Quiz-2: Core Java

1. Are there any compile errors in the following code.
2. Prior to JDK 1.5

ArrayList dates = new ArrayList();

dates.add(new Date());

dates.add(new String());

1. Yes B. No
2. Are there any compile errors in the following code.

ArrayList<Date> dates = new ArrayList<>();

dates.add(new Date());

dates.add(new String());

1. Yes B. No
2. What is wrong in the following code?

ArrayList dates = new ArrayList();

dates.add(new Date());

Date date = dates.get(0);

1. No wrong. B. Casting is needed C. No Casting is needed D. None of the Above.
2. What is wrong in the following code?

ArrayList<Date> dates = new ArrayList<>();

dates.add(new Date());

Date date = dates.get(0);

1. No wrong. B. Casting is needed C. No Casting is needed D. None of the Above.
2. What are the benefits of using generic types?
3. One important benefit is improving reliability and robustness.
4. Potential errors can be detected by the compiler.
5. Both A and B
6. None of the above.
7. What is the generic definition for java.lang.Comparable in the Java API?
8. package java.lang;

public interface Comparable<E> {

public int compareTo(E o) { }

}

1. package java.util;

public interface Comparable<E> {

public String compareTo(E o) { }

}

1. Since you create an instance of ArrayList of strings using new ArrayList<String>(), should the constructor in the ArrayList class be defined as

public ArrayList<E>()

1. No. B. Yes
2. Can a generic class have multiple generic parameters?
3. Yes. B. No.
4. Given int[] list = {1, 2, -1}, can you invoke sort(list) using the sort method in Listing 19.4?
5. No, because list is of type int[], but the sort method requires E[], where E is an object type.
6. Yes,
7. Given int[] list = {new Integer(1), new Integer(2), new Integer(-1)}, can you invoke sort(list) using the sort method in Listing 19.4?
8. No, because list is still of type int[], but the sort method requires E[], where E is an object type.
9. Yes.
10. What is a raw type?
11. When you use generic type without specifying an actual parameter, it is called a raw type.
12. When you use generic type specifying an actual parameter, it is called a raw type.
13. None the above.
14. Both A and B
15. Why is a raw type unsafe?
16. A raw type is unsafe, because some errors cannot be detected by the compiler.
17. A raw type is not unsafe.
18. None of the above
19. Both A and B
20. Why is the raw type allowed in Java?
21. The raw type is not allowed in Java for backward compatibility.
22. The raw type is allowed in Java for backward compatibility.
23. The raw type is allowed in Java for removing compile error.
24. What is the syntax to declare an ArrayList reference variable using the raw type and assign a raw type ArrayList object to it?
25. ArrayList list = new ArrayList();
26. HashMap list=new List();
27. List list=new List();
28. Set list=new HashSet();
29. Is GenericStack the same as GenericStack<Object>?
30. No, GenericStack is roughly equivalent to GenericStack<Object>, but they are not the same. GenericStack<Object> is a generic instantiation, but GenericStack is a raw type.
31. Yes, it is same
32. None of the above.
33. What are an unbounded wildcard, a bounded wildcard, and a lower-bound wildcard?
34. ? is unbounded wildcard ,? super T is lower bounded wildcard ? extends T is bounded wildcard
35. ? is unbounded wildcard , ? extends T is bounded wildcard, ? super T is lower bounded wildcard
36. ? extends T is bounded wildcard, ? super T is lower bounded wildcard, ? is unbounded wildcard ,
37. If your program uses ArrayList<String> and ArrayList<Date> , does the JVM load both of them?
38. No. Only ArrayList is loaded.
39. Yes, Both will load
40. Can you create an instance using new E() for a generic type E? Why?
41. No, because the type information is not available at runtime.
42. Yes

19. Why are generics used?  
a) Generics make code more fast  
b) Generics make code more optimised and readable  
c) Generics add stability to your code by making more of your bugs detectable at compile time  
d) Generics add stability to your code by making more of your bugs detectable at run time  
  
20. Which of these type parameters is used for a generic class to return and accept any type of object?  
a) K  
b) N  
c) T  
d) V

21. Which of these type parameters is used for a generic class to return and accept a number?  
a) K  
b) N  
c) T  
d) V  
  
22. Which of these is an correct way of defining generic class?  
a) class name(T1, T2, …, Tn) { /\* … \*/ }  
b) class name { /\* … \*/ }  
c) class name[T1, T2, …, Tn] { /\* … \*/ }  
d) class name{T1, T2, …, Tn} { /\* … \*/ }

23. Which of the following is incorrect statement regarding the use of generics and parameterized types in Java?  
a) Generics provide type safety by shifting more type checking responsibilities to the compiler  
b) Generics and parameterized types eliminate the need for down casts when using Java Collections  
c) When designing your own collections class (say, a linked list), generics and parameterized types allow you to achieve type safety with just a single class definition as opposed to defining multiple classes  
d) All of the mentioned

24. Which of the following reference types cannot be generic?  
a) Anonymous inner class  
b) Interface  
c) Inner class  
d) All of the mentioned

Quiz-4, Core Java

1. Which of the following class that you enable to create and control thread?

a) java.io.thread

b) java.lang.thread

c) java.util.\*

d) java.lang.system

2. How many main parts of thread or execution context?

a) 4 b)5

c)3 d)2

3. Which of the following main parts of thread?

a)A virtual CPU

b) the data on which the code works

c)the code that the CPU execute

d) above all

4. Two thread shared the same data when they share access to a common\_\_\_\_\_\_\_.

a) class b) method

c) object d) interface

5. A thread constructor takes an argument that is an instance of\_\_\_\_\_.

a) Running b) New

c) Dead d) Runnable

6. To create a newly thread you must call which method.

a) close() b) start ()

c) sleep() d) wait ()

7. The model of preemptive scheduler is that many threads might be runnable but how many thread is running?

a) two b) three

c) one d) four

8. When a thread complete execution and terminates, it can’t run again?

a) True

b) False

9. Which method is to used to determine if a thread is still visible?

a)alive b) isAlive c) runnable d) dead

10. The sleep method is one way to\_\_\_a thread for a period of time.

a) moving b)halt

c) running d) none

11. Join methods also depands on

a) operating system timers

b) schedulers

c) a+b d) none

12. Join also responds to nan interrupt an exit with an

a)i/oException b)ArithematicException

c)NullPointerException

d)InterruptedException

13. Which method we use to give other runnable threads a chance to execute?

a) Thread.yield() b) Thread.wait()

c) Thread.sleep() d) none

14. A mechanism that enables a programmer to control thread that are sharing data is called

a) thread b) synchronize

c) wait d) deadlock

15. Which of the following serial of lifecycle method of a thread?

a) Runnable –New—Dead—Running--Nonrunnable

b) New—Runnable—Running—Nonrunable—Dead

c) Running—Dead—Nonrunnable—New--Runnable

d) New—Running—Runnable—Nonrunnable—Dead

16. If two Thread instance of same class the can share same code when they execute.

a. True

b. False

17. An instance of Runnable is made from a\_\_\_\_\_\_\_\_\_\_\_

a. Thread Object.

b. Thread mathod.

c. Object.

d. Class.

18. Multithreaded programming environment enables you to creat multiple thread based on the\_\_\_\_\_\_\_\_\_\_

a. Different Runnable instance.

b. Same Runnable instance.

c. Two Runnable instance.

d. Three Runnable instance.

19. Which method run newly created Thread autometically?

a. begin();

b. stop();

c. trim();

d. start();

20. Preemptive and time-sliced are similer?

a. True

b. False

21. How many different states Thread object lifetime?

a. Two

b. Three

c. Four

d. Five

22. By which method can push Thread for preiod of time?

a. Thread.sleep();

b. Thread.start();

c. Thread.start-sleep();

d. Thread.sleepthread();

23. How many Thread Priority in java ?

a. One

b. Two

c. Three

d. Four

24. What is the default prioruty in java Thread ?

a. Thread.MIN\_PRIORITY

b. Thread.NORM\_PRIORITY

c. Thread.MAX\_PRIORITY

25. What dose Thread.yield() method do ?

a. stop Thread

b. start Thread

c. gives other runnable thread a chance to excute.

d. gives same runnable thread a chance to excute.

26. Which class is enables to create and control threads?

a. Java.swing.thread

b. Java.awt.thread

c. Java.lang.thread

d. Javax.swing.thread

27. Which one is true?

a. 2 threads can share the same data when they share access to a common object

b. 2 threads can share the same data when they share access to a different object

c. 2 threads can share the same data when they execute code from instance of the different class

28. Which one is true?

a. A newly created thread can be run automatically

b. A newly created thread cannot be run automatically

c. A newly created thread may be run automatically

29. Generally In java technology threads are \_\_\_\_\_\_\_?

a. Primitive

b. Boolean

c. Preemptive

d. Characteristics

30. The word preemptive means ---

a. Previously it was empty

b. Not primitive

c. Time-slicing

d. None of these

31. Which method is used to pausing a thread for some time?

a. Thread.pause ()

b. Thread.stop ()

c. Thread.sleep ()

32. Is it possible to make some actions at a time on a machine with one CPU by using thread?

a. Yes

b. No

33. The sleep is a \_\_\_\_ method in the thread class.

a. Dynamic

b. Static

c. Different

d. None of these

34. The word in thread “isAlive” means the thread is still \_\_\_\_\_\_\_?

a. Running

b. Alive

c. Not destroy

d. Viable

35. The term “isAlive” means is details \_\_\_\_\_\_\_?

a. The thread has been started and its task has been finished

b. The thread has been started but its task has not been completed

c. The thread has been started and already completed its job

d. The thread has been started and still it continues

36. In thread class “getPriority” method is a \_\_\_\_\_ type value.

a. Floating

b. Double

c. Int

d. Point

37. In thread Priority method default priority is -------

a. DEF\_PRIORITY

b. SET\_DEF\_PRIORITY

c. NORM\_PRIORITY

d. MIN\_PRIORITY

e. MAX\_PRIORITY

38. Which methods are responds to an interrupted method?

a. Sleep

b. InterruptedException

c. Join

d. None of the above

e. A & C

f. B & C

g. A & B

39. Why we use thread.yield () method---

a. To stop other runnable threads

b. To give other runnable threads a chance to execute

c. To pause other runnable threads and a chance to restart

d. All are false

40. Which keyword we used to stop corrupting data when more than single thread is running ---

a. Sleep

b. Break

c. Synchronized

d. Nothing of these

41. In java technology is there any “flag” option when creating object?

a. Yes

b. No

42. How many methods provide the “java.lang.Object” class?

a. 2

b. 3

c. 4

d. 1

43. Which are the methods of “java.lang.Object” class?

a. Wait

b. Notify

c. Break

d. A & C

e. A & B

f. B & C

Quiz-5, Core Java

1. Why is multithreading needed?

A. Multithreading can make your program more responsive and interactive, and enhance the performance.

B. Multithreading is needed in many situations, such as animation and client/server computing.

C. Both A and B

D. None of the above.

2. What is a runnable object?

A. An instance of Runnable is a runnable object.

B. An instance of Running state is a runnable object.

C. An instance of sleep state is a runnable object.

3. What is a thread?

A. A thread is wrapper object for a runnable object for executing a runnable task.

B. A thread is a runnable task.

C. A thread is an Interface

D. None of the above.

4. If a loop contains a method that throws an InterruptedException, why should the loop be placed inside a try-catch block?

A. It will produce compile error

B. If the loop is outside the try-catch block, the thread may continue to execute even though it is being interrupted.

C. None of the above.

5. How do you set a priority for a thread?

A. You use the putPriority() method to set the priority for a thread

B. You use the addPriority() method to set the priority for a thread

C. You use the setPriority() method to set the priority for a thread

D. You use the getPriority() method to set the priority for a thread

6.What is the default priority?

A. The default priority of the thread is Thread.NORM\_PRIORITY (1).

B. The default priority of the thread is Thread.NORM\_PRIORITY (5).

C. The default priority of the thread is Thread.NORM\_PRIORITY (10).

D. The default priority of the thread is Thread.NORM\_PRIORITY (0).

7. Is an instance of FlashText a runnable object?

A. Yes. Because it implements the Runnable interface.

B. No. Because it does not implement the Runnable interface.

8. What is the purpose of using Platform.runLater?

A. Invoking Platform.runLater(Runnable r) tells the system to run a task in the JavaFX application thread.

B. Invoking Platform.runLater(Runnable r) does not tell the system to run a task in the JavaFX application thread.

9. Can the wait(), notify(), and notifyAll() be invoked from any object?

A. Yes.

B. No

10. Can the read and write methods in the Buffer class be executed concurrently?

A. Yes.

B. No

11. When invoking the read method, what happens if the queue is empty?

A. It will wait for a signal for the queue to be not empty.

B. It will sleep.

C. It will be dead.