

OOPS JAVA UNIT4 TEST-2

Total questions: 22

Worksheet time: 44mins

Instructor name: Mr. PRASHANT ATMAKURI

Name

Class

Date

1. What is the advantage of Exception Handling
 - a) None of these
 - b) To find out errors
 - c) To avoid abnormal termination of a program
 - d) To Debug program
2. What is the parent class of All Exceptions in JAVA
 - a) Error
 - b) Bug
 - c) Exception
 - d) Throw
3. Which of the following key word is optional in Exception handling program
 - a) throw
 - b) try
 - c) catch
 - d) finally
4. try block may or may not contains catch blocks
 - a) True
 - b) False
5. What is the purpose of 'throw' keyword
 - a) To Raise an exception implicitly
 - b) To Raise an exception explicitly
 - c) To create custom exceptions
 - d) To create user defined exceptions
6. What is an exception
 - a) Error occurred during the execution of a program
 - b) Bug occurred during the compile time of a program
 - c) It is related to input values
 - d) Simply an Error

7. How to create our own exception

- a) using 'finally' keyword
- b) using 'throwable' keyword
- c) using 'throw' keyword
- d) using 'throws' keyword

8. Exception classes belongs to following package

- a) import java.util.*
- b) import java.lang.Exception.*
- c) import java.lang.*
- d) import java.io.*

9. Which of these keywords is not a part of exception handling?

- a) finally
- b) thrown
- c) try
- d) catch

10. java.lang.NullPointerException is a

- a) runtime exception
- b) Compile time exception
- c) Error
- d) None

11. Which of these class is highest in hierarchy in java

- a) java.lang.Error
- b) java.lang.Exception
- c) java.lang.Object
- d) java.lang.Throwable

12. Choose the CORRECT exception

```
String department="JTMK";  
Integer no=Integer.parseInt(department);
```

- a) ArrayIndexOutOfBoundsException
- b) NullPointerException
- c) NumberFormatException
- d) ArithmeticException

13. Choose an exception if attempt to divide number by zero

```
int number = 89 / 0;  
System.out.println("The answer is " + number);
```

- | | |
|-------------------------|-----------------------------------|
| a) ArithmeticException | b) NumberFormatException |
| c) NullPointerException | d) ArrayIndexOutOfBoundsException |

14.

```
public class Foo  
{  
    public static void main(String[] args)  
    {  
        try  
        {  
            return;  
        }  
        finally  
        {  
            System.out.println( "Finally" );  
        }  
    }  
}
```

- | | |
|---------------------------------------|-----------------------|
| a) An exception is thrown at runtime. | b) Finally |
| c) The code runs with no output | d) Compilation fails. |

15. What will be the output of the program?

```
try
{
int x = 0;
int y = 5 / x;
}
catch (Exception e)
{
System.out.println("Exception");
}
catch (ArithmeticException ae)
{
System.out.println(" Arithmetic Exception");
}
System.out.println("finished");
```

a) runtime error.

b) Compilation fails

c) finished

d) Exception.

```
16. public class X
    {
    public static void main(String [] args)
    {
    try
    {
    badMethod();
    System.out.print("A");
    }
    catch (Exception ex)
    {
    System.out.print("B");
    }
    finally
    {
    System.out.print("C");
    }
    System.out.print("D");
    }
    public static void badMethod()
    {
    throw new Error(); /* Line 22 */
    }
    }
```

- | | |
|---|-----------------------|
| a) C is printed before exiting with an error message. | b) Compilation fails. |
| c) BC is printed before exiting with an error message | d) ABCD |

```

17. public class X
    {
    public static void main(String [] args)
    {
    try
    {
    badMethod();
    System.out.print("A");
    }
    catch (RuntimeException ex) /* Line 10 */
    {
    System.out.print("B");
    }
    catch (Exception ex1)
    {
    System.out.print("C");
    }
    finally
    {
    System.out.print("D");
    }
    System.out.print("E");
    }
    public static void badMethod()
    {
    throw new RuntimeException();
    }
    }

```

a) BD

b) BDE

c) DE

d) BD

18. FileNotFoundException

a) Found in java.io package

b) Is a subclass/extends IOException

c) All

d) Is a Compile time exception

19. class Main {
 public static void main(String args[]) {
 try {
 throw 10;
 }
 catch(int e) {
 System.out.println("Got the Exception " + e);
 }
 }
}

- | | |
|-------------------------|------------------------|
| a) Got the Exception 10 | b) Got the Exception 0 |
| c) Compiler Error | d) none |

20. class exception_handling
{ public static void main(String args[])
{ try
{ int a, b;
 b = 0;
 a = 5 / b;
 System.out.print("A");
} catch(ArithmeticException e)
{ System.out.print("B");
}
finally
{ System.out.print("C");
}
}
}

- | | |
|-------|------|
| a) AC | b) A |
| c) BC | d) B |

21. class IT

```
{  
static public void main(String...args)  
{  
int k=Integer.parseInt(args[0]);  
System.out.println(k);  
}  
}
```

if args[0]="gec" then what type of exception is thrown by the above program.

- | | |
|--------------------------|-----------------------------------|
| a) NullPointerException | b) ArrayIndexOutOfBoundsException |
| c) NumberFormatException | d) InvalidStringException |

22. class IT

```
{  
static public void main(String...args)  
{  
String s=null;  
System.out.println("length is:"+s.length());  
}  
}
```

- | | |
|-----------------------------------|-----------------------------|
| a) It throws NullPointerException | b) It throws StringExeption |
| c) it throws IOException | d) output is 0 |

Answer Keys

1. c) To avoid abnormal termination of a program
2. d) Throw
3. d) finally
4. a) True
5. b) To Raise , To create an exception explicitly
6. a) To create a user defined exceptions of a program
7. c) using 'throw' keyword
8. c) import java.lang.*
9. b) thrown
10. a) runtime exception
11. c) java.lang.Object
12. c) NumberFormatException
13. a) ArithmeticException
14. b) Finally
15. b) Compilation fails
16. a) C is printed before exiting with an error message.
17. b) BDE
18. c) All
19. c) Compiler Error
20. c) BC
21. c) NumberFormatException
22. a) It throws NullPointerException

