**Unit 4 Question Bank**

**Fill in the blanks/ True or False/ MCQ**

**Q1**What will be the output of the below program?

class ParentClass extends Exception

{

};

class ChildClass extends ParentClass

{

};

public class Tester {

public static void main(String[] args) {

try {

throw new ChildClass();

} catch (ParentClass p) {

System.out.println("Caught ParentClass exception");

} catch (ChildClass e) {

System.out.println("Caught ChildClass exception");

}

}

}

**Options**

Caught Parent Class Exception

Caught Child Class Exception

Compilation error because the ChildClass is not throwable

Compilation error because the catch block of a parent class exception must appear after the catch block of a child class exception

**Q2**

class Tester {

public static void main(String[] args) {

try {

// Insert code here

} catch (RuntimeException e) {

}

}

}

Which of the following statements could be inserted at the comment (// Insert code here) to allow the above code to compile and run?

**Options**

1. throw new ArrayIndexOutOfBoundsException();
2. throw new Exception();
3. throw new NullPointerException();
4. throw new IOException();

**Q3**What will be the output of the below program?

public class Tester {

public static void display() {

System.out.print(" inside display()");

throw new RuntimeException();

}

public static void main(String[] args) {

try {

System.out.print("main");

display();

} catch (Exception e) {

System.out.print(" caught");

} finally {

System.out.print(" finally");

}

System.out.print(" end");

}

}

**Options**

Compilation error as 'throws' keyword is not used

main inside display() caught end

main inside display() caught finally end

Runtime exception

**Q4**

What will be the output of the below program?

public class Tester {

public static void main(String[] args) {

try {

method1();

System.out.print("A");

} catch (Exception e) {

System.out.print("B");

} catch (Throwable e) {

System.out.print("C");

} finally {

System.out.print("D");

}

System.out.println("E");

}

public static void method1() {

throw new Error();

}

}

Options

ABDE

BDE

ABCDE

CDE

**Q5**

What will be the output of the below program?

interface A {

void eval(int x);

}

class B implements A {

public void eval(int x) {

if (x == 0)

throw new RuntimeException();

System.out.println("Evaluating...");

}

}

public class ExceptionDemo {

public static void main(String[] args) {

B obj = new B();

obj.eval(0);

}

}

**Options**

An exception of type RuntimeException will be thrown.

Compilation error because overridden method eval() cannot throw an exception

Compilation error because there is no throws declaration in eval() method in B

Compilation error because the abstract method in A doesn't declare exception to be thrown

Q 6 What will happen on executing the code given below?

interface BankOperations {

void withdraw();

void deposit();

}

abstract class Branch implements BankOperations {

public void withdraw() {

System.out.println("Amount withdrawn");

}

}

public class SubBranch extends Branch {

public void deposit() {

System.out.println("Amount deposited");

}

public static void main(String args[]) {

BankOperations subBranch = new SubBranch();

subBranch.deposit();

}

}

**Options**

Compilation error: Abstract class cannot implement an interface

Compilation error: class Branch should implement all the abstract methods

Compilation error: Cannot call deposit() method using BankOperation's reference

Code will get executed successfully and display "Amount deposited"

Q 7 Which of the following statements is true about Java's finally block?

* The finally block is only executed if an exception is thrown in the try block
* The finally block is only executed if an exception is thrown in the catch block
* The finally block is only executed if an exception is not thrown in the try or catch block
* The finally block is executed regardless of whether an exception is thrown in the try or catch block

Q 8 Which keyword raises an exception in Java code?

* try
* throw
* break
* throws

Q 9  If a class inheriting an abstract class does not define all of its function then it will be known as?

a) Abstract  
b) A simple class  
c) Static class

Q.10 An abstract class in Java can be created using the Keyword\_\_\_

A) final

B) interface

C) abstract

D) static

###### Q. 11 Which of the following statements applies to methods of an interface in Java?

**A** An interface can only contain abstract methods.

**B** You can define a method in an interface

**C** Private and protected access modifiers can also be used to declare methods in an interface

**D** None of the above

Q.11 Which of these access specifiers can be used for an interface?

a) Public  
b) Protected  
c) private  
d) All of the mentioned

Q 12 A java interface can contain \_\_\_\_\_\_\_.

A. Abstract methods(unimplemented) and implemented methods both  
B. public Abstract methods  
C. public static Final Variables only  
D. public static Final Variables and abstract methods both

Q 13 Which of the following is a correct interface?

A. abstract interface A { abstract void print(); { }}  
B. abstract interface A { print(); }  
C. interface A { void print() { } }  
D. interface A { void print(); }

Q 14 Which of these can be used to fully abstract a class from its implementation?

A. Objects  
B. Packages  
C. Interfaces  
D. None of the Mentioned

Q 15 Predict the output of following Java program

|  |
| --- |
| 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

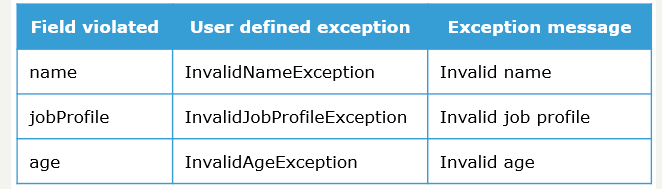
Q .16 Predict the output of the following program.

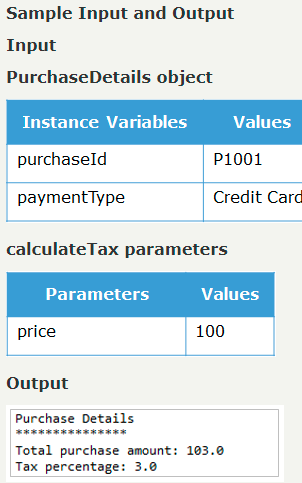
|  |
| --- |
| Class Test  {   int count = 0;      void A() throws Exception      {        try    {              count++;              try              {                  count++;                  try                  {                      count++;                      throw new Exception();                   }                  catch(Exception ex)                  {                      count++;                      throw new Exception();                  }              }              catch(Exception ex)              {                  count++;              }          }            catch(Exception ex)          {              count++;          }      }      void display()      {        System.out.println(count);      }      public static void main(String[] args) throws Exception      {          Test obj = new Test();          obj.A();          obj.display();      }  }  **Options** |

**(A)** 4  
**(B)** 5  
**(C)** 6  
**(D)** Compilation error

**Solve the following**

* 1. Write java program to throw user defined exception based on below descriptions.



* 1. Create a class telephone with lift ( ) and disconnected ( ) methods as abstract methods create another class smart telephone and demonstrate polymorphism .
  2. Write java program for e-commerce company by implementing an application for paying tax
  3. 

Q. 4 Write a java program to calculate area of rectangle using Interface and abstract class

Q.5 Create an abstract class 'Animals' with two abstract methods 'cats' and 'dogs'. Now create a class 'Cats' with a method 'cats' which prints "Cats meow" and a class 'Dogs' with a method 'dogs' which prints "Dogs bark", both inheriting the class 'Animals'. Now create an object for each of the subclasses and call their respective methods.

Q. 6 Write a java program using multiple catch blocks. Create a class CatchExercise inside the try block declare an array a[] and initialize with value a[5] =30/5; . In each catch block show Arithmetic exception and ArrayIndexOutOfBoundsException.

**Test Data:**

a[5] =30/5;

**Expected Output :**

ArrayIndexOutOfBoundsException occurs