**IIG Varsity**

E/43, Invocity Ave, iHub, IT Park, Sailashree Vihar Name of the Student

Patia, Bhubaneswar 751024

Test: **IIG-FND-004**, Time: **2 hours**, Date: **26-08-2022** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Section-1** (Answer any 3 questions) Mark: 45

Upload the program to your GitHub repository in your respective folder at

<https://github.com/milandas63/IIG-batch1>

* Write a program to convert the table below into an array and print the array in a tabular manner with rows and columns?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Mahatma | Gandhi | Father of the Nation | India | 1869 AD |
| Narendra | Modi | Prime Minister | India | 1950 AD |
| Nelson | Mandela | President | S. Africa | 1918 AD |
| Goutam | Buddha | Spiritual Teacher | India | 624 BC |

* Write a program to decode the following text:

Example: **Ksh mw Kssh - God is Good**

**Lmx sr ipizirxl - ?**

* Write a program to find the power of a number using recurssion?
* Write 2 programs to print next 5 numbers in the following series:
* 36, 34, 30, 28, 24, ...
* 22, 21, 23, 22, 24, 23, ...

**Section-2** (Answer all questions) Mark: 35

Colour the right answer in blue colour

* 1.Number of primitive data types in Java are?
* 6
* 7
* 8
* 9
* 2.What is the size of float and double in java?
* 32 and 64
* 32 and 32
* 64 and 64
* 64 and 32
* 3.Automatic type conversion is possible in which of the possible cases?
* Byte to int
* int to Long
* Long to int
* Short to int
* 4.Find the output of the following program.

public class Solution{

public static void main(String[] args){

short x = 10;

x = x \* 5;

System.out.print(x);

}

}

* 59
* 10
* Comiler error
* Exception
* 5.Find the output of the following program.

public class Solution{

public static void main(String[] args){

byte x = 127;

x++;

x++;

System.out.print(x);

}

}

* -127
* 127
* 129
* 2
* 6.Select the valid statement.
* char[] ch = new char(5);
* char[] ch = new char[5];
* char[] ch = new char();
* char[] ch = new char[];
* 7.Find the output of the following program.

public class Solution{

public static void main(String[] args){

int[] x = {120, 200, 016};

for(int i=0; i<x.length; i++){

System.out.print(x[i] + “ “);

}

}

}

* 120 200 016
* 120 200 14
* 120 200 16
* None
* 8.When an array is passed to a method, what does the method receive?
* A reference to the array
* A copy of the array
* Length of the array
* Copy of the first element
* 9.Select the valid statement to declare and initialize an array.
* int[] A = {};
* int[] A = {1,2,3};
* int[] A = (1,2,3);
* int[][] A = {1,2,3};
* 10.Find the value of A[1] after execution of the following program.

int[] A = {0,2,4,1,3};

for(int i = 0; i < a.length; i++){

a[i] = a[(a[i] + 3) % a.length];

}

* 0
* 1
* 2
* 3
* 11.Array in Java are:
* object reference
* objects
* Primitive data type
* None
* 12.When is the object created with new keyword?
* At run time
* At compile time
* Depends on the code
* None
* 13.Identify the corrected definition of a package.
* A package is a collection of editing tools
* A package is a collection of classes
* A package is a collection of classes and interfaces
* A package is a collection of interfaces
* 14.Identify the correct restriction on static methods.
* They must access only static data
* They can only call other static methods
* They cannot refer to this or super
* i & ii
* ii & iii
* Only iii
* i, ii & iii
* 15.Identify the keyword among the following that makes a variable belong to a class,

rather than being defined for each instance of the class.

* final
* static
* volatile
* abstract
* 16.Identify what can directly access and change the value of the variable res.

package com.mypackage;

public class Solution {

private int res = 100;

}

* Any class
* Only solution class
* Any class that extends Solution
* None
* 17.In which of the following is toString() method defined?
* java.lang.Object
* java.lang.String
* java.lang.util
* None
* 18.compareTo() returns
* True
* False
* An int value
* None
* 19.Where does the system stores parameters and local variables whenever a method is invoked?
* Heap
* Stack
* Array
* Tree
* 20.Identify the modifier which cannot be used for constructor.
* public
* protected
* private
* static
* 21.What is the variables declared in a class for the use of all methods of the class called?
* Object
* Instance variables
* Reference variables
* None
* 22.What is the implicit return type of constructor?
* No return type
* A class object in which it is defined
* void
* None
* 23.Identify the prototype of the default constructor.

public class Solution {}

* Solution(void) {}
* Solution() {}
* public Solution(void) {}
* public Solution() {}
* 24.Identify the correct way of declaring constructor.

public class Solution {}

* Solution() {}
* public Solution() {}
* Solution(void) {}
* Both (a) and (b)
* 25.Find the output of the following code.

public class Solution{

public static void main(String args[]){

int i;

for(i = 1; i < 6; i++){

if(i > 3) continue;

}

System.out.println(i);

}

}

* 3
* 4
* 5
* 6
* 26.How many times will “Interviewbit” be printed.

int count = 0;

do {

System.out.println(“Interviewbit”);

count++;

} while(count<10);

* 8
* 9
* 10
* 11
* 27.Identify the infinite loop.
* for( ; ; )
* for(int i=0; i<1; i--)
* for(int i=0; ; i++)
* All the above
* 28.When does Exceptions in Java arises in code sequence?
* Run Time
* Compilation Time
* Can Occur Any Time
* None of the mentioned
* 29. Which of these keywords is not a part of exception handling?
* try
* finally
* thrown
* catch
* 30. Which of these keywords must be used to monitor for exceptions?
* try
* finally
* throw
* catch
* 31.Which of these keywords must be used to handle the exception thrown by try block in some rational manner?
* try
* finally
* throw
* catch
* 32.Which of these keywords is used to manually throw an exception?
* try
* finally
* throw
* catch
* 33.Which of these is a super class of all exceptional type classes?
* String
* RuntimeExceptions
* Throwable
* Cachable
* 34.Which of these classes is related to all the exceptions that can be caught by using catch?
* Error
* Exception
* RuntimeExecption
* All of the mentioned
* 35.Which of these handles the exception when no catch is used?
* Default handler
* finally
* throw handler
* Java run time system
* 36.Which of these keywords is used to manually throw an exception?
* try
* finally
* throw
* catch
* 37.Which of these keywords is used to generate an exception explicitly?
* try
* finally
* throw
* catch
* 38.Which of these classes is related to all the exceptions that are explicitly thrown?
* Error
* Exception
* Throwable
* Throw
* 39.Which of these operator is used to generate an instance of an exception than can be thrown by using throw?
* new
* malloc
* alloc
* throw
* 40.Which of these keywords is used to by the calling function to guard against the exception that is thrown by called function?
* try
* throw
* throws
* catch

**Section-3** (Answer all questions) Mark: 20

Colour the right answer in blue colour

* 1.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);

}

}

}

* Got the Exception 10
* Got the Exception 0
* Compiler Error
* None of the above
* 2.Predict the output of following Java program

class Test extends Exception {}

class Main {

public static void main(String args[]) {

try {

throw new Test();

} catch(Test t) {

System.out.println("Got the Test Exception");

} finally {

System.out.println("Inside finally block ");

}

}

}

* Got the Test Exception

Inside finally block

* Got the Test Exception
* Inside finally block
* Compiler Error
* 3.What is the output of following Java program?

class Main {

public static void main(String args[]) {

int x = 0;

int y = 10;

int z = y/x;

}

}

* Compiler Error
* Compiles and runs fine
* Compiles fine but throws ArithmeticException exception
* None of the above
* 4.Observe the code and determine the action

class Base extends Exception {}

class Derived extends Base {}

public class Main {

public static void main(String args[]) {

try {

throw new Derived();

} catch(Base b) {

System.out.println("Caught base class exception");

} catch(Derived d) {

System.out.println("Caught derived class exception");

}

}

}

* Caught base class exception
* Caught derived class exception
* Compiler Error because derived is not throwable
* Compiler Error because base class exception is caught before derived class
* 5.Observe the code and determine the action

class Test {

public static void main (String[] args) {

try {

int a = 0;

System.out.println ("a = " + a);

int b = 20 / a;

System.out.println ("b = " + b);

} catch(ArithmeticException e) {

System.out.println ("Divide by zero error");

} finally {

System.out.println ("inside the finally block");

}

}

}

* Compile error
* Divide by zero error
* a = 0

Divide by zero error

inside the finally block

* a = 0
* inside the finally block
* 6.Observe the code and determine the action

class Test {

public static void main(String[] args) {

try {

int a[]= {1, 2, 3, 4};

for (int i = 1; i <= 4; i++) {

System.out.println ("a[" + i + "]=" + a[i] + "n");

}

} catch (Exception e) {

System.out.println ("error = " + e);

} catch (ArrayIndexOutOfBoundsException e) {

System.out.println ("ArrayIndexOutOfBoundsException");

}

}

}

* Compiler error
* Run time error
* ArrayIndexOutOfBoundsException
* Error Code is printed
* Array is printed
* 7.Predict the output of the following program.

class Test {

String str = "a";

void A() {

try {

str +="b";

B();

} catch (Exception e) {

str += "c";

}

}

void B() throws Exception {

try {

str += "d";

C();

} catch(Exception e) {

throw new Exception();

} finally {

str += "e";

}

str += "f";

}

void C() throws Exception {

throw new Exception();

}

void display() {

System.out.println(str);

}

public static void main(String[] args) {

Test object = new Test();

object.A();

object.display();

}

}

* abdef
* abdec
* abdefc
* None of the above
* 8.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();

}

}

* 4
* 5
* 6
* Compilation error
* 9.Which of these is a super class of all errors and exceptions in the Java language?
* RuntimeExceptions
* Throwable
* Catchable
* None of the above
* 10.The built-in base class in Java, which is used to handle all exceptions is
* Raise
* Exception
* Error
* Throwable