**Question 1**

Give names of two sorting algorithms?

**Question 2**

What is priority queue?

**Question 3**

What is the basic difference between a stack and a queue?

**Question 4**

Consider the following code snippet.

int i = 10;

int n = i++%5;

1. What are the values of i and n after the code is executed?
2. What are the final values of i and n if instead of using the postfix increment operator (i++), you use the prefix version (++i))?

**Question 5**

What is the result of trying to compile and run this program.

public class Test{

public static void main(String[] args){

int[] a = {1};

Test t = new Test();

t.increment(a);

System.out.println(a[a.length - 1]);

}

void increment(int[] i){

i[i.length - 1]++;

}

}

**Question 6**

What is the result of attempting to compile and run this ?

public class Test {

public static void main(String[] args){

String s = "HelloWorld".substring(5,10);

System.out.println(s);

}

}

**Question 7**

What is the result of attempting to compile and run this ?

public class Test {

public static void main(String[] args){

byte a = 10;

Byte b = new Byte(a);

Byte c = new Byte(11);

System.out.println(b.compareTo(c));

}

}

**Question 8**

What is the fundamental property that distinguishes Sets from other types of Collections?

**Question 9**

Is there anything wrong with this exception handler as written? Will this code compile?

try {

} catch (Exception e) {

} catch (ArithmeticException a) {

}

**Question 10**

What would be the output of the following program:

**public** **class** Overloader{

**public** **static** **void** read(String s){

System.out.println("read(String)");

}

**public** **static** **void** read(Integer i){

System.out.println("read(Integer)");

}

**public static void** read(Object o){

System.out.println("read(Object)");

}

**public static void** main(String args[]){

Object s = **new** String("Java");

Integer i = new Integer(10);

Overloader.read(s);

Overloader.read(i);

}

}

**Question 11**

What is the result of attempting to compile and run the following code?

public class Test {

public static void main(String[] args){

Integer a = new Integer(4);

Integer b = new Integer(8);

Integer c = new Integer(4);

HashSet hs = new HashSet();

hs.add(a);

hs.add(b);

hs.add(c);

System.out.println(hs);

}

}

**Question 12**

Comsider the following class hierarchy. What will be the outcome on attempting to compile and run this ?

interface A {

public void method1();

}

class One implements A{

public void method1(){

System.out.println("hello");

}

}

class Two extends One{}

public class Test extends Two{

public static void main(String[] args)

{

A a;

Two t = new Two();

a = t;

a.method1();

}

}

**Question 13**

What will this program print out ?

class Base{

int value = 0;

Base(){

addValue();

}

void addValue(){

value += 10;

}

int getValue(){

return value;

}

}

class Derived extends Base{

Derived(){

addValue();

}

void addValue(){

value += 20;

}

}

public class Test {

public static void main(String[] args){

Base b = new Derived();

System.out.println(b.getValue());

}

}

**Question 14**

What is the result of attempting to compile and run this ?

interface ITest{

public void setVal();

}

public class Test {

private String a;

void aMethod(){

final String b = " World";

ITest it = new ITest() {

public void setVal(){

a = "Hello" + b;

}};

it.setVal();

System.out.println(a);

}

public static void main(String[] args) {

Test t = new Test();

t.aMethod();

}

}

**Question 15**

What is the advantage of using a PreparedStatement?

**Question 16**

Explain about servletConfig and servletContext in applications?

**Question 17**

How does JSP handle runtime exceptions?

**Question 18**

**What is the difference between using getSession(true) and getSession(false) methods?**

**Question 19**

How many JSP scripting elements and what are they?

**Question 20**

How do I perform browser redirection from a JSP page?