

1. Which of the following are true about Error and Exception classes?

- A. Both classes extend Throwable.
- B. The Error class is final and the Exception class is not.
- C. The Exception class is final and the Error class is not.
- D. Both classes implement Throwable.

2. Given the code snippet below,

```
try{  
    String obj=(String)new Object();  
}
```

Which of these could be used in an appropriate catch block?

- A. NumberFormatException
- B. NullPointerException
- C. IllegalArgumentException
- D. ClassCastException

3. Analyze the code snippet below:

```
class One{  
    public void showMessage()throws IOException{ }  
}  
class Two extends One{  
    @Override  
    public void showMessage() throws _____{  
    }  
}
```

Fill in the blank, to make the above program compile successfully

- A. Exception
- B. Throwable
- C. RuntimeException
- D. FileNotFoundException

4. Analyze the code snippet below:

```
public class Example {  
    public static void main(String[] args) {  
        try{  
            int a=10/0;  
            System.out.println("a="+a);  
        }  
    }  
}
```

```

        }catch(Exception e){
            System.out.println("Exception -1");
        }catch(ArithmeticException e){
            System.out.println("ArithmeticException");
        }
    }
}

```

What is the output?

- A. Exception – 1
- B. ArithmeticException
- C. A= 10
- D. Compilation Error

5. Analyze the code snippet below:

```

public class Example {
    public static void main(String[] args) {
        try{
            int a=10/0;
            System.out.println("a="+a);

        }catch(NumberFormatException e){
            System.out.println("Exception -1");
        }finally{
            System.out.println("Finally ");
        }
    }
}

```

What is the output?

- A. Exception – 1 and Finally
- B. 10 and Finally
- C. Exception message and Finally
- D. Compilation Error

6. Analyze the code snippet below:

```

package com.core.ct;
public class Example {
    public static void main(String[] args) {
        try{

```

```

        int a=10/0;
        System.out.println("a =" +a);
    }catch(NumberFormatException e){
        System.out.println("Exception -1");
    }catch (ArithmeticException e) {
        System.out.println("ArithmeticException");
    }
    finally{
        System.out.println("Finally ");
    }
}
}

```

What is the output?

- A. Exception – 1 and Finally
- B. 10 Exception -1 Finally
- C. ArithmeticException and Finally
- D. Compilation Error

7. Analyze the code snippet below:

```

package com.core.ct;
public class Example {
    public static void main(String[] args) {
        try{
            String name="Java";
            System.out.println(name.substring(1,5));
        }catch(Exception e){
            System.out.println("Exception in :"+name);
        }
    }
}

```

What is the output?

- A. ava
- B. java
- C. Exception in : Java
- D. Compilation Error

8. Which of the following events would most likely throw an exception at compilation time?

- A. Dividing any number by 0
- B. Attempt to search an array beyond the index position
- C. Trying to open a file
- D. Attempt to add two integer objects by + operator

9. Analyze the code snippet below:

```
package com.core.ct;
public class Example {
    public static void main(String[] args) {
        System.out.println("Main method");
        try{
            doStuff();
        }catch(RuntimeException e){
            System.out.println("Exception - 1");
        }finally{
            System.out.println("Finally");
        }
        System.out.println("End of main");
    }
    public static int doStuff(){
        throw new ArithmeticException();
    }
}
```

What is the output?

- A. Main method
Exception - 1
- B. Main method
Exception – 1
End of main
- C. Main method
Exception – 1
Finally
End of main
- D. Compilation Error

10. Analyze the code snippet below:

```
package com.core.ct;
public class Example {
    public static void main(String[] args) {
        System.out.println("Main method");
        try{
            int a=10/0;
        }finally{
            System.out.println("Finally");
        }
        System.out.println("End of main");
    }
}
```

What is the output?

- A. Main method
Finally
- B. Main method
Finally
End of main
- C. Main method
Finally
Exception
- D. Compilation Error

11. Analyze the code snippet below:

```
package com.core.ct;
public class Example {
    public static void main(String[] args) {
        System.out.println("Main method");
        try{
            return;
        }finally{
            System.out.println("Finally");
        }
        System.out.println("End of main");
    }
}
```

What is the output?

- A. Main method
Finally
- B. Main method
Finally
End of main
- C. Main method
- D. Compilation Error

12. Exception and Error super class is _____ and its super class is _____

- A. RuntimeException , Object
- B. CheckedException, Object
- C. Throwable, Object
- D. Throwable, Exception

13. Any user-defined exception class is a subclass of the _____ class

- A. NumberFormatException
- B. NullPointerException
- C. Exception/RunTimeException
- D. None of the above

14. By using throws clause, how many exceptions can be thrown in method declaration?

- A. 1
- B. 2
- C. 3
- D. Any number of exceptions

15. Which are the keywords used in java exception handling

- A. try, catch, finally, thrown
- B. try, catch, finally, throws
- C. try, catch, finally, throws, throw
- D. none of the above

16. Which of the following are the valid code snippets to override the **show()** method in the **Example2** class.

```
class One extends Exception{
}
class Two extends One{
}
class Three extends Two{
}
class Example1{
    public void show()throws Three{    }
}
class Example2 extends Example1{
    //show method overriding
}
```

- A. public void show()throws IOException{ }
- B. public void show(){ }
- C. public void show() throws Two
- D. public void show() throws One
- E. public void show() throws Three

17. What is the output of the following code snippet?

```
package spaneos.demo;
class Example1 {
    public int getResult(int a, int b) {
        System.out.print("I ");    return a / b;
    }
}
class Example2 extends Example1 {
    public int getResult(int a, int b) {
        int res = 0;
        try {
            res = super.getResult(a, b);
            System.out.println(" am ");
        } catch (Exception e) {
            System.out.print(" am an ");
            throw e;
        }
        return res;
    }
}
public class Example {
    public static void main(String... args) {
        Example2 obj = new Example2();
    }
}
```

```

        try {
            int res = obj.getResult(10, 0);
            System.out.println("The result is " + res);
        } catch (Exception e) {
            System.out.print(" Exception");
        }
    }
}

```

- A. I am Exception
- B. I am an Exception
- C. I an am Exception
- D. am I Exception
- E. None of the above

18. Which of the following modification makes your program compilation and execution successful?

```

package spaneos.demo;
class InvalidAgeException extends Exception{
    //Code goes here
}
class Employee {
    private int age;
    public void setAge(int age) {
        if(age<=0)
            throw new InvalidAgeException();
        this.age=age;
    }
}

public class Example {
    public static void main(String... args){
        Employee obj=new Employee();
        try{
            obj.setAge(-10);
        }catch(InvalidAgeException e){
            System.out.println("Invalid age exception");
        }
    }
}

```

- A. Main method with throws InvalidAgeException
- B. setAge() with the throws InvalidAgeException
- C. Making InvalidAgeException as runtime exception
- D. Remove the throw class in setAge()
- E. None of the above

19. What is the output of the following code snippet?

```
public class Example {  
    public static void main(String... args) {  
        int a = 100, b = 0, c;  
        try {  
            System.out.print("You");  
            if (b == 0)  
                throw new Throwable("sorry!");  
            c = a / b;  
            System.out.println("Result is :" + c);  
        } catch (Exception e) {  
            System.out.println(" are into a simple problem " + e.getMessage());  
        } catch (Throwable e) {  
            System.out.println(" are into a big problem " + e.getMessage());  
        }  
    }  
}
```

- A. You are into a big problem sorry!
- B. You are into a simple problem sorry!
- C. Result is : 0 and exception message
- D. No output but it displays exception message
- E. You followed by exception message

20. What is the output of the following code snippet?

```
public class Example {  
    public static void main(String... args) {  
        int a = 100, b = 0, c;  
        try {  
            System.out.print("You");  
            if (b == 0)  
                throw new Throwable("Sorry!");  
            c = a / b;  
            System.out.println("Result is :" + c);  
        } catch (Exception e) {  
            System.out.println(" are into a simple problem " + e.getMessage());  
        } catch (Throwable e) {  
            System.out.println(" are into a big problem " + e.getMessage());  
        } finally {  
            System.out.print("Don't worry");  
        }  
        System.out.print(" We help you to solve your prob");  
    }  
}
```

- A. You are into a simple problem Sorry!
Don't worry We help you to solve your prob
- B. You are into a simple problem Sorry!
Don't worry
- C. You are into a simple problem Sorry!
We help you to solve your prob
- D. You are into a big problem Sorry!
Don't worry We help you to solve your prob
- E. You are into a simple problem Sorry!
Followed by exception message