M5 {  public static void main(String[] args) throws Exception {  File f1 = new File("hello5.txt");  System.out.println("a:" + f1.exists());  System.out.println("b:" + f1.createNewFile());  System.out.println("c:" + f1.exists());  System.out.println("done:");  }  }  // assume that you are running second time what could be the outputQ771 |
| |  | | --- | | A.  a:true  b:false  c:true  done: |  |  | | --- | | B.  a: false  b:true  c: true  done: |  |  | | --- | | C.  compile Time Error |  |  | | --- | | D.  None | | |
| **8.** | package com.lara;  import java.io.File;  import java.io.IOException;  public class M6 {  public static void main(String[] args) {  File f1 = new File("hello6.txt");  System.out.println("a:" + f1.exists());  try  {  System.out.println("b:" + f1.createNewFile());  }  catch(IOException ex)  {  ex.printStackTrace();  }  System.out.println("c:" + f1.exists());  System.out.println("done:");  }  }  // assume that you are running first time what could be the outputQ772 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  a:false  b:true  c:true  done: |  |  | | --- | | D.  None | | |

|  |  |  |
| --- | --- | --- |
| **9.** | package com.lara;  import java.io.File;  import java.io.IOException;  public class M6 {  public static void main(String[] args) {  File f1 = new File("hello6.txt");  System.out.println("a:" + f1.exists());  try  {  System.out.println("b:" + f1.createNewFile());  }  catch(IOException ex)  {  ex.printStackTrace();  }  System.out.println("c:" + f1.exists());  System.out.println("done:");  }  }  // assume that you are running second time what could be the outputQ773 | |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  a:false  b:true  c:true  done: |  |  | | --- | | D.  None | | | |
| **10.** | | package com.lara;  import java.io.File;  import java.io.IOException;  public class M7 {  public static void main(String[] args) {  File f1 = new File("E:\\hello7.txt");  System.out.println("a:" + f1.exists());  try  {  System.out.println("b:" + f1.createNewFile());  }  catch(IOException ex)  {  ex.printStackTrace();  }  System.out.println("c:" + f1.exists());  System.out.println("done:");  }  }  // assume that you are running first time what could be the outputQ774 |
| |  | | --- | | A.  a: false  b:true  c: true  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  compile Time Error |  |  | | --- | | D.  None | | | |

|  |  |
| --- | --- |
| **11.** | package com.lara;  import java.io.File;  import java.io.IOException;  public class M7 {  public static void main(String[] args) {  File f1 = new File("E:\\hello7.txt");  System.out.println("a:" + f1.exists());  try  {  System.out.println("b:" + f1.createNewFile());  }  catch(IOException ex)  {  ex.printStackTrace();  }  System.out.println("c:" + f1.exists());  System.out.println("done:");  }  }  // assume that you are running first time what could be the outputQ775 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  compile Time Error |  |  | | --- | | D.  None | | |
| **12.** | package com.lara;  import java.io.File;  import java.io.IOException;  public class M8 {  public static void main(String[] args) {  File f1 = new File("E:/hello8.txt");  System.out.println("a:" + f1.exists());  try  {  System.out.println("b:" + f1.createNewFile());  }  catch(IOException ex)  {  ex.printStackTrace();  }  System.out.println("c:" + f1.exists());  System.out.println("done:");  }  }  // assume that you are running first time what could be the outputQ776 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  compile Time Error |  |  | | --- | | D.  a:false  b:true  c:true  done: | | |

|  |  |
| --- | --- |
| **13.** | package com.lara;  import java.io.File;  import java.io.IOException;  public class M8 {  public static void main(String[] args) {  File f1 = new File("E:/hello8.txt");  System.out.println("a:" + f1.exists());  try  {  System.out.println("b:" + f1.createNewFile());  }  catch(IOException ex)  {  ex.printStackTrace();  }  System.out.println("c:" + f1.exists());  System.out.println("done:");  }  }  // assume that you are running second time what could be the outputQ777 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  compile Time Error |  |  | | --- | | D.  a:false  b:true  c:true  done: | | |
| **14.** | package com.lara;  import java.io.File;  import java.io.IOException;  public class M9 {  public static void main(String[] args) {  File f1 = new File("E://ocjp", "hello9.txt");  System.out.println("a:" + f1.exists());  try  {  System.out.println("b:" + f1.createNewFile());  }  catch(IOException ex)  {  ex.printStackTrace();  }  System.out.println("c:" + f1.exists());  System.out.println("done:");  }  }  // assume that you are running first time what could be the outputQ778 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  a:false  b:true  c:true  done: | | |

|  |  |
| --- | --- |
| **15.** | package com.lara;  import java.io.File;  import java.io.IOException;  public class M9 {  public static void main(String[] args) {  File f1 = new File("E://ocjp", "hello9.txt");  System.out.println("a:" + f1.exists());  try  {  System.out.println("b:" + f1.createNewFile());  }  catch(IOException ex)  {  ex.printStackTrace();  }  System.out.println("c:" + f1.exists());  System.out.println("done:");  }  }  // assume that you are running second time what could be the outputQ779 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  a:false  false  c:false  done: | | |
| **16.** | package com.lara;  import java.io.File;  import java.io.IOException;  public class M10 {  public static void main(String[] args) {  File dir = new File("E://ocjp");  File f1 = new File(dir, "hello10.txt");  System.out.println("a:" + f1.exists());  try  {  System.out.println("b:" + f1.createNewFile());  }  catch(IOException ex)  {  ex.printStackTrace();  }  System.out.println("c:" + f1.exists());  System.out.println("done:");  }  }  // assume that you are running first time what could be the outputQ780 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:false  b:true  c:true  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: | | |

|  |  |
| --- | --- |
| **17.** | package com.lara;  import java.io.File;  import java.io.IOException;  public class M10 {  public static void main(String[] args) {  File dir = new File("E://ocjp");  File f1 = new File(dir, "hello10.txt");  System.out.println("a:" + f1.exists());  try  {  System.out.println("b:" + f1.createNewFile());  }  catch(IOException ex)  {  ex.printStackTrace();  }  System.out.println("c:" + f1.exists());  System.out.println("done:");  }  }  // assume that you are running second time what could be the outputQ781 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  none | | |
| **18.** | package com.lara;  import java.io.File;  import java.io.IOException;  public class M11 {  public static void main(String[] args) {  File dir = new File("E://ocjp1");  File f1 = new File(dir, "hello11.txt");  System.out.println("a:" + f1.exists());  try  {  System.out.println("b:" + f1.createNewFile());  }  catch(IOException ex)  {  ex.printStackTrace();  }  System.out.println("c:" + f1.exists());  System.out.println("done:");  }  }  // assume that you are running first time what could be the outputQ782 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:false  b:true  c:true  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: | | |

|  |  |
| --- | --- |
| **19.** | package com.lara;  import java.io.File;  import java.io.IOException;  public class M11 {  public static void main(String[] args) {  File dir = new File("E://ocjp1");  File f1 = new File(dir, "hello11.txt");  System.out.println("a:" + f1.exists());  try  {  System.out.println("b:" + f1.createNewFile());  }  catch(IOException ex)  {  ex.printStackTrace();  }  System.out.println("c:" + f1.exists());  System.out.println("done:");  }  }  // assume that you are running second time what could be the outputQ783 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  none | | |
| **20.** | package com.lara;  import java.io.File;  public class M12 {  public static void main(String[] args) {  File f1 = new File("dir1");  f1.mkdir();  System.out.println("done");  }  }  // assume that you are running first time what could be the outputQ784 |
| |  | | --- | | A.  done |  |  | | --- | | B.  RuntimeException |  |  | | --- | | C.  None | | |

|  |  |
| --- | --- |
| **21.** | package com.lara;  import java.io.File;  public class M12 {  public static void main(String[] args) {  File f1 = new File("dir1");  f1.mkdir();  System.out.println("done");  }  }  // assume that you are running second time what could be the outputQ785 |
| |  | | --- | | A.  done |  |  | | --- | | B.  RuntimeException |  |  | | --- | | C.  None | | |
| **22.** | package com.lara;  import java.io.File;  public class M13 {  public static void main(String[] args) {  File f1 = new File("dir2");  System.out.println("a:" + f1.exists());  System.out.println("b:" + f1.mkdir());  System.out.println("c:" + f1.exists());  System.out.println("done");  }  }  // assume that you are running first time what could be the outputQ786 |
| |  | | --- | | A.  a: false  b:true  c: true  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  none | | |

|  |  |
| --- | --- |
| **23.** | package com.lara;  import java.io.File;  public class M13 {  public static void main(String[] args) {  File f1 = new File("dir2");  System.out.println("a:" + f1.exists());  System.out.println("b:" + f1.mkdir());  System.out.println("c:" + f1.exists());  System.out.println("done");  }  }  // assume that you are running second time what could be the outputQ787 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  none | | |
| **24.** | package com.lara;  import java.io.File;  public class M14 {  public static void main(String[] args) {  File f1 = new File("D:/java8.0/dir3");  System.out.println("a:" + f1.exists());  System.out.println("b:" + f1.mkdir());  System.out.println("c:" + f1.exists());  System.out.println("done");  }  }  // assume this path is available in your system then you run the program for the first time what could be the outputQ788 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  a:false  b:false  c:false  done |  |  | | --- | | D.  a:false  b:true  c:true  done: |  |  | | --- | | E.  None | | |

|  |  |
| --- | --- |
| **25.** | package com.lara;  import java.io.File;  public class M14 {  public static void main(String[] args) {  File f1 = new File("D:/java8.0/dir3");  System.out.println("a:" + f1.exists());  System.out.println("b:" + f1.mkdir());  System.out.println("c:" + f1.exists());  System.out.println("done");  }  }  // assume this path is available in your system then you run the program for the second time what could be the outputQ789 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  a:false  b:false  c:false  done |  |  | | --- | | D.  a:false  b:true  c:true  done: |  |  | | --- | | E.  None | | |
| **26.** | package com.lara;  import java.io.File;  public class M15 {  public static void main(String[] args) {  File f1 = new File("D:/java8.0/bin", "dir4");  System.out.println("a:" + f1.exists());  System.out.println("b:" + f1.mkdir());  System.out.println("c:" + f1.exists());  System.out.println("done");  }  }  // assume this path is available in your system then you run the program for the first time time what could be the outputQ790 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  a:false  b:false  c:false  done |  |  | | --- | | D.  a:false  b:true  c:true  done: |  |  | | --- | | E.  None | | |

|  |  |
| --- | --- |
| **27.** | package com.lara;  import java.io.File;  public class M15 {  public static void main(String[] args) {  File f1 = new File("D:/java8.0/bin", "dir4");  System.out.println("a:" + f1.exists());  System.out.println("b:" + f1.mkdir());  System.out.println("c:" + f1.exists());  System.out.println("done");  }  }  // assume this path is available in your system then you run the program for the second time what could be the outputQ791 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  a:false  b:false  c:false  done |  |  | | --- | | D.  a:false  b:true  c:true  done: |  |  | | --- | | E.  None | | |
| **28.** | package com.lara;  import java.io.File;  public class M16 {  public static void main(String[] args) {  File dir = new File("D:/java8.0/bin");  File f1 = new File(dir, "dir6");  System.out.println("a:" + f1.exists());  System.out.println("b:" + f1.mkdir());  System.out.println("c:" + f1.exists());  System.out.println("done");  }  }  // assume this path is available in your system then you run the program for the first time time what could be the outputQ792 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  a:false  b:false  c:false  done |  |  | | --- | | D.  a:false  b:true  c:true  done: |  |  | | --- | | E.  None | | |

|  |  |
| --- | --- |
| **29.** | package com.lara;  import java.io.File;  public class M16 {  public static void main(String[] args) {  File dir = new File("D:/java8.0/bin");  File f1 = new File(dir, "dir6");  System.out.println("a:" + f1.exists());  System.out.println("b:" + f1.mkdir());  System.out.println("c:" + f1.exists());  System.out.println("done");  }  }  // assume this path is available in your system then you run the program for the second time time what could be the outputQ793 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  a:false  b:false  c:false  done |  |  | | --- | | D.  a:false  b:true  c:true  done: |  |  | | --- | | E.  None | | |
| **30.** | package com.lara;  import java.io.File;  public class M17 {  public static void main(String[] args) {  File dir = new File("D:/java8.0/xyz");  File f1 = new File(dir, "dir6");  System.out.println("a:" + f1.exists());  System.out.println("b:" + f1.mkdir());  System.out.println("c:" + f1.exists());  System.out.println("done");  }  }  // assume this path is available in your system then you run the program for the first time what could be the outputQ794 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  a:false  b:true  c:true  done: |  |  | | --- | | D.  none | | |

|  |  |
| --- | --- |
| **31.** | package com.lara;  import java.io.File;  public class M17 {  public static void main(String[] args) {  File dir = new File("D:/java8.0/xyz");  File f1 = new File(dir, "dir6");  System.out.println("a:" + f1.exists());  System.out.println("b:" + f1.mkdir());  System.out.println("c:" + f1.exists());  System.out.println("done");  }  }  // assume this path is available in your system then you run the program for the second time time what could be the outputQ795 |
| |  | | --- | | A.  a: false  b:true  c: false  done: |  |  | | --- | | B.  a:true  b:false  c:true  done: |  |  | | --- | | C.  a:false  b:true  c:true  done: |  |  | | --- | | D.  none | | |
| **32.** | package com.lara;  import java.io.File;  public class M18 {  public static void main(String[] args) {  File f1 = new File("hello1.txt");  File f2 = new File("dir1");  System.out.println("f1.isFile():" + f1.isFile());  System.out.println("f2.isFile():" + f2.isFile());  System.out.println("f1.isDirectory:" + f1.isDirectory());  System.out.println("f2.isDirectory:" + f2.isDirectory());  }  }  Q796 |
| |  | | --- | | A.  f1.isFile():true  f2.isFile():false  f1.isDirectory:false  f2.isDirectory:true |  |  | | --- | | B.  f1.isFile():false  f2.isFile():true  f1.isDirectory:true  f2.isDirectory:false |  |  | | --- | | C.  None | | |

|  |  |
| --- | --- |
| **33.** | package com.lara;  import java.io.File;  public class M19 {  public static void main(String[] args) {  File f1 = new File("D:\\java8.0\\bin");  File[] members = f1.listFiles();  for(File file : members)  {  System.out.println(file.getName());  }  }  }  // assume this path is available in your system then you run the program then you will get list of files names or not?Q797 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |
| **34.** | package com.lara;  import java.io.File;  public class M20 {  public static void main(String[] args) {  File src = new File("F:\\may7th");  File[] members = src.listFiles();  for(File member : members)  {  System.out.println(member.getName() + ":" + (member.isFile() ? " file " : " directory")) ;  }  }  }  // assume this path is available in your system then you run the program then you will get list of files names or not?Q798 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |

|  |  |
| --- | --- |
| **35.** | package com.lara;  import java.io.File;  public class M21 {  public static void main(String[] args) {  File src = new File("F:\\may7th");  File[] members = src.listFiles();  int dirCount = 0, fileCount = 0;  for(File member : members)  {  if(member.isFile())  {  fileCount ++;  }  else  {  dirCount ++;  }  }  System.out.println("files count:" + fileCount) ;  System.out.println("dir count:" + dirCount);  }  }  // assume this path is available in your system then you run the program then you will get count of files and directories or not?Q799 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |
| **36.** | package com.lara;  import java.io.File;  import java.util.HashMap;  import java.util.Scanner;  public class M22 {  public static void main(String[] args) {  Scanner sc = new Scanner(System.in);  System.out.println("Enter source path");  String src = sc.next();  File srcDir = new File(src);  HashMap<String, Integer> map = new HashMap<String, Integer>();  File[] srcDirMembers = srcDir.listFiles();    String fileType;  for(File file : srcDirMembers)  {  if(file.isFile())  {  fileType = file.getName().substring(file.getName().indexOf(".") + 1);  if(map.get(fileType) == null)  {  map.put(fileType, 1);  }  else  {  map.put(fileType, map.get(fileType) + 1);  }  }  System.out.println(file.getName() + ":" + map);  }  System.out.println("final map:" + map);  System.out.println("total members:" + srcDirMembers.length);  }  }  // assume  F:\may7th  you give the input of this path and is available in your system then you run the program then you will get all the files and directories or not?Q800 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |

|  |  |
| --- | --- |
| **37.** | package com.lara;  import java.io.File;  import java.util.HashMap;  import java.util.Scanner;  public class M23 {    static HashMap<String, Integer> map = new HashMap<String, Integer>();  static HashMap<String, Integer> dirMap = new HashMap<String, Integer>();    public static void main(String[] args) {  Scanner sc = new Scanner(System.in);  System.out.println("Enter source path");  String src = sc.next();  File srcDir = new File(src);  File[] srcDirMembers = srcDir.listFiles();  String fileType;  for(File file : srcDirMembers)  {  if(file.isFile())  {  countFile(file);  }  else  {  dirMap = new HashMap<String, Integer>();  processDir(file);  System.out.println(file.getName() + ":" + dirMap);  }  }  System.out.println(srcDir + ":" + map);  }  static void countFile(File f1)  {  String fileType = f1.getName().substring(f1.getName().indexOf(".") + 1);  if(map.get(fileType) == null)  {  map.put(fileType, 1);  }  else  {  map.put(fileType, map.get(fileType) + 1);  }  if(dirMap.get(fileType) == null)  {  dirMap.put(fileType, 1);  }  else  {  dirMap.put(fileType, dirMap.get(fileType) + 1);  }  }  static void processDir(File dir)  {  File[] srcDirMembers = dir.listFiles();  for(File file : srcDirMembers)  {  if(file.isFile())  {  countFile(file);  }  else  {  processDir(file);  }  }  }  }  // assume  F:\may7th  you give the input of this path and is available in your system then you run the program then you will get all the directories with directories and files inside it or not?Q801 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |
| **38.** | package io2;  import java.io.File;  import java.io.FileWriter;  import java.io.IOException;  public class M1 {  public static void main(String[] args) throws IOException{  File f1 = new File("test1.txt");  FileWriter out = new FileWriter(f1);  out.write("Hello");  out.write("test");  out.write("abc");  out.write("xyz");  out.write("end");  out.flush();  out.close();  System.out.println("done");  }  }  Q802 |
| |  | | --- | | A.  done |  |  | | --- | | B.  throws Exception | | |

|  |  |
| --- | --- |
| **39.** | package io2;  import java.io.File;  import java.io.FileWriter;  import java.io.IOException;  public class M2 {  public static void main(String[] args) throws IOException{  File f1 = new File("test2.txt");  FileWriter out = new FileWriter(f1);  out.write("Hello");  out.write("test");  out.write("abc");  out.write("\n");  out.write("xyz");  out.write("end");  out.flush();  out.close();  System.out.println("done");  }  }  Q803 |
| |  | | --- | | A.  done |  |  | | --- | | B.  throws Exception | | |
| **40.** | package io2;  import java.io.File;  import java.io.FileWriter;  import java.io.IOException;  public class M3 {  public static void main(String[] args) throws IOException{  FileWriter out = new FileWriter("test3.txt", true);  out.write("Hello4444444");  out.write("test");  out.write("abc444444");  out.write("\n");  out.write("xyz");  out.write("end444444444");  out.flush();  out.close();  System.out.println("done");  }  }  Q804 |
| |  | | --- | | A.  done |  |  | | --- | | B.  throws Exception |  |  | | --- | | C.  true | | |

|  |  |
| --- | --- |
| **41.** | package io2;  import java.io.File;  import java.io.FileWriter;  import java.io.IOException;  public class M4 {  public static void main(String[] args) {  File f1 = new File("test4.txt");  FileWriter out = null;  try  {  out = new FileWriter(f1);  out.write("hello");  out.write("test");  out.write("end");  }  catch(IOException ex)  {  ex.printStackTrace();  }  finally  {  if(out != null)  {  try  {  out.flush();  out.close();  }  catch(IOException ex)  {  ex.printStackTrace();  }  }  }    System.out.println("done");          }  }  //Whether it compiles successfull or not?Q805 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |
| **42.** | package io2;  import java.io.File;  import java.io.FileWriter;  import java.io.IOException;  public class M5 {  public static void main(String[] args) {  File f1 = new File("test5.txt");  try (FileWriter out = new FileWriter(f1)) //JDK1.7  {  out.write("hello");  out.write("test");  out.write("end");  }  catch(IOException ex)  {  ex.printStackTrace();  }  }  }  // Whether it compiles successfull are not?Q806 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |

|  |  |
| --- | --- |
| **43.** | package io2;  import java.io.File;  import java.io.FileReader;  import java.io.IOException;  public class M6 {  public static void main(String[] args) throws IOException {  File f1 = new File("test3.txt");  char[] chars = new char[50];  FileReader fin = new FileReader(f1);  fin.read(chars);  fin.close();  String s1 = new String(chars);  System.out.println(s1);  }  }  // assume test3.txt fille in your system in that file we wrote some content by using the following comands  out.write("Hello4444444");  out.write("test");  out.write("abc444444");  out.write("\n");  out.write("xyz");  out.write("end444444444");  then what could be the outputQ807 |
| |  | | --- | | A.  xyzend444444444Hello4444  Hello4444444testabc444444 |  |  | | --- | | B.  Hello4444444testabc444444  xyzend444444444Hello4444 |  |  | | --- | | C.  throws Exception | | |
| **44.** | package io2;  import java.io.File;  import java.io.FileReader;  import java.io.IOException;  public class M7 {  public static void main(String[] args)  {  File f1 = new File("test3.txt");  FileReader fin = null;  char[] chars = new char[110];  try  {  fin = new FileReader(f1);  fin.read(chars);  }  catch(IOException ex)  {  ex.printStackTrace();  }  finally  {  if(fin != null)  {  try  {  fin.close();  }  catch(IOException ex)  {  ex.printStackTrace();  }  }  }  String s1 = new String(chars);  System.out.println(s1);    }  }  // assume test3.txt fille in your system in that file we wrote some content two times by using the following comands  out.write("Hello4444444");  out.write("test");  out.write("abc444444");  out.write("\n");  out.write("xyz");  out.write("end444444444");  then what could be the outputQ808 |
| |  | | --- | | A.  xyzend444444444Hello4444  Hello4444444testabc444444 |  |  | | --- | | B.  Hello4444444testabc444444  xyzend444444444Hello4444 |  |  | | --- | | C.  Hello4444444testabc444444  xyzend444444444Hello4444444testabc444444  xyzend444444444 |  |  | | --- | | D.  Hello4444444testabc444444  xyzend444444444  Hello4444444testabc444444  xyzend444444444 | | |

|  |  |
| --- | --- |
| **45.** | package io2;  import java.io.File;  import java.io.FileReader;  import java.io.IOException;  public class M8 {  public static void main(String[] args)  {  File f1 = new File("test3.txt");    try (FileReader fin = new FileReader(f1))  {  char[] chars = new char[110];  fin.read(chars);  String s1 = new String(chars);  System.out.println(s1);  }  catch(IOException ex)  {  ex.printStackTrace();  }  }  }  Q809 |
| |  | | --- | | A.  Hello4444444testabc444444  xyzend444444444Hello4444444testabc444444  xyzend444444444 |  |  | | --- | | B.  Hello4444444testabc444444  xyzend444444444  Hello4444444testabc444444  xyzend444444444 |  |  | | --- | | C.  throws Exception |  |  | | --- | | D.  Compile Time Error | | |
| **46.** | package io2;  import java.io.BufferedWriter;  import java.io.FileWriter;  import java.io.IOException;  public class M9 {  public static void main(String[] args) throws IOException {  FileWriter out = new FileWriter("test22.txt", true);  BufferedWriter bout = new BufferedWriter(out);  bout.write("test");  bout.write("test");  bout.newLine();  bout.write("test");  bout.write("test");  bout.newLine();  bout.write("test");  bout.write("test");  bout.newLine();  bout.write("test");  bout.write("test");  bout.flush();  bout.close();  out.close();  System.out.println("done");    }  }  Q810 |
| |  | | --- | | A.  done |  |  | | --- | | B.  throws Exception | | |

|  |  |
| --- | --- |
| **47.** | package com.lara;  import java.io.Serializable;  public class B implements Serializable{  int i;  transient int j;  static int k;  B(int i, int j, int k)  {  this.i = i;  this.j = j;  this.k = k;  }  @Override  public String toString() {  return "(" + i + ", " + j + ", " + k + ")";  }  }  //whether it compiles successfully or not?Q811 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |
| **48.** | package com.lara;  import java.io.Serializable;  class C implements Serializable  {  int i;  }  public class D extends C{  int j;  D(int i, int j)  {  this.i = i;  this.j = j;  }  @Override  public String toString() {  return "(" + i + ", " + j + ")";  }  }  //whether it compiles successfully or not?Q812 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |

|  |  |
| --- | --- |
| **49.** | package com.lara;  import java.io.Serializable;  class E  {  int i;  E()  {  i = 1;  System.out.println("E()");  }  }  class F extends E  {  int j;  F()  {  j = 2;  System.out.println("F()");  }  }  public class G extends F implements Serializable{  int k;  G()  {  k = 3;  System.out.println("G()");  }  @Override  public String toString() {  return "(" + i + ", " + j + ", " + k + ")";  }  }  //whether it compiles successfully or not?Q813 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |
| **50.** | package com.lara;  import java.io.IOException;  import java.io.ObjectInputStream;  import java.io.ObjectOutputStream;  import java.io.Serializable;  public class H implements Serializable  {  int x;  double y;  String z;    H(int x, double y, String z)  {  this.x = x;  this.y = y;  this.z = z;  }    @Override  public String toString() {  return "(" + x + ", " + y + ", " + z + ")";  }    private void writeObject(ObjectOutputStream out) throws IOException  {  System.out.println("i am from writeObject");  out.defaultWriteObject();  out.writeInt(5000);  }    private void readObject(ObjectInputStream in) throws IOException, ClassNotFoundException  {  System.out.println("i am from readObject");  in.defaultReadObject();  int data = in.readInt();  System.out.println(data);  }    }  //whether it compiles successfully or not?Q814 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |

|  |  |
| --- | --- |
| **51.** | package com.lara;  import java.io.Externalizable;  import java.io.IOException;  import java.io.ObjectInput;  import java.io.ObjectInputStream;  import java.io.ObjectOutput;  import java.io.ObjectOutputStream;  import java.io.Serializable;  public class I implements Externalizable  {  int x;  double y;  String z;  I(int x, double y, String z)  {  this.x = x;  this.y = y;  this.z = z;  }  @Override  public String toString() {  return "(" + x + ", " + y + ", " + z + ")";  }  @Override  public void writeExternal(ObjectOutput out) throws IOException {  out.writeInt(x);  //out.writeDouble(y);  out.writeUTF(z);  }    @Override  public void readExternal(ObjectInput in) throws IOException, ClassNotFoundException {  x = in.readInt();  //y = in.readDouble();  z = in.readUTF();  }  }  Q815 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |
| **52.** | Generics are used for changing data type of any members from one object to another object.Q1415 |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | |

|  |  |
| --- | --- |
| **53.** | We can use generics to change field data type from one object to another object.Q1416 |
| |  | | --- | | A.  yes |  |  | | --- | | B.  no | | |
| **54.** | In which version of JDK, generics are introduced?Q1417 |
| |  | | --- | | A.  JDK1.8 |  |  | | --- | | B.  JDK1.5 |  |  | | --- | | C.  JDK1.6 | | |

|  |  |
| --- | --- |
| **55.** | 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");  }  }  Q1480 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |
| **56.** | 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");  }  }  Q1481 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |

|  |  |
| --- | --- |
| **57.** | 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 notQ1482 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |
| **58.** | 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 notQ1483 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |

|  |  |
| --- | --- |
| **59.** | 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 notQ1484 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |
| **60.** | 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?Q1485 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |

|  |  |
| --- | --- |
| **61.** | 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!");  }  }  Q1486 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  Hello World | | |
| **62.** | 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!");  }  }  Q1487 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  Hello World | | |

|  |  |
| --- | --- |
| **63.** | 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?Q1488 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |
| **64.** | 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?Q1489 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |

|  |  |
| --- | --- |
| **65.** | 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?Q1490 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |
| **66.** | 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?Q1491 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |

|  |  |
| --- | --- |
| **67.** | 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");  }  }  Q1492 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  done |  |  | | --- | | C.  Run time Error | | |
| **68.** | 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");  }  }  Q1493 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |

|  |  |
| --- | --- |
| **69.** | 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?Q1494 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |
| **70.** | 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");  }  }  Q1495 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |

|  |  |
| --- | --- |
| **71.** | 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");  }  }  Q1496 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |
| **72.** | 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");  }  }  Q1497 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |

|  |  |
| --- | --- |
| **73.** | 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");  }  }  Q1498 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |
| **74.** | 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");  }  }  Q1499 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |

|  |  |
| --- | --- |
| **75.** | 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");  }  }  Q1500 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |
| **76.** | 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");  }  }  Q1501 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |

|  |  |
| --- | --- |
| **77.** | 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");  }  }  Q1502 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |
| **78.** | 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");  }  }  Q1503 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |

|  |  |
| --- | --- |
| **79.** | 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");  }  }  Q1504 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |
| **80.** | 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");  }  }  Q1505 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |

|  |  |
| --- | --- |
| **81.** | 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");  }  }  Q1506 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |
| **82.** | 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");  }  }  Q1507 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |

|  |  |
| --- | --- |
| **83.** | 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");  }  }  Q1508 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |
| **84.** | 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");  }  }  Q1509 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |

|  |  |
| --- | --- |
| **85.** | 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");  }  }  Q1510 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |
| **86.** | 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");  }  }  Q1511 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |

|  |  |
| --- | --- |
| **87.** | 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");  }  }  Q1512 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |
| **88.** | 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");  }  }  Q1513 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |

|  |  |
| --- | --- |
| **89.** | 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");  }  }  Q1514 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |
| **90.** | 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");  }  }  Q1515 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |

|  |  |
| --- | --- |
| **91.** | 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");  }  }  Q1516 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |
| **92.** | 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");  }  }  Q1517 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |

|  |  |
| --- | --- |
| **93.** | 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");  }  }  Q1518 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |
| **94.** | 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");  }  }  Q1519 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |

|  |  |
| --- | --- |
| **95.** | 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");  }  }  Q1520 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |
| **96.** | 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");  }  }  Q1521 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |

|  |  |
| --- | --- |
| **97.** | 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");  }  }  Q1522 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |
| **98.** | 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");  }  }  Q1523 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |

|  |  |  |
| --- | --- | --- |
| **99.** | 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");  }  }  Q1524 | |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |
| **100.** | | 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");  }  }  Q1525 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | | |

|  |  |
| --- | --- |
| **101.** | 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?Q1526 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |
| **102.** | class A < T >  {  T obj;  }  class M49  {  static void test(A<? extends Number> arg)  {  arg.obj = 10;  }  }  // Will it compilation successfull or Not?Q1527 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |

|  |  |
| --- | --- |
| **103.** | class A < T >  {  void method(T obj)  {  }  }  class M50  {  static void test(A<? extends Number> arg)  {  arg.method(new Double(1.5));  }  }  Q1528 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |
| **104.** | 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");  }  }  Q1529 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |

|  |  |
| --- | --- |
| **105.** | 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");  }  }  Q1530 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |
| **106.** | 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");  }  }  Q1531 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |

|  |  |
| --- | --- |
| **107.** | 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");  }  }  Q1532 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |
| **108.** | class A < T >  {  T var;  }  class M55  {  public static void main(String[] args)  {  A<int> a1 = new A<int>();  System.out.println("done");  }  }  Q1533 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |

|  |  |
| --- | --- |
| **109.** | class A < T >  {  static T obj;  static void test(T arg)  {  }  }  class M56  {  public static void main(String[] args)  {  System.out.println("Hello World!");  }  }  Q1534 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |
| **110.** | interface A < T >  {  T obj;  }  class M57  {  public static void main(String[] args)  {  System.out.println("Hello World!");  }  }  Q1535 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |

|  |  |
| --- | --- |
| **111.** | 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");  }  }  Q1536 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |
| **112.** | class A < T >  {  T obj;  }  class M59  {  public static void main(String[] args)  {  A<Integer> a1 = new A<String>();  System.out.println("done");  }  }  Q1537 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |

|  |  |
| --- | --- |
| **113.** | 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");  }  }  Q1538 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |
| **114.** | 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");  }  }  Q1539 |
| |  | | --- | | A.  done |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error | | |

|  |  |
| --- | --- |
| **115.** | 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");  }  }  Q1540 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |
| **116.** | 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");  }  }  Q1541 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |

|  |  |
| --- | --- |
| **117.** | 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");  }  }  Q1542 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |
| **118.** | 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");  }  }  Q1543 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |

|  |  |
| --- | --- |
| **119.** | 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");  }  }  Q1544 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done | | |
| **120.** | 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);  }  }  Q1545 |
| |  | | --- | | A.  done with abc |  |  | | --- | | B.  Compile Time Error |  |  | | --- | | C.  Run time Error |  |  | | --- | | D.  done with 100 | | |

|  |  |
| --- | --- |
| **121.** | 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);  }  }  Q1546 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  done with abc  done with 500 |  |  | | --- | | D.  done with 500  done with abc | | |
| **122.** | 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);  }  }  Q1547 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  [90, 910, 190, 901]  2 |  |  | | --- | | D.  [90, 910, 190, 901]  190 | | |

|  |  |
| --- | --- |
| **123.** | 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);  }  }  Q1548 |
| |  | | --- | | 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 | | |
| **124.** | 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);  }  }  Q1549 |
| |  | | --- | | 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 | | |

|  |  |
| --- | --- |
| **125.** | 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);  }  }  Q1550 |
| |  | | --- | | 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 | | |
| **126.** | 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);  }  }  Q1551 |
| |  | | --- | | 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 | | |

|  |  |
| --- | --- |
| **127.** | 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);  }  }  Q1552 |
| |  | | --- | | A.  Compilation Error |  |  | | --- | | B.  Run Time Error |  |  | | --- | | C.  {abc=12, xyz=42, hello=22}  12 |  |  | | --- | | D.  {abc=12, xyz=42, hello=22}  abc | | |
| **128.** | 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?Q1553 |
| |  | | --- | | A.  Yes |  |  | | --- | | B.  No | | |

|  |  |
| --- | --- |
| **129.** | Lambda expression was introduced in which version of jdk?Q2857 |
| |  | | --- | | A.  jdk 1.5 |  |  | | --- | | B.  jk 1.8 |  |  | | --- | | C.  jdk 8.0 |  |  | | --- | | D.  none of the above | | |
| **130.** | lambda expression was used only for what?Q2858 |
| |  | | --- | | A.  sinle metod interface |  |  | | --- | | B.  multi level interface |  |  | | --- | | C.  jdk 8.0 |  |  | | --- | | D.  none of the above | | |

|  |  |
| --- | --- |
| **131.** | single method interface is also known as what?Q2859 |
| |  | | --- | | A.  method interface |  |  | | --- | | B.  attribute interface |  |  | | --- | | C.  functional intreface |  |  | | --- | | D.  none of the above | | |