

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

Question 1 : a) What is Object Oriented Paradigm? Explain features of Object Oriented Paradigm. Why Object Oriented Programming are preferred over structured programming ?

(5 Marks)

Answer 1 : a) Object oriented programming emerges. This approach to program organization and development attempts to eliminate some of the pitfalls of conventional programming by incorporating the best of the structured programming features with several powerful new concepts. This approach speeds the development of new programs, and, if properly used, improves the maintenance, reusability, and modifiability of software.

The problems arose due to the fundamental principle of this paradigm. The whole emphasis is on doing things. Functions do some activity, maybe a complex one, but the emphasis is still on doing. *Data are given a lower status.* For example in banking application, more emphasis is given to the function which collects the correct data in a desired format or the function which processes it by doing some summation, manipulation etc. or a function which displays it in the desired format or creates a report. But you will also agree that the important part is the data itself.

The world and its applications are not organized as functions and values separate from one another. The problem solvers do not think about the world in this manner. They always deal with their problems by concentrating on the **objects**, their **characteristics** and **behavior**.

Object Oriented approach

The major factor, which leads to the development of this new approach i.e, Object Oriented approach is to resolve many problems encountered earlier in the procedural approach.

In this approach, we decompose a problem into a number of entities called objects and then build data and functions around these entities. The notion of "Object" comes into the picture. *'A collection of data and its operations is referred to as an object'*. Data is a vital element in the program development. Data is local to an object. This is encapsulated within an object and is not accessible directly from outside the object. These objects know how to interact with another object through the interface (a set of operations).

The salient features of Object Oriented programming are:

- More emphasis is on data rather than procedure.
- Programs are modularized into entities called *objects*.
- Data structures methods characterize the objects of the problem.
- Since the data is not global, there is no question of any operations other than

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

those defined within the object, accessing the data. Therefore, there is no scope of accidental modification of data.

- It is easier to maintain programs. The manner in which an object implements its operations is internal to it. Therefore, any change within the object would not affect external objects. Therefore, systems built using objects are resilient to change.
- Object reusability, which can save many human hours of effort, is possible. An application developer can use objects like 'array', 'list', 'windows', 'menus', 'event' and many other components, which were developed by other programmers, in her program and thus reduce program development time.
- It employs bottom-up approach in program design.

b) Explain basic features of Java programming language.

(5 Marks)

Basic Features

Platform Independent

Java is Platform independent. The meaning of platform here may be confusing for you but actually this word is poorly defined. In the computer industry it typically means some combination of hardware and system software but here you can understand it as your operating system.

Object Oriented Java Language Basics

As you know that in objects represent object-oriented languages data. Objects have two sections. The first is **Data** (instance variables) and the second is **methods**. Data represents what an object is. A method represents what an object does. The Data and methods are closely related to the real world structure and behavior of objects. Object oriented programming has a number of advantages like. Simpler to read program, efficient reuse of programming segments, robust and error-free code

Easy to Learn

Java is easy to learn for programmers because it is (syntax) similar to C and C++ and most of the complex parts of C/C++ have been excluded including **operator overloading, multiple inheritance and pointers**. Approximately half of the bugs in C and C++ programs are related to memory allocation and de-allocation. Therefore the important addition in Java is automatic memory allocation and de-allocation. Do not think Java is very simple, it is both a simple as well as complex language depending on how you use it because Java has a wide range of applications, simple to complex.

Robust

Java provides checking for possible problems at two levels, one at the compile time

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

and the other at the run time, so programs are highly **reliable** and eliminate situations that are error-prone compared to C/C++. The best and worst features of C and C++ are **pointers** that help in direct manipulation of memory addresses. The power of pointers is as a great tool used by expert programmers for developing system software, driver, etc. But many times pointers are the main cause of **runtime errors** because of improper use of memory. Java eliminates pointer manipulation completely from the language, and therefore eliminates a large source of runtime errors. Java programmers need not remember to de-allocate memory in programs since there is a **garbage collection mechanism** which handles de-allocation of memory. It provides powerful a robust exception handling mechanism to deal with both expected and unexpected errors at run time.

Secure

Java is intended to work in networked and distributed environments by providing security. All the references to memory are **symbolic references**, meaning that the user is not aware where in the memory program is present, it totally depends on the JVM and machine on which the program is running. Each applet is loaded on its own memory space, which avoids the information interchange between applets.

Multi-threaded

Before answering what is multithreading, let me explain you what '**thread**' is. Simply, a thread is a program's path of execution. In your problems, when multiple events or actions need to occur at the same time, how you will handle it? For example, a program is not capable of drawing pictures when you keep pressing keys of the keyboard. The program gives its full attention to receiving the keyboard input and doesn't draw the picture properly.

Dynamic

Java was designed to adapt to an evolving environment, therefore the Java compiler is smart and dynamic. If you are compiling a file that depends on other non-compiled files, then the compiler will try to find and compile them also. The compiler can handle methods that are used before they're declared. It can also determine whether a source code has been changed since the last time it was compiled. In Java classes that were unknown to a program when it was compiled can still be loaded into it at runtime. For example, a web browser can load applets of other classes without recompilation.

High Performance

As we know in Java we have to first compile the program, then execute it using Java interpreter. In general, interpreters are slow, because an interpreter executes programs instruction by instruction while Java is a fast-interpreted language. Java has also been designed so that the run-time system can optimize their performance by compiling

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

bytecode to native machine code on the fly (execute immediately after compilation). This is called “**just in time**” (JIT) compilation.

Question 2 : a) What is static method? Explain why main method in Java is always static.

(2 Marks)

Answer 2 : a) Static methods

methods and variables that are not declared as static are known as instance methods and instance variables or in other words you can say that they belong to objects of the class.

A static method is a characteristic of a class, not of the objects it has created.

Static variables and methods are also known as class variables or class methods since each class variable and each class method occurs once per class Instance methods and variables occur once per instance of a class or you can say every object is having its own copy of instance variables.

One very important point to note here is that a program can execute a static method without creating an object of the class. All other methods must be invoked through an object, and, therefore an object must exist before they can be used. You have seen that every Java application program has one main() method. This method is always static because Java starts execution from this main() method, and at that time no object is created.

b) What are different bitwise operators available in Java? Write a Java program to explain the use of bitwise operators. **(5 Marks)**

Description of Boolean operators

Operator = &&

Use = A && B

Description = Conditional AND :If both A and B are true, result is true. If either A or B are false, the result is false. But if A is false, B will not be evaluated.

For example, (A > 4 && B <= 10) will evaluate to true if A is greater than 4, and B is less than or equal to 10.

Operator = &

Use = A & B

Description = Boolean AND: If both A and B are true, the result is true. If either A or B are false, the result is false and both A and B are evaluated before the test.

Bitwise operators

As you know in computers data is represented in binary (1's and 0's) form. The binary representation of the number 43 is 0101011. The first bit from the right to left in the binary representation is the least

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

significant bit, i.e. here value is 1. Each Bitwise operator allows you to manipulate integer variables at bit level.

c) What is constructor? Explain constructor overloading. Write a program in java to create Book class and define its constructor(s). (3 Marks)

Constructor

A Constructor initializes object with its creation. It has the same name as the name of its class. Once a constructor is defined, it is automatically called immediately the memory is allocated before the new operation completes. Constructor does not have any return type, its implicit return type is class object. You can see constructor as a class type. Its job is to initialise the instance variables of an object, so that the created object is usable just after creation.

Overloading Constructors

there may be more than one constructor in a class but all the constructors in a class either will have different types of arguments or different number of arguments passed to it. This is essential because without this it wouldn't be possible to identify which of the constructors is invoked. Having more than one constructor in a single class is known as constructor overloading. In the program given below more than one constructor in class Student are defined

```
example
class book
{
    Int book_id; String
    book_name;
    Book()
    {
        Book_id=12;
        Book_name="let us c";
    }
    Void Display()
    {
        System.out.println("book id="+book_id);
        System.out.println("book name="+book_name)
    }
}
Class demo
{
    Public static void main()
    {
```

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

```
Book b=new book();  
b.display();  
}  
}
```

Question 3 : a) What is overloading of methods? Explain with an example how overloading of methods is different from overriding of methods. **(5 Marks) Answer 3 :a) METHOD OVERLOADING**

Sometimes two or more methods which are very similar in nature are required.

All the methods of similar kinds in a class can have the same name, but remember all the methods will have different prototypes. This concept is known as method overloading. By matching the type and order of arguments passed to methods. The exact method that should execute is resolved. Sometimes overloaded methods may have different return types, the return type alone is not sufficient to differentiate two overloaded methods.

```
class Test  
{  
int Area( int i)  
{  
return i*i;  
}  
int Area(int a,int b)  
{  
return a*b;  
}  
}  
class Area_Overload  
{  
public static void main(String args[])  
{  
Test t = new Test();  
int area;  
area = t.Area(5);  
System.out.println("Area of Squire is : "+area);  
area = t.Area(5,4);  
System.out.println("Area of Rectangle is : "+area);  
}  
}
```

Output of this program
is: Area of Squire is: 25
Area of Rectangle is: 20

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

```
class Figure
{
double sidea;
double sideb;
Figure(double a, double b)
{
sidea = a;
sideb = b;
}
Figure(double a)
{
sidea = a;
sideb = a;
}
double area( )
{
System.out.println("Area inside figure is
Undefined."); return 0;
}
}
class Rectangle extends Figure
{
Rectangle( double a , double b)
{
super ( a, b);
}
double area ( )
{
System.out.println("The Area of Rectangle:");
return sidea*sideb;
}
}
class Squre extends Figure
{
Squre( double a )
{ 37 Inheritance and Polymorphism
```

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

```
super (a);
}
double area()
{
System.out.println("Area of Squire:
"); return sidea*sidea;
}
}
class Area_Overrid
{
public static void main(String[] args)
{
Figure f= new Figure(20.9, 67.9);
Rectangle r = new Rectangle( 34.2,
56.3); Squire s = new Squire( 23.1);
System.out.println("***** Welcome to Override Demo *****");
f.area();
System.out.println(" "+r.area());
System.out.println(" "+s.area());
}
}
```

Output:

```
***** Welcome to Override Demo *****
```

```
Area inside figure is Undefined.
```

```
The Area of Rectangle: 1925.46
```

b) What is abstract class? Explain advantages of abstract class with the help of an example. (5 Marks)

Abstract classes usually contain **abstract methods**. Abstract method is a method signature (declaration) without implementation. Basically these abstract methods provide a common interface to different derived classes. Abstract classes are generally used to provide common interface derived classes. You know a superclass is more general than its subclass(es). The superclass contains elements and properties common to all of the subclasses. Often, the superclass will be set up as an abstract class, which does not allow objects of its prototype to be created. In this case only objects of the subclass are created. To do this the reserved word abstract is included (prefixed) in the class definition.

For example, the class given below is an abstract class.

```
public abstract class Player // class is abstract
{
private String name; public
Player(String vname)
```

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

```
{
name=vname;
}
public String getName() // regular method
{
return (name);
}
public abstract void Play();
// abstract method: no implementation
}
```

Question 4 : a) What is polymorphism? Write a java program to show the advantage of polymorphism.
(5 Marks)

Polymorphism is the capability of a method to do different things based on the object through which it is invoked or object it is acting upon.

dynamic method binding. Java uses Dynamic Method Dispatch mechanism to decide at run time which overridden function will be invoked. Dynamic Method Dispatch mechanism is important because it is used to implement runtime polymorphism in Java. Java uses the principle: “a super class object can refer to a subclass object” to resolve calls to overridden methods at run time.

```
abstract class Player // class is abstract
{
private String name;
public Player(String nm)
{
name=nm;
}
public String getName() // regular method
{
return (name); 40
```

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

```
}  
public abstract void Play();  
// abstract method: no implementation  
}  
class Cricket_Player extends Player  
{  
    Cricket_Player( String var)  
    {  
    }  
    public void Play()  
    {  
        System.out.println("Play Cricket:"+getName());  
    }  
}  
class Hockey_Player extends Player  
{  
    Hockey_Player( String var)  
    {  
    }  
    public void Play()  
    {  
        System.out.println("Play Hockey:"+getName());  
    }  
}  
class Football_Player extends Player  
{  
    Football_Player( String var)  
    {  
    }  
    public void Play()  
    {  
        System.out.println("Play Football:"+getName());  
    }  
}  
public class PolyDemo  
{  
    public static void main(String[] args)  
    {  
        Player ref; // set up var for an Player!  
        Cricket_Player aCplayer = new Cricket_Player("Sachin"); // makes specific  
        objects Hockey_Player aHplayer = new Hockey_Player("Dhanaraj");  
        Football_Player aFplayer = new Football_Player("Bhutia");  
        // now reference each as an Animal  
        ref = aCplayer;
```

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

```
ref.Play();  
ref = aHplayer;  
ref.Play();  
ref = aFplayer;  
ref.Play();  
}  
}
```

Output:

```
Play Cricket:Sachin  
Play Hockey:Dhanaraj  
Play Football:Bhutia
```

b) What is package in Java ? Explain how to decide the need of package(s) in a system which is to be developed using Java. **(5 Marks)**

managing the program. The package statement is used to define space to store classes. In Java you can write your own package. Writing packages is just like you write any other Java program. You just have to take care of some points during writing packages, which are given below. A package is a collection of related classes and interfaces providing access protection and namespace management.

There must be not more than one public class per file. All files in the package must be named name_of_class.java where name_of_class is the name of the single public class in the file. The very first statement in each file in the package, before any import statements or anything put the statement package myPackageName;

Question 5: a) What is an exception? Explain how an exception is handled in Java. Also explain hierarchy of different exception classes in Java. **(5 Marks)**

An exceptional condition is considered as a problem, which stops program execution from continuation from the point of occurrence of it. Exception stops you from continuing because of lack of information to deal with the exception condition. In other words it is not known what to do in specific conditions. If the system does not provide it you would have to write your own routine to test for possible errors. You need to write a special code to catch exceptions before they cause an error.

HANDLING OF EXCEPTION

Exceptions in Java are handled by the use of these five keywords: **try, catch, throw, throws, and finally**. You have to put those statements of program on which you want to monitor for exceptions, in **try** block. If any exceptions occur that will be caught using **catch**. Java runtime system automatically throws system-generated exceptions. The **throw** keyword is used to throw exceptions manually.

Exception class and its Subclasses in Throwable class Hierarchy

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

```
class java.lang.Object
|
+ class java.lang.Throwable
|
+ class java.lang.Exception
|
+ class java.awt.AWTException
|
+ class java.lang.ClassNotFoundException
|
+ class java.lang.CloneNotSupportedException
|
+ class java.io.IOException
|
+ class java.lang.IllegalAccessException
|
+ class java.lang.InstantiationException
|
+ class java.lang.InterruptedException
|
+ class java.lang.NoSuchMethodException
|
+ class java.lang.NoSuchMethodException
|
+ class java.lang.RuntimeException
|
+ class java.lang.ArithmeticException
|
+ class java.lang.ArrayStoreException
|
+ class java.lang.ClassCastException
|
+ class java.util.EmptyStackException
|
+ class java.lang.IllegalArgumentException
|
+ class java.lang.Error
```

b) What is String class in java? Explain how it is different from String Buffer class. Also write a java program to find whether a given string is a palindrome or not. **(5 Marks)**

Java provides two classes for handling string values, which are **String**

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

The String class is provided for strings whose value will not change. For example, in program you write a method that requires string data and it is not going to modify the string.

The StringBuffer class is used for strings that will be modified. String buffers are generally used for constructing character data dynamically, for example, when you need to store information, which may change.

Content of strings do not change that is why they are more efficient to use than string buffers. So it is good to use String class wherever you need **fixed-length** objects that are **immutable** (size cannot be altered).

For example, the string contained by myString object given below cannot be changed. String myString = "This content cannot be changed!".

```
import java .io.*;
class palindrome
{
    public static void main(String args[])throws IOException
    {
        BufferedReader in=new BufferedReader(new
InputStreamReader(System.in));
        String s1,s2="";
        int i,l;
        char ch;
        System.out.println("ENTER STRING TO
CHECK"); s1=in.readLine();
        l=s1.length();
        for(i=l-1;i>=0;i--)
        {
            ch=s1.charAt(i);
            s2=ch+s2;
        }
        if(s1.equals(s2))
            System.out.println("PALINDROME");

        else
            System.out.println("NOT PALINDROME");
    }
}
```

Question 6 : a) What is multithreading ? Explain advantages of multithreaded programming with the help of a Java program. (5 Marks)

Multithreaded programs support more than one concurrent thread of execution. This means they are able to simultaneously execute multiple sequences of instructions. Each instruction sequence has its own unique flow of control that is independent of all others. These independently executed instruction sequences are known as threads.

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

Your PC has only a single CPU; you might ask how it can execute more than one thread at the same time? In single processor systems, only a single thread of execution occurs at a given instant. But multiple threads in a program increase the utilization of CPU.

Advantages of Multithreading

The advantages of multithreading are:

CPD

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

- i. Concurrency can be used within a process to implement multiple instances of simultaneous services.
- ii. Multithreading requires less processing overhead than multiprogramming because concurrent threads are able to share common resources more efficiently.

b) What is I/O stream in Java? Write a program in Java to create a file to store a Java program in it.

(5 Marks)

An **InputStream** represents a stream of data from which data can be read. Again, this stream will be either directly connected to a device or else to another stream.

An **OutputStream** represents a stream to which data can be written. Typically, this stream will either be directly connected to a device, such as a file or a network connection, or to another output stream.

```
import java.io.*;
class FileWriteDemo
{
public static void main(String[] args) throws IOException
{
// open keyboard for input
BufferedReader stdin = new BufferedReader(new InputStreamReader(System.in));
String s = "output.txt";
// check if output file
exists File f = new File(s);
if (f.exists())
{
System.out.print("Overwrite " + s + " (y/n)? ");
if(!stdin.readLine().toLowerCase().equals("y"))
return;
}
// open file for output
PrintWriter outFile = new PrintWriter(new BufferedWriter(new FileWriter(s)));
System.out.println("Enter some text on the keyboard...");
System.out.println("^z to terminate");
// read from keyboard, write to file output
stream String s2;
while ((s2 = stdin.readLine()) != null)
outFile.println(s2);
// close disk file
outFile.close();
}
}
```

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

Output:

Enter some text on the keyboard...

(^z to terminate)

hello students ! enjoying Java

Session ^Z

Open out.txt you will find

“hello students ! enjoying Java Session” is stored in it.

Question 7 : a) How a Java Applet is different from Java Application program? Create an Applet program to display current date and time. (4 Marks)

Applets differ from Java applications in the way that they are not allowed to access certain resources on the local computer, such as files and serial devices (modems, printers, etc.), and are prohibited from communicating with most other computers across a network. The common rule is that an Applet can only make an Internet connection to the computer from which the Applet was sent.

b) What is layout manager? Which is the default layout manager? Differentiate among the way of managing components by 5 different layouts. (6 Marks)

add a component to an applet or a container, the container uses its *layout manager* to decide where to put the component. Different *LayoutManager* classes use different rules to place components.

java.awt.LayoutManager is an interface. Five classes in the *java* packages implement it:

- *FlowLayout*
- *BorderLayout*
- *CardLayout*
- *GridLayout*
- *GridBagLayout*
- plus *javax.swing.BoxLayout*

Question 8 : a) What is a socket ? Explain how a network socket is created using Java. (5 Marks)

Stream Sockets

Stream Sockets are used to provide a connection-oriented service (i.e. TCP Transmission Control Protocol).

With stream sockets a process establishes a connection to another process. Once the connection is in place, data flows between processes in continuous streams.

Datagram Sockets

This socket are used to provide a connection-less service, which does not guarantee that packets reach the destination and they are in the order at the destination. In this, individual packets of information are transmitted. In fact it is observed that packets

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

can be lost, can be duplicated, and can even be out of sequence.

b) What is DNS? Explain components of DNS. (5 Marks)

Domain Name System (DNS) is one of the industry-standard suite of protocols that comprise TCP/IP. Microsoft Windows Server 2003. DNS is implemented using two software components: the DNS server and the DNS client (or resolver). Both components are run as background service applications.

Network resources are identified by numeric IP addresses, but these IP addresses are difficult for network users to remember. The DNS database contains records that map user-friendly alphanumeric names for network resources to the IP address used by those resources for communication. In this way, DNS acts as a mnemonic device, making network resources easier to remember for network users.

DNS COMPONENTS:

The DNS consists of three components. The first is a "Name Space" that establishes the syntactical rules for creating and structuring legal DNS names. The second is a "Globally Distributed Database" implemented on a network of "Name Servers". The third is "Resolver" software, which understands how to formulate a DNS query and is built into practically every Internet-capable application.

(A) Name Space:

The DNS "Name Space" is the familiar inverted tree hierarchy with a null node named "" at the top. The child nodes of the root node are the Top Level Domains (TLDs)-.com, .net, .org, .gov, .mil-and the country code TLDs, including .jp, .uk, .us, .ca, and so forth. Node names, known as labels, can be as many as 63 characters long, with upper- and lower-case alphabetical letters, numerals, and the hyphen symbol constituting the complete list of legal characters. Labels cannot begin with a hyphen. Upper- and lower-case letters are treated equivalently. A label can appear in multiple places within the name space, but no two nodes with the same label can have the same parent node: A node name must be unique among its siblings.

(B) Name Servers:

The second key component of the DNS is a globally connected network of "name servers". Each zone has a primary or master name server, which is the authoritative source for the zone's resource records. The primary name server is the only server that can be updated by means of

For 100% Result Oriented IGNOU Coaching and Project Training

Call CPD: 011-65164822, 08860352748

local administrative activity. Secondary or slave name servers hold replicated copies of the primary server's data in order to provide redundancy and reduce the primary server's workload.

Furthermore, name servers generally cache data they have looked up, which can greatly speed up subsequent queries for the same data. Name servers also have a built-in agent mechanism that knows where to ask for data it lacks. If a name server can't find a domain within its zone, it sends the query a step closer to the root, which will resend it yet a step closer if it can't find the domain itself. The process repeats until it reaches a TLD, which ensures that the entire depth of the name space will be queried if necessary.

The combination of all the DNS name servers and the architecture of the system creates a remarkable database. There are more than 32 million domain names in the popular TLDs for which the whois utility works. Nominum, whose chief scientist, Paul Mockapetris, invented DNS, claims that there are more than 100 million domain names stored and that the system can easily handle 24,000 queries per second. The database is distributed-no single computer contains all the data. Nevertheless, data is maintained locally even though it's distributed globally, and any device connected to the IP network can perform lookups. The update serial number mechanism in each zone ensures a form of loose coherency on the network-if a record is out of date, the querier knows to check a more authoritative name server.

(C) Resolver:

The third component of the DNS is the "resolver". The resolver is a piece of software that's implemented in the IP stack of every destination point, or "host" in IETF-speak. When a host is configured, manually or through DHCP, it's assigned at least one default name server along with its IP address and subnet mask. This name server is the first place that the host looks in order to resolve a domain name into an IP address. If the domain name is in the local zone, the default name server can handle the request. Otherwise, the default name server queries one of the root servers. The root server responds with a list of name servers that contain data for the TLD of the query. This response is known as a referral. The name server now queries the TLD name server and receives a list of name servers for the second-level domain name. The process repeats until the local name server receives the address for the domain name. The local server then caches the record and returns the address or other DNS data to the original querier