



## PROGRAMMING IN JAVA

### Assignment6

TYPE OF QUESTION: MCQ

Number of questions: 10

Total mark:  $10 \times 1 = 10$

---

#### **QUESTION 1:**

Which of the following is NOT a method of the Thread class in Java?

- a. `isInterrupted()`
- b. `interrupt()`
- c. `joins()`
- d. `sleep()`

**Correct Answer: c**

**Detailed Solution:**

`join()` is a method in the pre-defined Java class Thread but not `joins()`. Other methods like `isInterrupted()`, `interrupt()` and `sleep()` are defined in the Thread class.

---

#### **QUESTION 2:**

Which of the following statement(s) is/are true?

- a. `public int getId() :` returns the id of the thread.
- b. `public boolean isAlive() :` tests if the thread is alive.
- c. `public void interrupt() :` interrupts the thread.
- d. `public boolean isInterrupted() :` tests if the thread has been interrupted.

**Correct Answer: a,b,c,d**

**Detailed Solution:**

All options are correct.

---



**QUESTION 3:**

**Which of the following can be used to create an instance of Thread?**

- a. By implementing the Runnable interface.
- b. By extending the Thread class.
- c. By creating a new class named Thread and calling method run ().
- d. By importing the Thread class from package.

**Correct Answer: a, b**

**Detailed Solution:**

An application that creates an instance of Thread must provide the code that will run in that thread. There are two ways to do this:

- *Provide a Runnable object.* The [Runnable](#) interface defines a single method, run, meant to contain the code executed in the thread. The Runnable object is passed to the Thread constructor
- *Subclass Thread.* The Thread class itself implements Runnable, though its run method does nothing. An application can subclass Thread, providing its own implementation of run

**Reference:**<https://docs.oracle.com/javase/tutorial/essential/concurrency/runthread.html>

---



**QUESTION 4:**

**What is the output of the following program?**

```
public class Question
{
    public static void main(String[] args) {
        try {
            int a=5/0;
        } catch (Exception e) {
            catch (ArithmeticException a) {
            }
        }
        System.out.println("Hello World");
    }
}
```

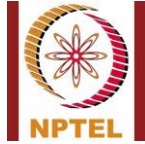
- a. Hello World
- b. 5
- c. Compile time error
- d. ArithmeticException

**Correct Answer: c**

**Detailed Solution:**

This first handler catches exceptions of type Exception; therefore, it catches any exception, including ArithmeticException. The second handler could never be reached. This code will not compile.

---



**QUESTION 5:**

Which one of these keywords must be used to handle the exception thrown by try block in some rational manner?

- a. try
- b. finally
- c. throw
- d. catch

**Correct Answer: d**

**Detailed Solution:**

The catch block is responsible for handling the exceptions raised by try block.

---

**QUESTION 6:**

Assume the following method is properly synchronized and called from a thread A on an object B: wait(2000); After calling this method, when will the thread A become a candidate to get another turn at the CPU?

- a) After thread A is notified, or after two seconds.
- b) Two seconds after thread A is notified.
- c) After the lock on B is released, or after two seconds.
- d) Two seconds after lock B is released.

**Answer : a**

**Explanation:** Either of the two events (notification or wait time expiration) will make the thread become a candidate for running again

---



**QUESTION 7:**

The following is a simple program using the concept of thread.

```
public class Question extends Thread{  
    public void run(){  
        for(int i=1;i<5;i++){  
  
            System.out.println(i++);  
        }  
    }  
    public static void main(String args[]){  
        Question t1=new Question();  
        t1.run();  
  
    }  
}
```

**What is the output of the above program?**

- a. 1  
3
- b. 1  
2  
3  
4
- c. Runtime error
- d. 1  
2

**Correct Answer: a**

**Detailed Solution:** Run to see the output.

---



**QUESTION 8:**

For the program given below, what will be the output after its execution?

```
public class Main{  
    public static void main(String[]args){  
        Thread thread=Thread.currentThread();  
        System.out.println(thread.activeCount());  
    }  
}
```

- a. 0
- b. true
- c. 1
- d. false

**Correct Answer: c**

**Detailed Solution:**

**java.lang.Thread.activeCount()** : Returns an estimate of the number of active threads in the current thread's thread group and its subgroups.

---

**QUESTION 9:**

Which of the following is a correct constructor for a thread object?

- a. Thread(Runnable a, String str);
- b. Thread(Runnable a, int priority);
- c. Thread(Runnable a, ThreadGroup t);
- d. Thread(int priority);

**Correct Answer: a**

**Detailed Solution:**

Thread(Runnable a, String str) creates a new Thread object. The others are not valid constructors to create a thread object.

---



**QUESTION 10:**

**Which of these keyword(s) is used to manually throw an exception?**

- a. try
- b. finally
- c. throw
- d. catch

**Correct Answer: c**

**Detailed Solution:**

The throw keyword is used to manually throw an exception.

---

\*\*\*\*\*END\*\*\*\*\*