PRACTICAL NO 1

Create a program to send encrypted messages from sender and decrypt a message at receiver end

Create a sender.py

```
import socket

import random


def main():

    # Connect to the server

    with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as sc:

        sc.connect(('localhost', 6017))


        # Get input from the user

        s = input("Enter the string: ")


        # Initialize variables

        key = []

        ct = ''


        # Encrypt the message

        for char in s:

            j = random.randint(0, 49)  # Random integer between 0 and 49

            key.append(str(j))          # Append to key list

            ct += chr(ord(char) + j)   # Encrypt character


            print(f"j={j}")             # Print the random number used for each character


        # Convert key list to a comma-separated string

        key_str = ','.join(key)


        # Print the key and encrypted message
```

```
        print(f"Key={key_str}")
        print(f"Encrypted message: {ct}")

        # Send the encrypted message and key
        message = f"{ct},{key_str}"
        sc.sendall(message.encode('utf-8'))  # Send as bytes

if __name__ == "__main__":
    main()
```

Receiver.py

```
import socket

def main():
    # Create a server socket
    with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as
server_socket:
        server_socket.bind(('localhost', 6017))
        server_socket.listen(1)  # Listen for incoming connections
        print("Waiting for a connection...")

        # Accept a connection
        conn, addr = server_socket.accept()
        with conn:
            print(f"Connected by {addr}")

            # Read the incoming data
            ct = conn.recv(1024).decode('utf-8')  # Buffer size is 1024 bytes
            s = ct.split(",")  # Split the message and key
```

```
        encrypted_message = s[0]
        key = list(map(int, s[1:]))  # Convert key strings to integers

        print(f"Encrypted message: {encrypted_message}")

        # Decrypt the message
        pt = ''
        for i in range(len(encrypted_message)):
            j = key[i]
            pt += chr(ord(encrypted_message[i]) - j)  # Decrypt character
            print(f"Key={j}")

        print(f"Message from Sender: {pt}")

if __name__ == "__main__":
    main()
```

FIRST RUN RECEIVER THEN SENDER

receiver.py - C:\Users\admin\Downloads\receiver.py (3.11.0)

File Edit Format Run Options Window Help

```python
import socket

def main():
    # Create a server socket
    with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as server_socket:
        server_socket.bind(('localhost', 6017))
        server_socket.listen(1)  # Listen for incoming connections
        print("Waiting for a connection...")

        # Accept a connection
        conn, addr = server_socket.accept()
        with conn:
            print(f"Connected by {addr}")

            # Read the incoming data
            ct = conn.recv(1024).decode('utf-8')  # Buffer size is 1024 bytes
            s = ct.split(",")  # Split the message and key
            encrypted_message = s[0]
            key = list(map(int, s[1:]))  # Convert key strings to integers

            print(f"Encrypted message: {encrypted_message}")

            # Decrypt the message
            pt = ''
            for i in range(len(encrypted_message)):
                j = key[i]
                pt += chr(ord(encrypted_message[i]) - j)  # Decrypt character
                print(f"Key={j}")

            print(f"Message from Sender: {pt}")

if __name__ == "__main__":
    main()
```

sender.py - C:\Users\admin\Downloads\sender.py (3.11.0)

File Edit Format Run Options Window Help

```python
import socket
import random

def main():
    # Connect to the server
    with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as sc:
        sc.connect(('localhost', 6017))

        # Get input from the user
        s = input("Enter the string: ")

        # Initialize variables
        key = []
        ct = ''

        # Encrypt the message
        for char in s:
            j = random.randint(0, 49)  # Random integer between (
            key.append(str(j))          # Append to key list
            ct += chr(ord(char) + j)   # Encrypt character

            print(f"j={j}")             # Print the random number

        # Convert key list to a comma-separated string
        key_str = ','.join(key)

        # Print the key and encrypted message
        print(f"Key={key_str}")
        print(f"Encrypted message: {ct}")

        # Send the encrypted message and key
        message = f"{ct},{key_str}"
        sc.sendall(message.encode('utf-8'))  # Send as bytes

if __name__ == "__main__":
```

OUTPUT

```
IDLE Shell 3.11.0
File  Edit  Shell  Debug  Options  Window  Help
Python 3.11.0 (main, Oct 24 2022, 18:26:48) [MSC
v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()
" for more information.
>>>
================== RESTART: C:\Users\admin\Downlo
ads\sender.py =================
Enter the string: SHRAVANI
j=35
j=24
j=21
j=34
j=9
j=13
j=36
j=10
Key=35,24,21,34,9,13,36,10
Encrypted message: v`gc_NrS
>>>
```

```
IDLE Shell 3.11.0
File  Edit  Shell  Debug  Options  Window  Help
Python 3.11.0 (main, Oct 24 2022, 18:26:48) [MSC
v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()
" for more information.
>>>
================= RESTART: C:\Users\admin\Downloa
ds\receiver.py ================
Waiting for a connection...
Connected by ('127.0.0.1', 58491)
Encrypted message: v`gc_NrS
Key=35
Key=24
Key=21
Key=34
Key=9
Key=13
Key=36
Key=10
Message from Sender: SHRAVANI
>>>
```

PRACTICAL NO 2

```
import logging

# Configure the logger
logger = logging.getLogger('cfprac2')
logger.setLevel(logging.DEBUG)

# Create a file handler that logs debug and higher level messages
file_handler = logging.FileHandler('D:/mylogfile.log', mode='a')  # 'a' for append mode
file_handler.setLevel(logging.DEBUG)

# Create a formatter and set it for the handler
formatter = logging.Formatter('%(asctime)s - %(levelname)s - %(message)s')
file_handler.setFormatter(formatter)

# Add the file handler to the logger
logger.addHandler(file_handler)
```

Log messages

logger.info("My first log")

logger.info("This is CFL Prac 2")

print("Logging complete. Check 'D:/mylogfile.log' for details.")

log.py - C:\Users\admin\Downloads\log.py (3.11.0)

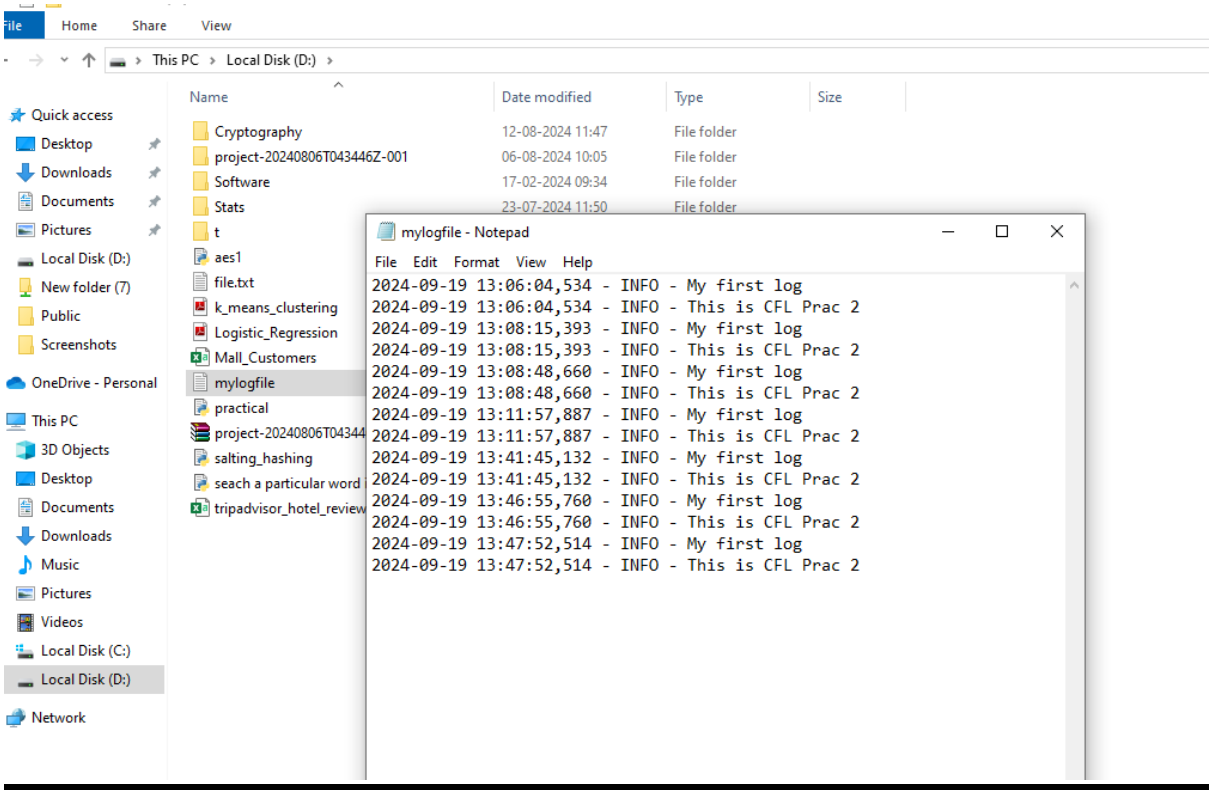
File Edit Format Run Options Window Help

```
import logging

# Configure the logger
logger = logging.getLogger('cfprac2')
logger.setLevel(logging.DEBUG)

# Create a file handler that logs debug and higher level messages
file_handler = logging.FileHandler('D:/mylogfile.log', mode='a')  # 'a' for ap
file_handler.setLevel(logging.DEBUG)

# Create a formatter and set it for the handler
formatter = logging.Formatter('%(asctime)s - %(levelname)s - %(message)s')
file_handler.setFormatter(formatter)

# Add the file handler to the logger
logger.addHandler(file_handler)

# Log messages
logger.info("My first log")
logger.info("This is CFL Prac 2")

print("Logging complete. Check 'D:/mylogfile.log' for details.")
```

IDLE Shell 3.11.0

File Edit Shell Debug Options Window Help

```
Python 3.11.0 (main, Oct 24 2022, 18:26:48) [MSC v.1933 64 bit (AMD64)] on win3
Type "help", "copyright", "credits" or "license()" for more information.
>>>
==================== RESTART: C:\Users\admin\Downloads\log.py ==================
Logging complete. Check 'D:/mylogfile.log' for details.
>>>
```



PRACTICAL NO 3

WWRITE A PROGRAM FOR SEACHING FILE IN GIVEN DIRECTORY

```
 import os

def main():
    # Get directory from user
    dir_path = input("Enter Directory: ")

    # Get first letter of file from user
    first_letter = input("Enter first letter of file: ")

    try:
        # List files in the directory
        files = os.listdir(dir_path)
```

```
        # Filter files that start with the specified letter
        filtered_files = [file for file in files if file.startswith(first_letter)]

        if not filtered_files:
            print("No files found starting with that letter.")
        else:
            for filename in filtered_files:
                print(filename)

    except FileNotFoundError:
        print("Either dir does not exist or is not a directory")
    except Exception as e:
        print(f"An error occurred: {e}")

if __name__ == "__main__":
    main()
```

searching file in given directory.py - C:\Users\admin\Downloads\searching file in given directory.py (3.11.0)

File Edit Format Run Options Window Help

```python
import os

def main():
    # Get directory from user
    dir_path = input("Enter Directory: ")

    # Get first letter of file from user
    first_letter = input("Enter first letter of file: ")

    try:
        # List files in the directory
        files = os.listdir(dir_path)

        # Filter files that start with the specified letter
        filtered_files = [file for file in files if file.startswith(first_letter

        if not filtered_files:
            print("No files found starting with that letter.")
        else:
            for filename in filtered_files:
                print(filename)

    except FileNotFoundError:
        print("Either dir does not exist or is not a directory")
    except Exception as e:
        print(f"An error occurred: {e}")

if __name__ == "__main__":
    main()
```

IDLE Shell 3.11.0

File Edit Shell Debug Options Window Help

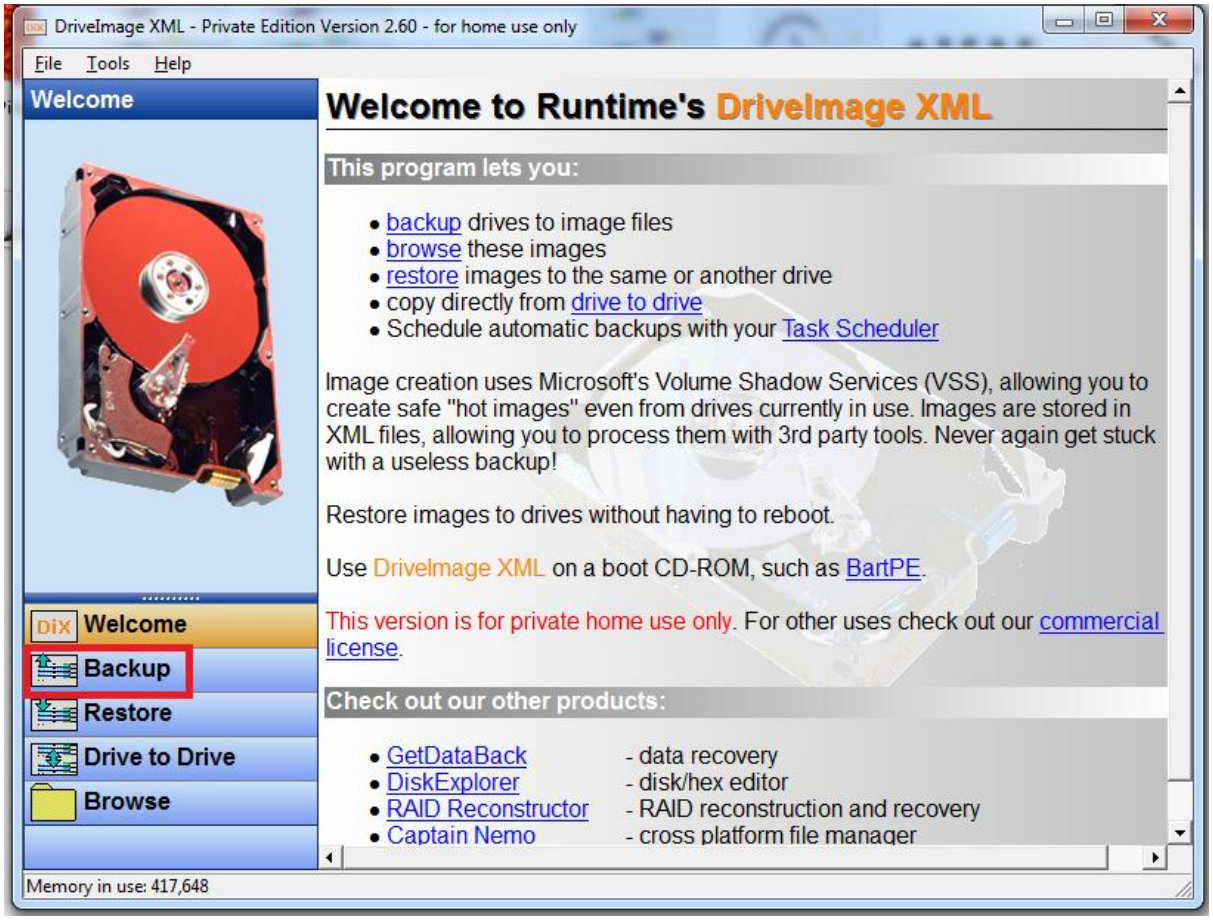
```
Python 3.11.0 (main, Oct 24 2022, 18:26:48) [MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
==== RESTART: C:\Users\admin\Downloads\searching file in given directory.py ====
Enter Directory: D:
Enter first letter of file: M
Mall_Customers.csv
>>>
```

PRACTICAL NO 4

WRITE A PROGRAM TO SEARCH A PARTICULAR WORD IN A FILE

PRATICAL NO 5

USE DRIVE IMAGE XML TO IMAGE A HARD DRIVE



DriveImage XML - Private Edition Version 2.60 - for home use only

File Tools Help

Backup

Select a drive you wish to backup.

The backup will create two files, a *.XML which contains the drive description and a *.DAT which contains the imaged drive's binary data.

These files can be accessed later through Browse or Restore.

DiX Welcome
Backup
Restore
Drive to Drive
Browse

Check one or more drives to backup:

Drive	Label	Type	Capacity	% used	Physical drive
C:		NTFS	244 GB	17	DISK0#2
D:		NTFS	117 GB	0	DISK0#3
E:		NTFS	164 GB	0	DISK0#4
G:	System Reserved	NTFS	100.0 MB	28	DISK0#1

Drive details:

Logical Information		Physical Information	
Drive:	D:	Drive:	DISK0
Label:		Drive name:	ST500DM002-1BD142
File system:	NTFS	Total sectors:	976,773,168
Total sectors:	245,759,992		
		Partition:	#3
Used bytes:	97,959,936 (93 MB)	Start sector on drive:	512,002,048
Free bytes:	125,731,155,968 (117 GB)	Sectors in partition:	245,760,000
Total bytes:	125,829,115,904 (117 GB)		

Next

Memory in use: 768,448

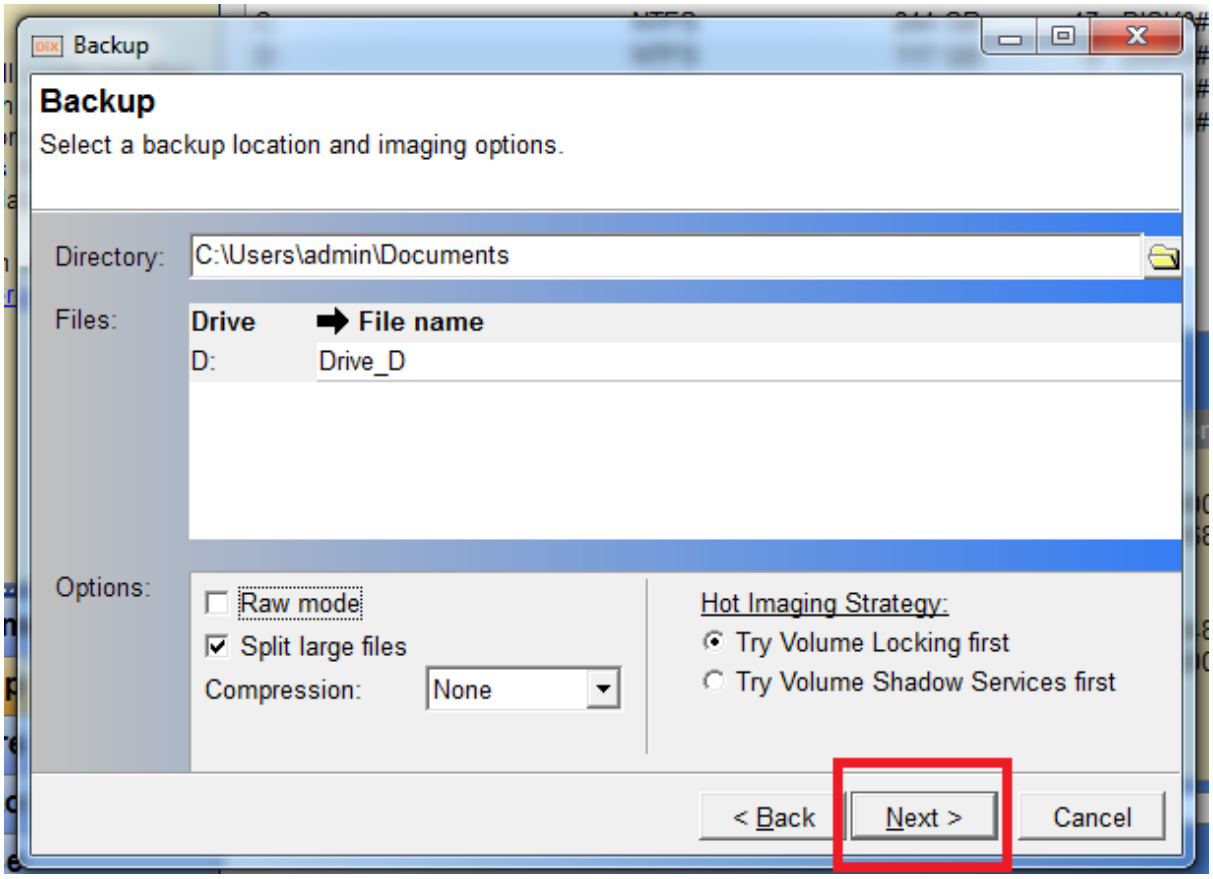
Backup

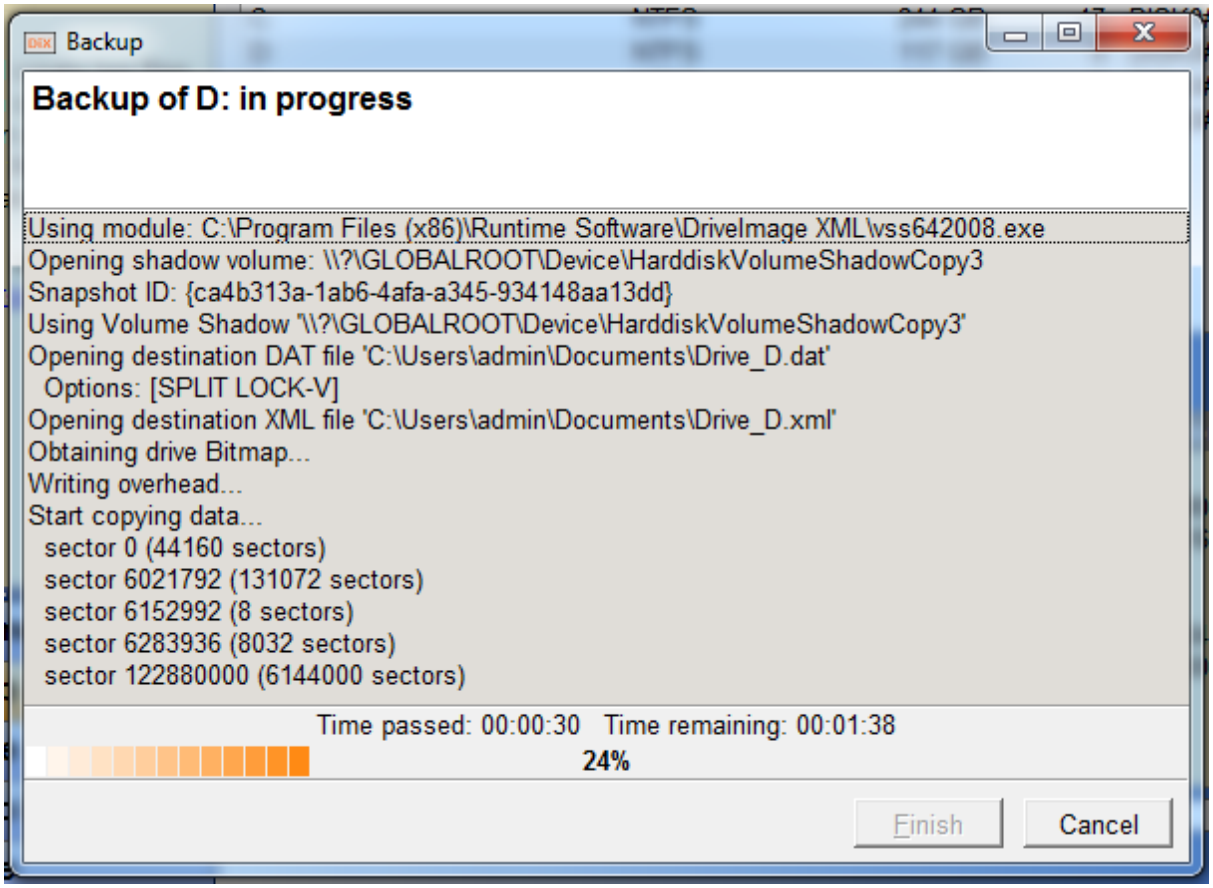
Welcome to the drive backup wizard!

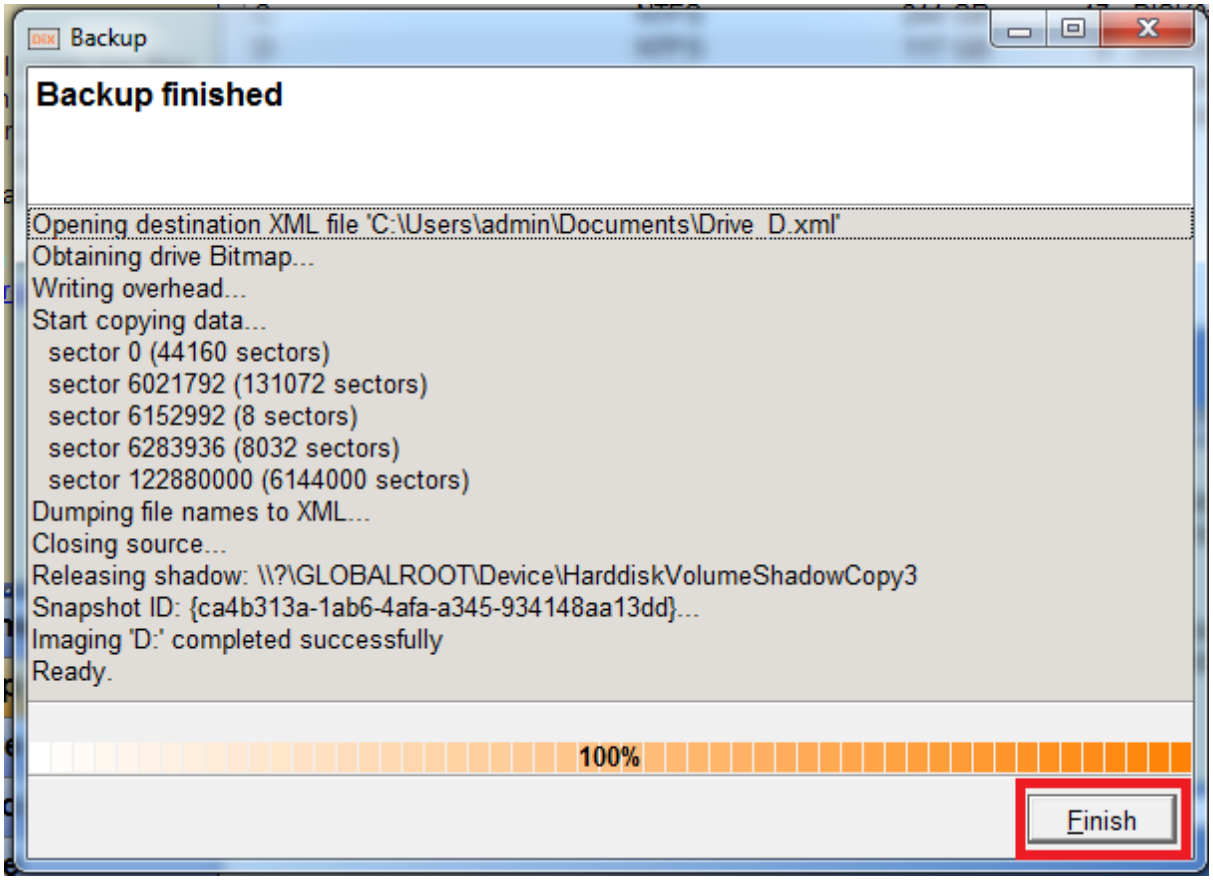
This wizard will assist you with backing up your drives:

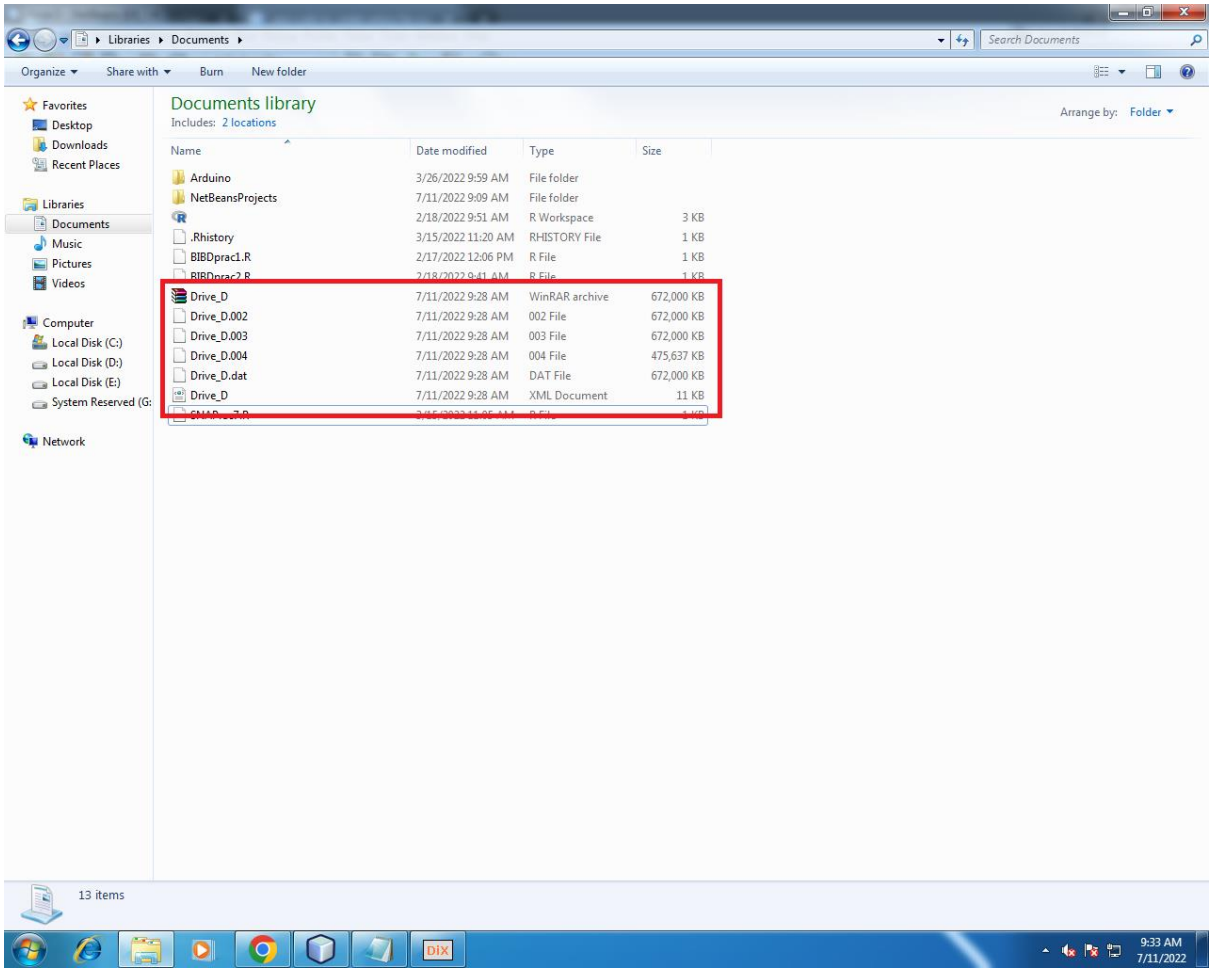
Drive	Label	Type	Capacity	% used	Physic...
D:		NTFS	117 GB	0	DISK0#3

Next > Cancel


Backup
Backup
Select a backup location and imaging options.
Directory:
C:\Users\admin\Documents
Files:
Drive
File name
D:
Drive_D
Options:
Raw mode
Split large files
Compression:
None
Hot Imaging Strategy:
Try Volume Locking first
Try Volume Shadow Services first
< Back
Next >
Cancel

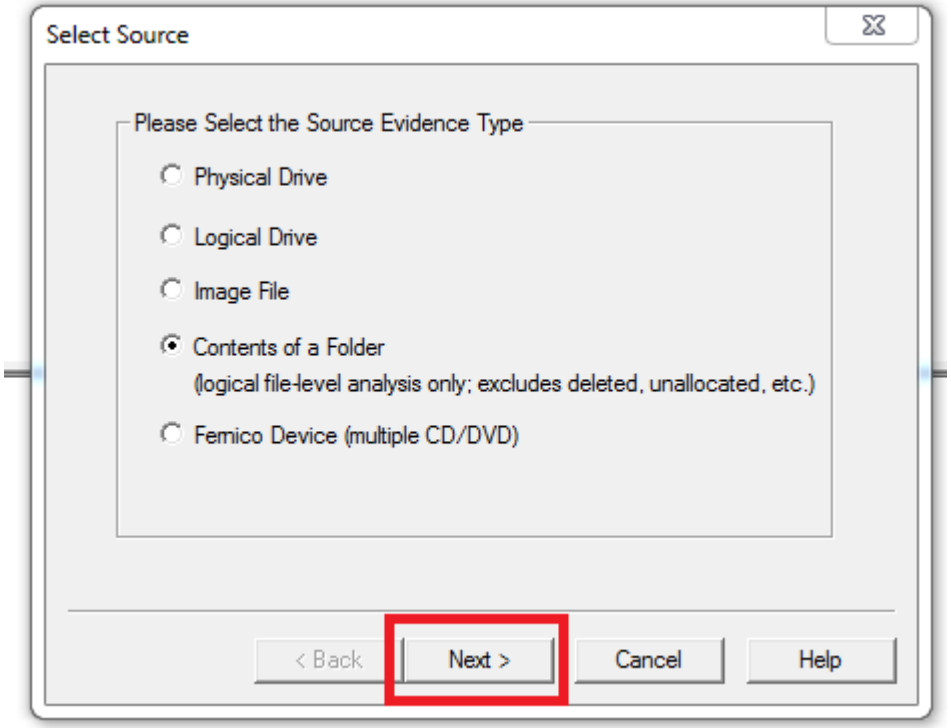

Backup
Backup of D: in progress
Using module: C:\Program Files (x86)\Runtime Software\DriveImage XML\vss642008.exe
Opening shadow volume: \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy3
Snapshot ID: {ca4b313a-1ab6-4afa-a345-934148aa13dd}
Using Volume Shadow '\\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy3'
Opening destination DAT file 'C:\Users\admin\Documents\Drive_D.dat'
Options: [SPLIT LOCK-V]
Opening destination XML file 'C:\Users\admin\Documents\Drive_D.xml'
Obtaining drive Bitmap...
Writing overhead...
Start copying data...
sector 0 (44160 sectors)
sector 6021792 (131072 sectors)
sector 6152992 (8 sectors)
sector 6283936 (8032 sectors)
sector 122880000 (6144000 sectors)
Time passed: 00:00:30 Time remaining: 00:01:38
24%
Finish
Cancel

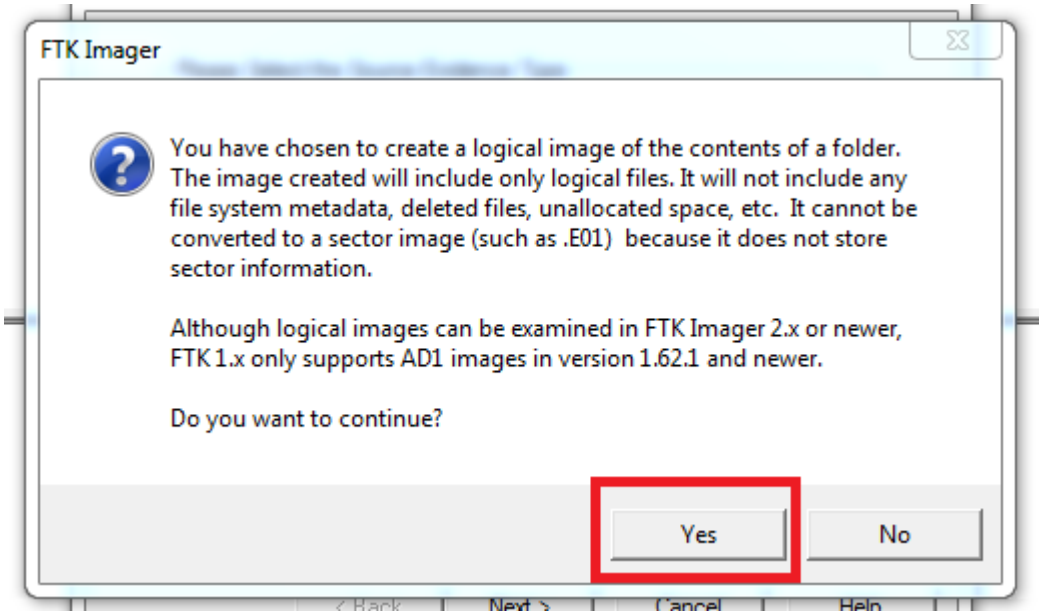

Backup
Backup finished
Opening destination XML file 'C:\Users\admin\Documents\Drive_D.xml'
Obtaining drive Bitmap...
Writing overhead...
Start copying data...
sector 0 (44160 sectors)
sector 6021792 (131072 sectors)
sector 6152992 (8 sectors)
sector 6283936 (8032 sectors)
sector 122880000 (6144000 sectors)
Dumping file names to XML...
Closing source...
Releasing shadow: \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy3
Snapshot ID: {ca4b313a-1ab6-4afa-a345-934148aa13dd}...
Imaging 'D:' completed successfully
Ready.
100%
Finish

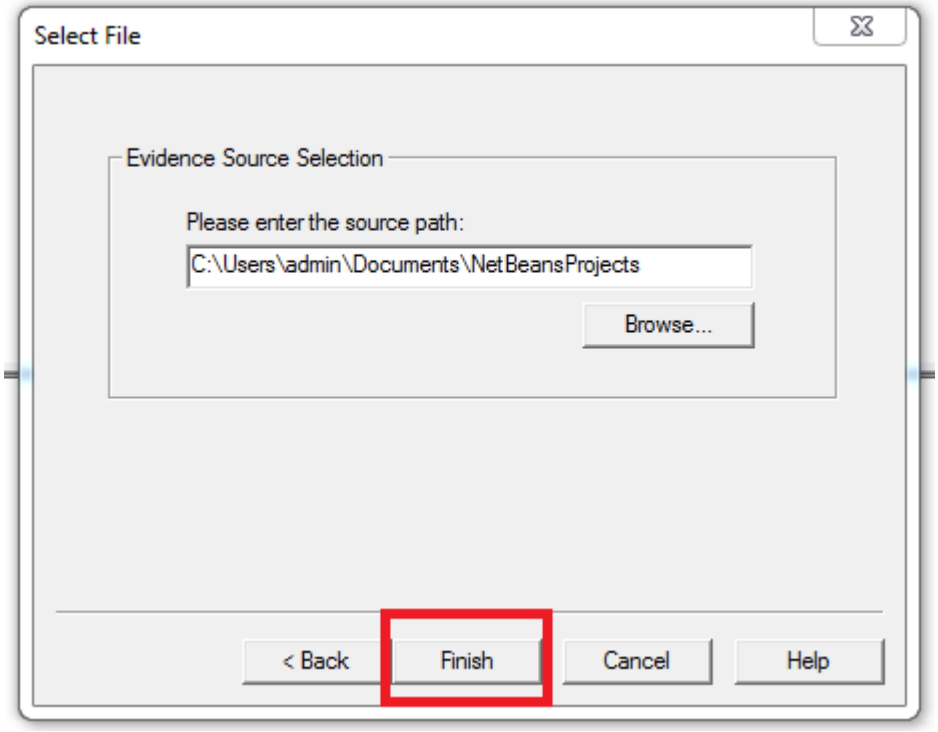

Libraries ▸ Documents ▸
Search Documents
Organize
Share with
Burn
New folder
Favorites
Desktop
Downloads
Recent Places
Libraries
Documents
Music
Pictures
Videos
Computer
Local Disk (C:)
Local Disk (D:)
Local Disk (E:)
System Reserved (G:
Network
Documents library
Includes: 2 locations
Arrange by: Folder
Name
Date modified
Type
Size
Arduino 3/26/2022 9:59 AM File folder
NetBeansProjects 7/11/2022 9:09 AM File folder
2/18/2022 9:51 AM R Workspace 3 KB
.Rhistory 3/15/2022 11:20 AM RHISTORY File 1 KB
BIBDprac1.R 2/17/2022 12:06 PM R File 1 KB
Drive_D 7/11/2022 9:28 AM WinRAR archive 672,000 KB
Drive_D.002 7/11/2022 9:28 AM 002 File 672,000 KB
Drive_D.003 7/11/2022 9:28 AM 003 File 672,000 KB
Drive_D.004 7/11/2022 9:28 AM 004 File 475,637 KB
Drive_D.dat 7/11/2022 9:28 AM DAT File 672,000 KB
Drive_D 7/11/2022 9:28 AM XML Document 11 KB
13 items
9:33 AM
7/11/2022

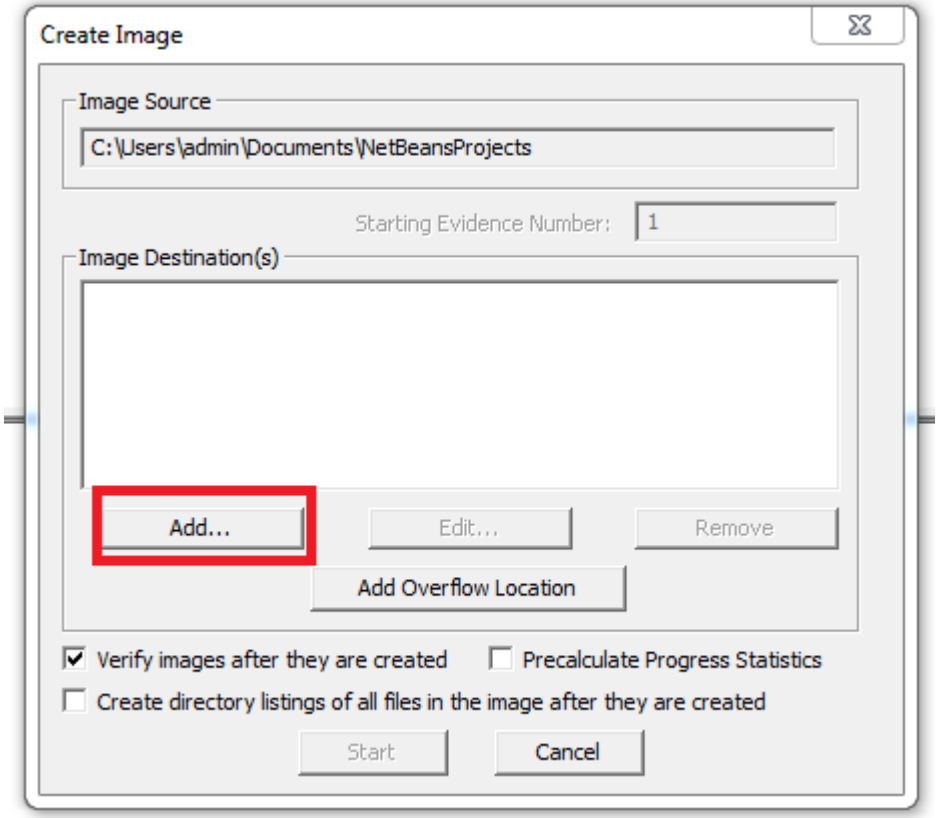
Practical 7

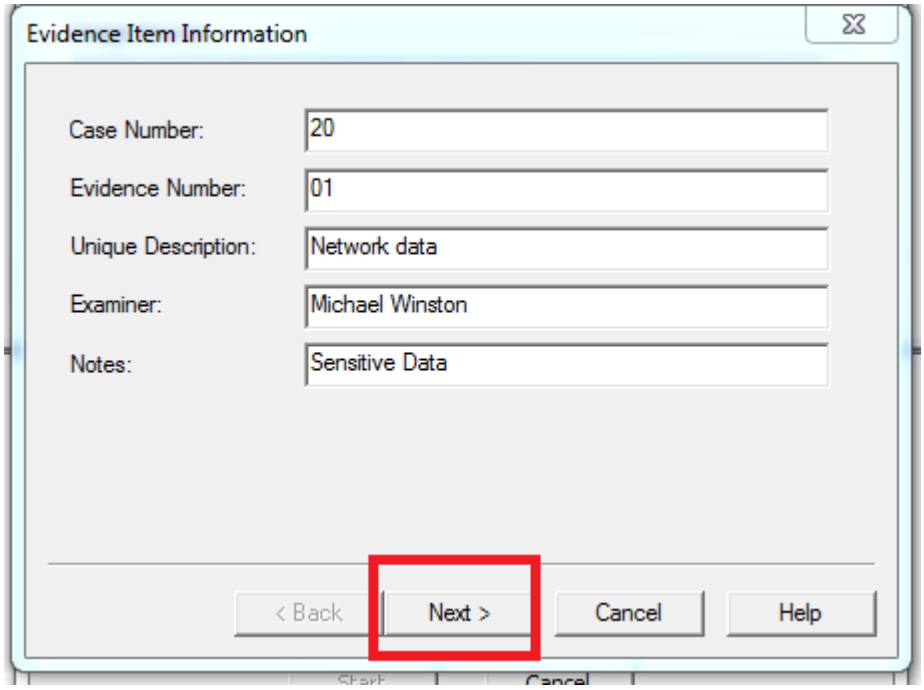
Aim:- Create forensic images of digital devices from volatile data such as memory using imager for computer system.

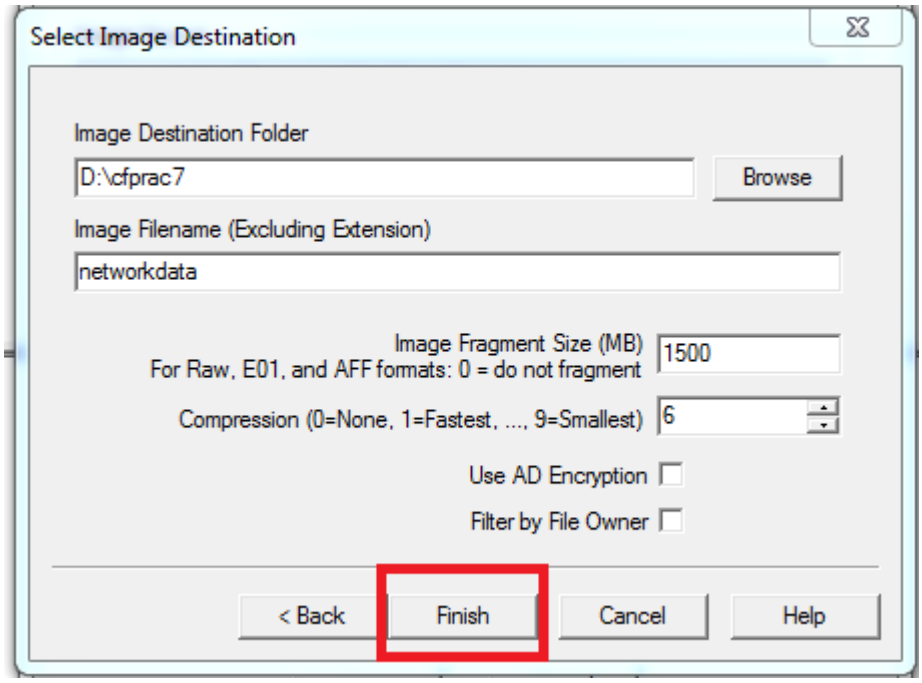


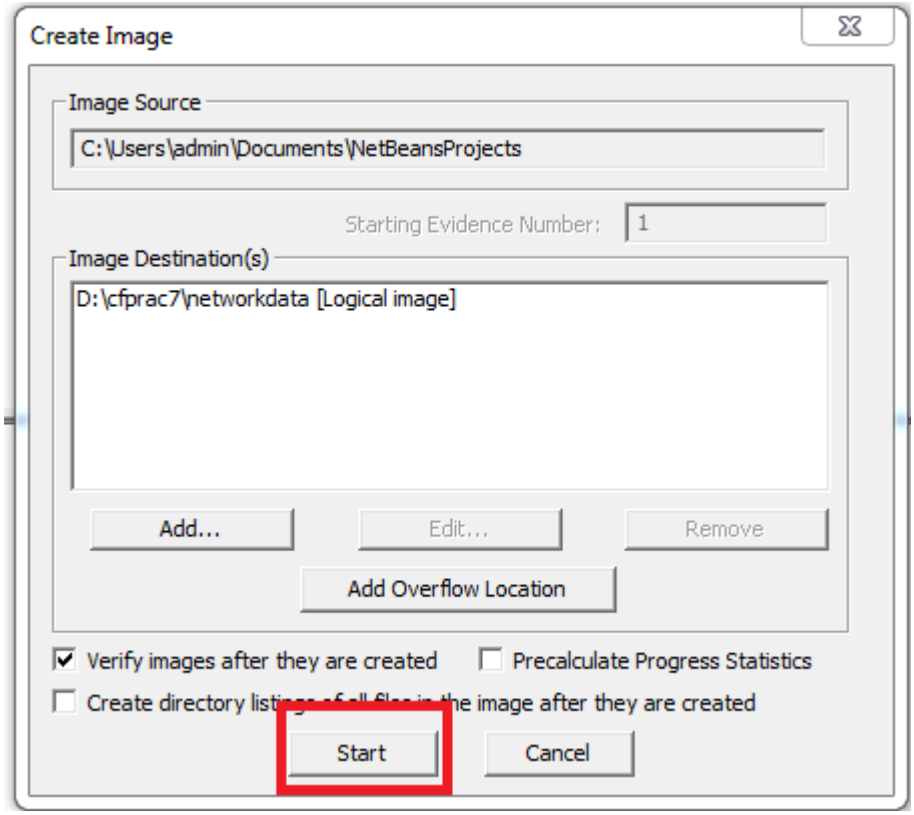


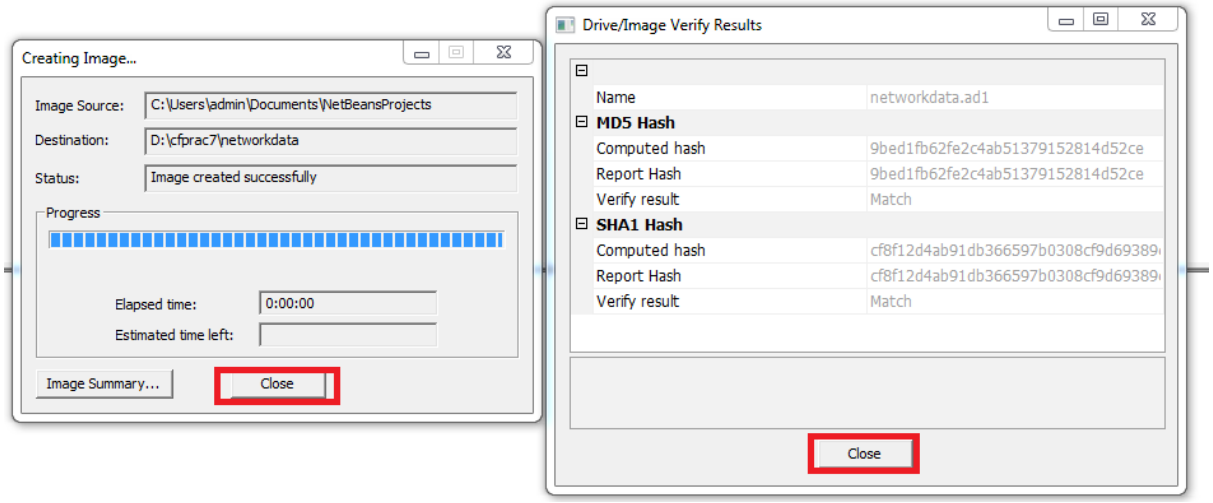

Select File
Evidence Source Selection
Please enter the source path:
C:\Users\admin\Documents\NetBeansProjects
Browse...
< Back
Finish
Cancel
Help

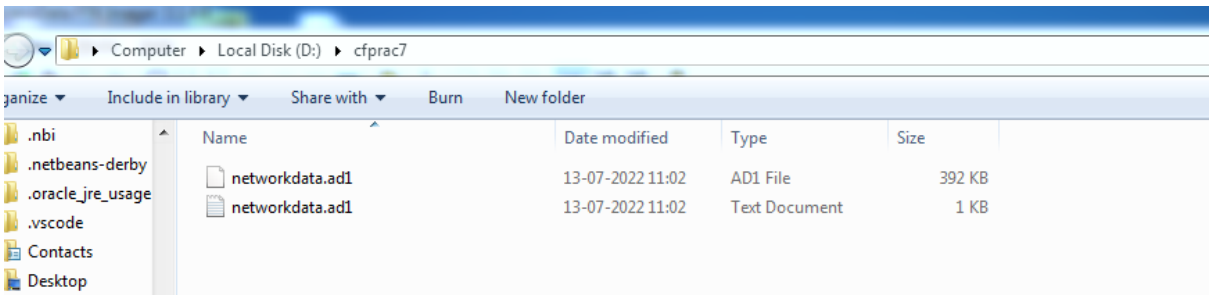

Create Image
Image Source
C:\Users\admin\Documents\NetBeansProjects
Starting Evidence Number:
1
Image Destination(s)
Add...
Edit...
Remove
Add Overflow Location
Verify images after they are created
Precalculate Progress Statistics
Create directory listings of all files in the image after they are created
Start
Cancel


Evidence Item Information
Case Number:
20
Evidence Number:
01
Unique Description:
Network data
Examiner:
Michael Winston
Notes:
Sensitive Data
< Back
Next >
Cancel
Help


Select Image Destination
Image Destination Folder
D:\cfprac7
Browse
Image Filename (Excluding Extension)
networkdata
Image Fragment Size (MB)
For Raw, E01, and AFF formats: 0 = do not fragment
1500
Compression (0=None, 1=Fastest, ..., 9=Smallest)
6
Use AD Encryption
Filter by File Owner
< Back
Finish
Cancel
Help


Create Image
Image Source
C:\Users\admin\Documents\NetBeansProjects
Starting Evidence Number:
1
Image Destination(s)
D:\cfprac7\networkdata [Logical image]
Add...
Edit...
Remove
Add Overflow Location
Verify images after they are created
Precalculate Progress Statistics
Create directory listings of all files in the image after they are created
Start
Cancel


Creating Image...
Image Source:
C:\Users\admin\Documents\NetBeansProjects
Destination:
D:\cfprac7\networkdata
Status:
Image created successfully
Progress
Elapsed time:
0:00:00
Estimated time left:
Image Summary...
Close
Drive/Image Verify Results
Name
networkdata.ad1
MD5 Hash
Computed hash
9bed1fb62fe2c4ab51379152814d52ce
Report Hash
9bed1fb62fe2c4ab51379152814d52ce
Verify result
Match
SHA1 Hash
Computed hash
cf8f12d4ab91db366597b0308cf9d69389
Report Hash
cf8f12d4ab91db366597b0308cf9d69389
Verify result
Match
Close


Computer
Local Disk (D:)
cfprac7
Include in library
Share with
Burn
New folder
.nbi
.netbeans-derby
.oracle_jre_usage
.vscode
Contacts
Desktop
Name
Date modified
Type
Size
networkdata.ad1
13-07-2022 11:02
AD1 File
392 KB
networkdata.ad1
13-07-2022 11:02
Text Document
1 KB

networkdata.ad1 - Notepad

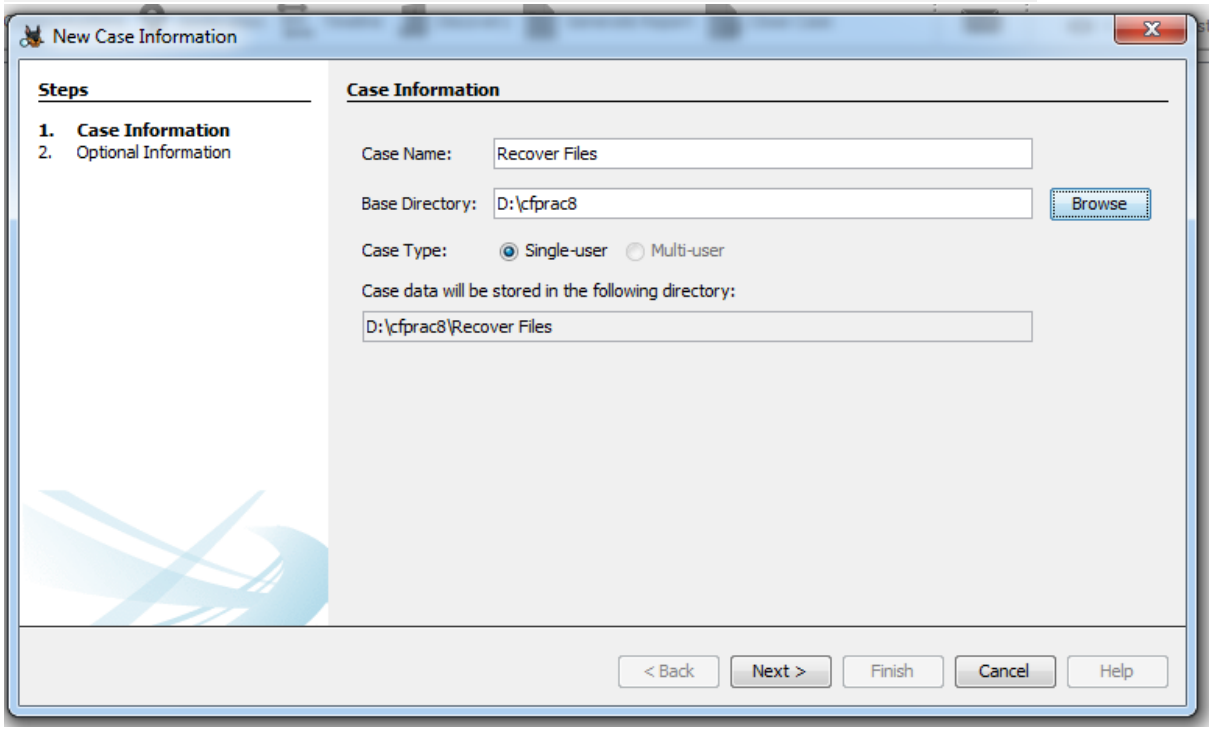
File Edit Format View Help

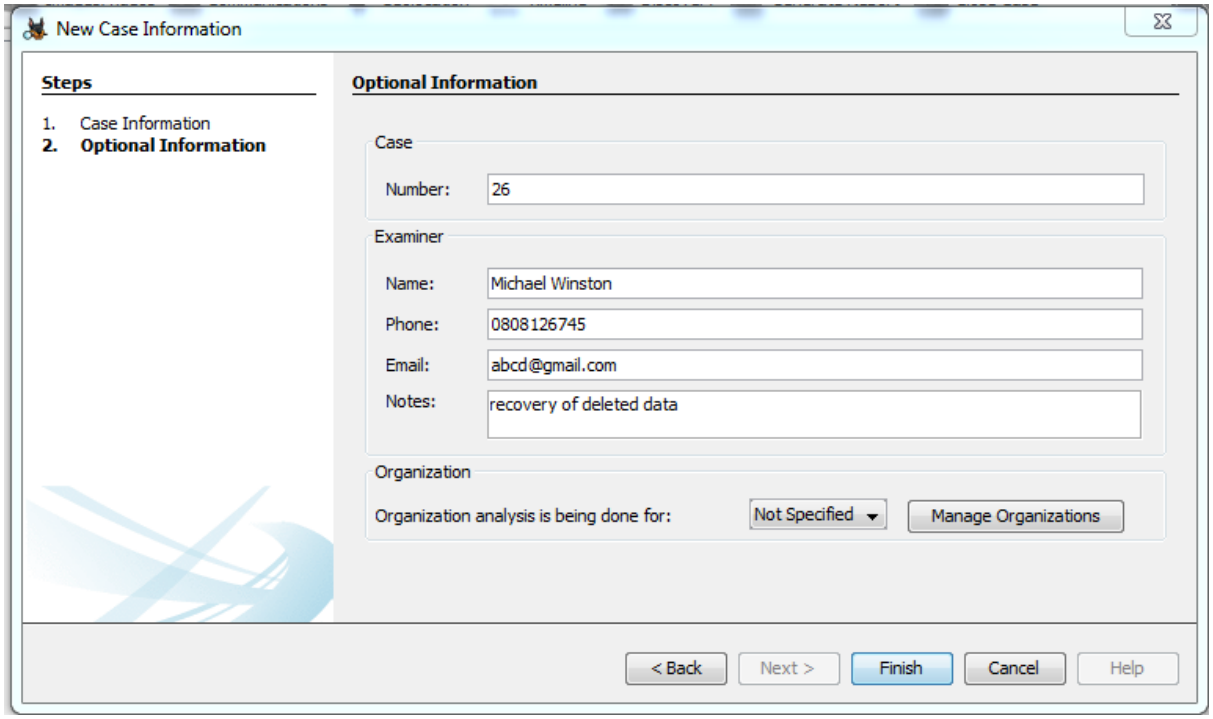
```
Created By AccessData® FTK® Imager 3.1.4.6

Case Information:
Acquired using: ADI3.1.4.6
Case Number: 20
Evidence Number: 01
Unique Description: Network data
Examiner: Michael Winston
Notes: Sensitive Data

---------------------------------------------------------------

Information for D:\cfprac7\networkdata.ad1:
[Computed Hashes]
 MD5 checksum:    9bed1fb62fe2c4ab51379152814d52ce
 SHA1 checksum:   cf8f12d4ab91db366597b0308cf9d69389cf64ff

Image information:
 Acquisition started:   Wed Jul 13 11:02:31 2022
 Acquisition finished:  Wed Jul 13 11:02:31 2022
 Segment list:
  D:\cfprac7\networkdata.ad1

Image Verification Results:
 Verification started:  Wed Jul 13 11:02:31 2022
 Verification finished: Wed Jul 13 11:02:31 2022
 MD5 checksum:    9bed1fb62fe2c4ab51379152814d52ce : verified
 SHA1 checksum:   cf8f12d4ab91db366597b0308cf9d69389cf64ff : verified
```

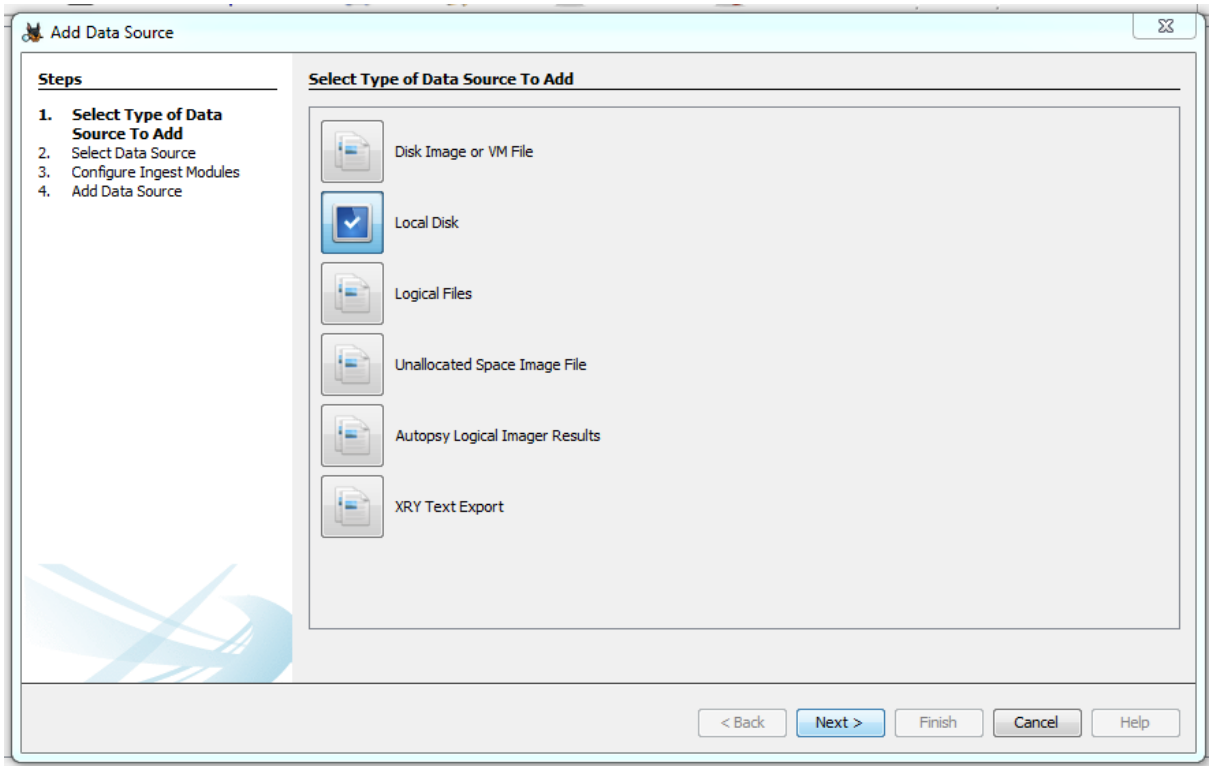
Practical 8

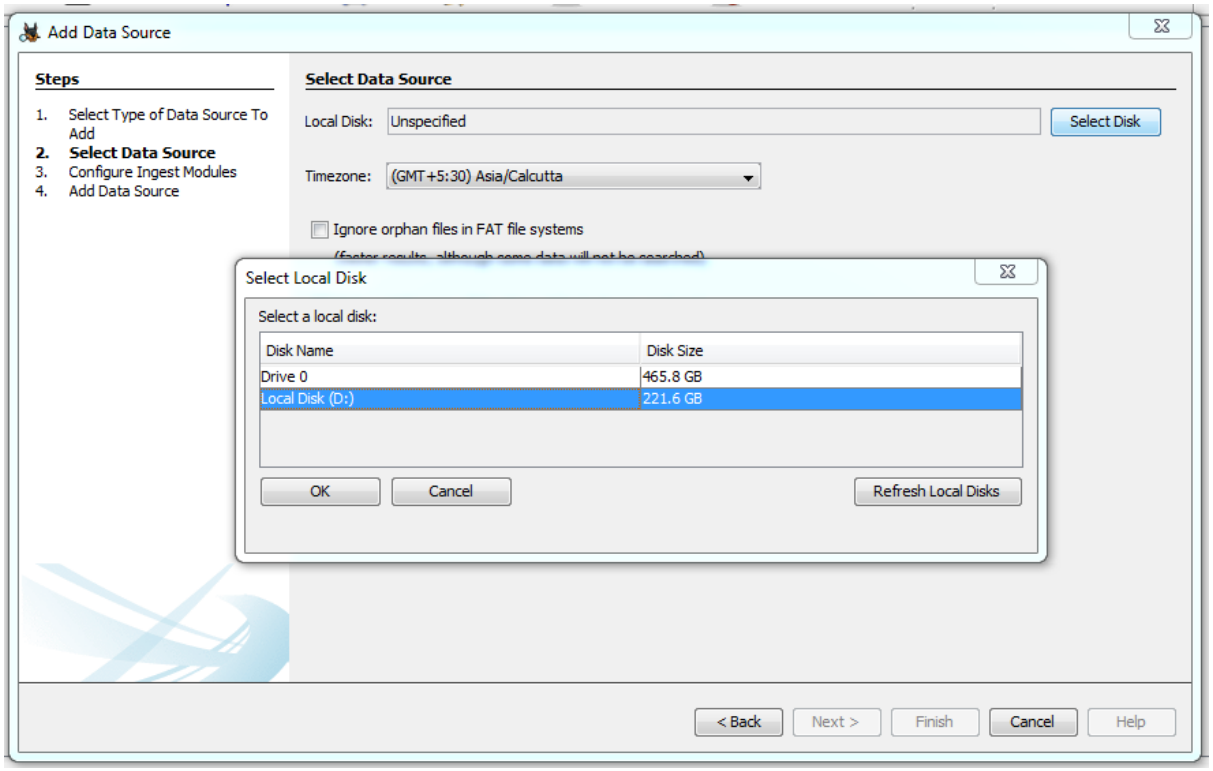
Aim:- Recovering and inspecting deleted files.

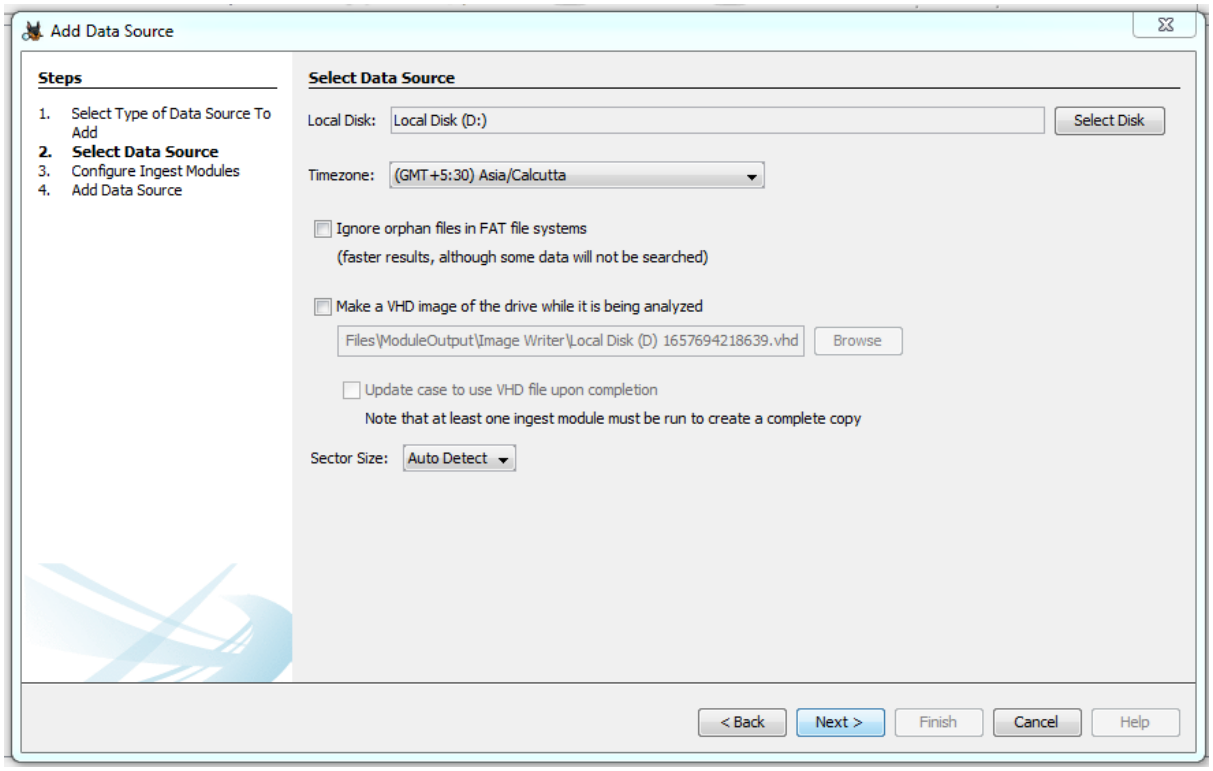

Welcome
Autopsy®
OPEN | EXTENSIBLE | FAST
New Case
Open Recent Case
Open Case
Close

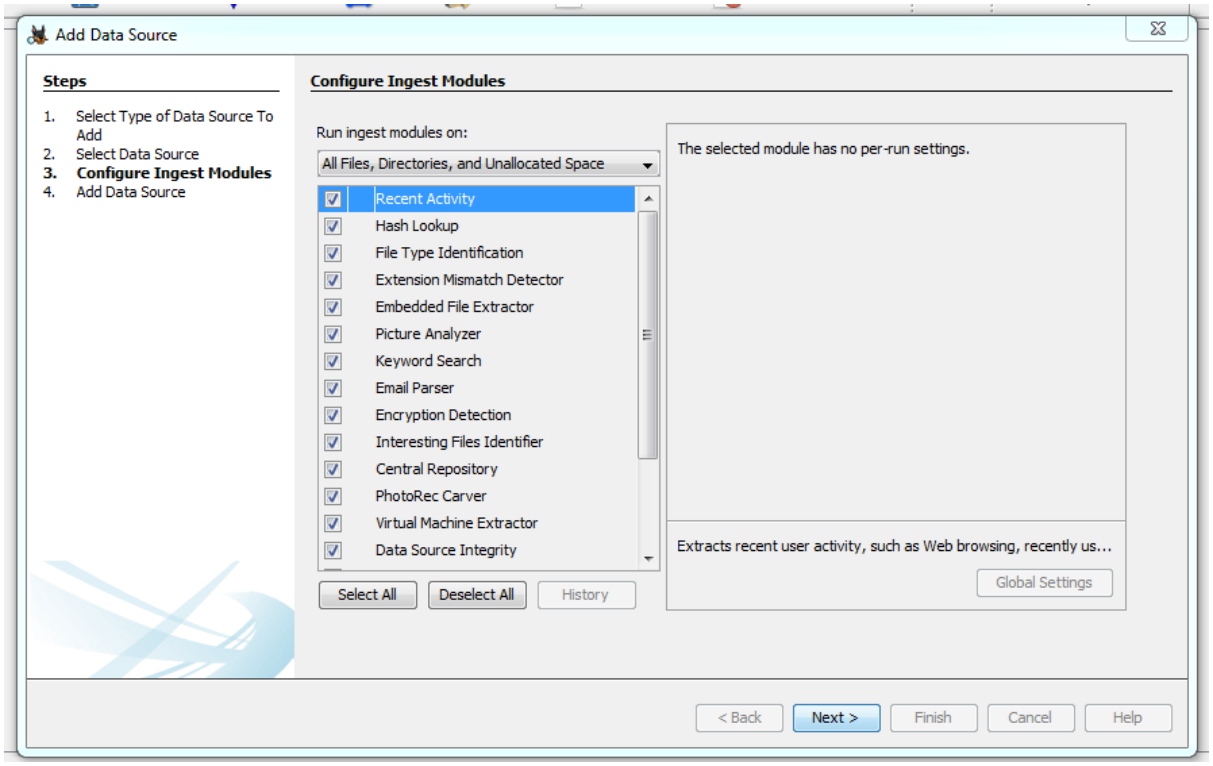

New Case Information
Steps
1. Case Information
2. Optional Information
Case Information
Case Name: Recover Files
Base Directory: D:\cfprac8
Browse
Case Type: Single-user Multi-user
Case data will be stored in the following directory:
D:\cfprac8\Recover Files
< Back
Next >
Finish
Cancel
Help

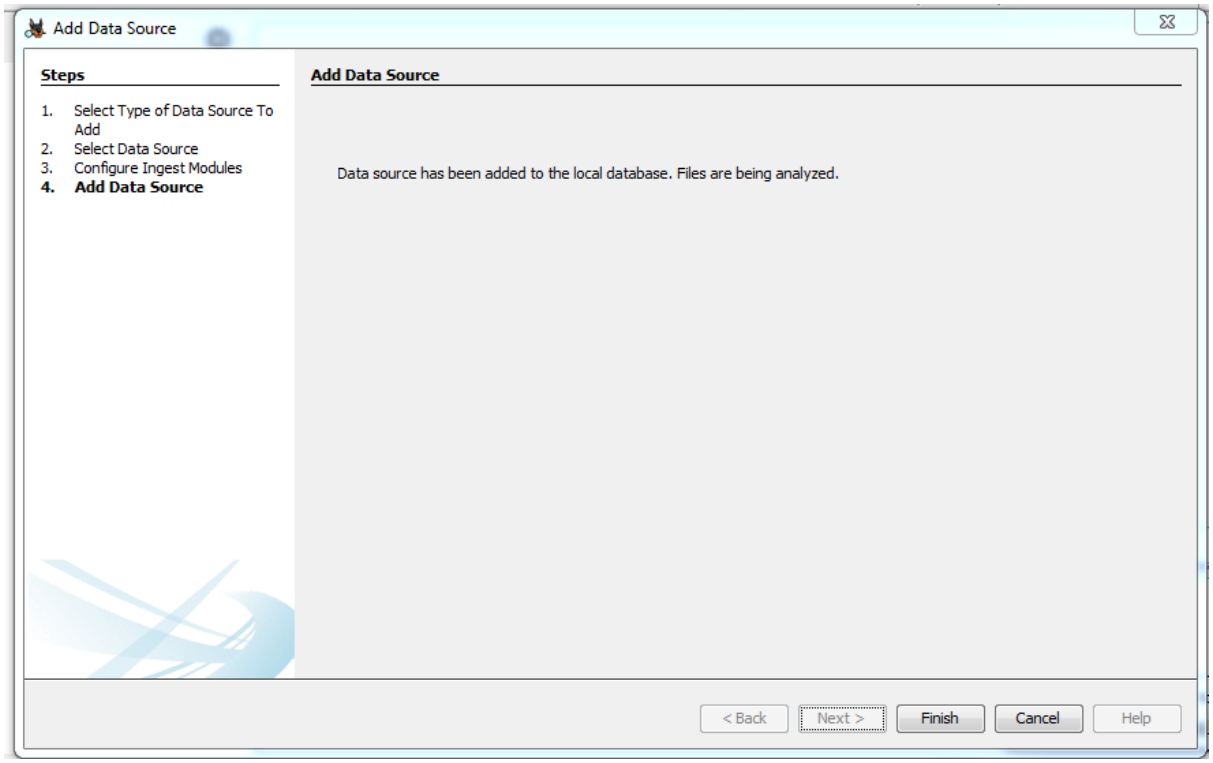

New Case Information
Steps
1. Case Information
2. Optional Information
Optional Information
Case
Number: 26
Examiner
Name: Michael Winston
Phone: 0808126745
Email: abcd@gmail.com
Notes: recovery of deleted data
Organization
Organization analysis is being done for: Not Specified
Manage Organizations
< Back
Next >
Finish
Cancel
Help

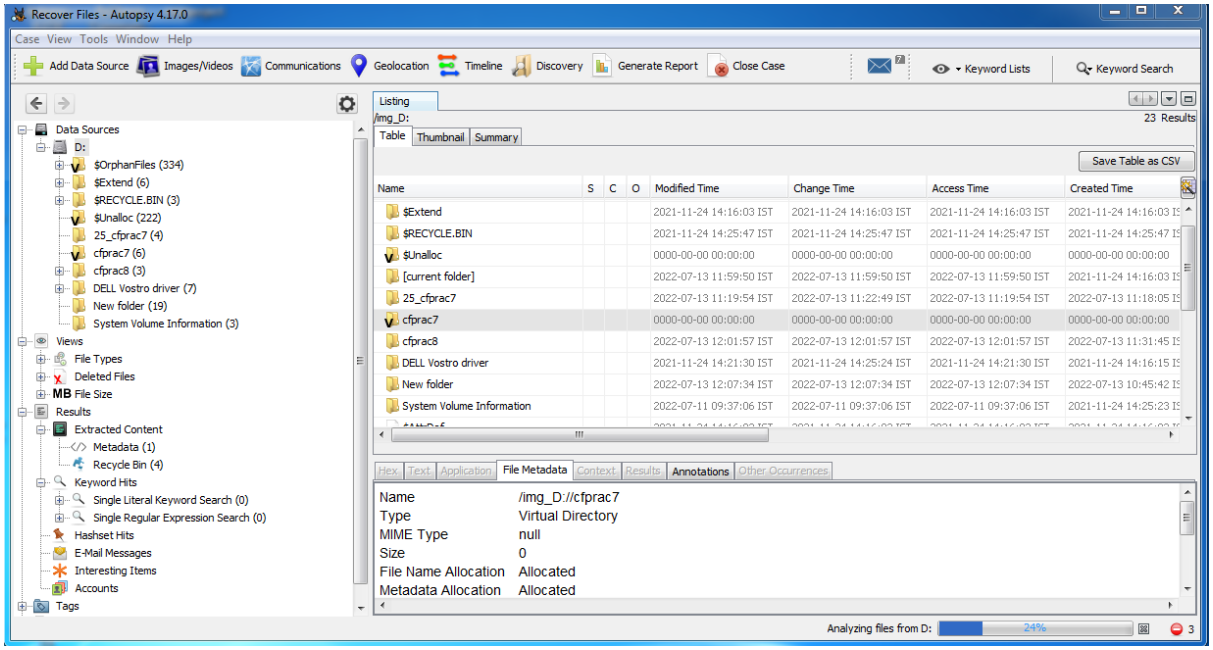

Add Data Source
Steps
1. Select Type of Data Source To Add
2. Select Data Source
3. Configure Ingest Modules
4. Add Data Source
Select Type of Data Source To Add
Disk Image or VM File
Local Disk
Logical Files
Unallocated Space Image File
Autopsy Logical Imager Results
XRY Text Export
< Back
Next >
Finish
Cancel
Help

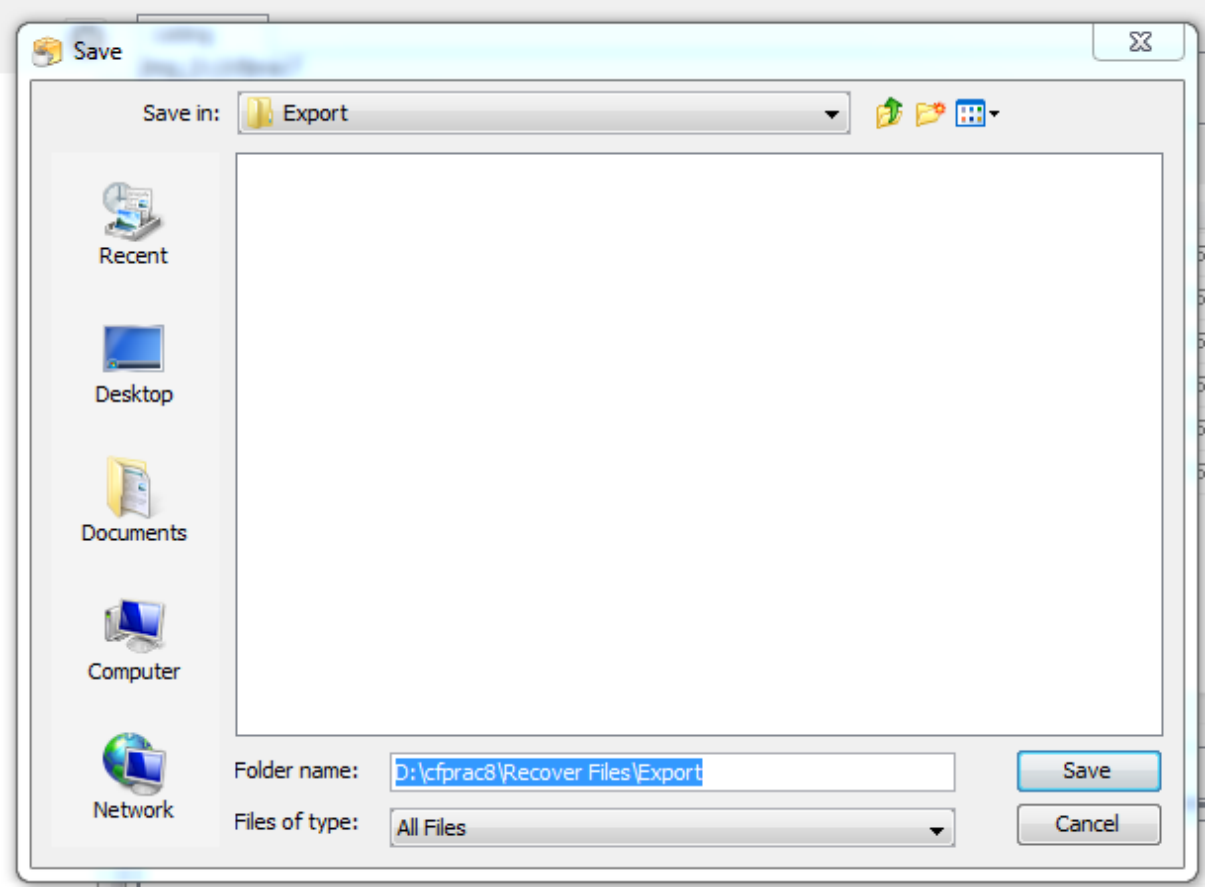

Add Data Source
Steps
1. Select Type of Data Source To Add
2. Select Data Source
3. Configure Ingest Modules
4. Add Data Source
Select Data Source
Local Disk: Unspecified
Select Disk
Timezone: (GMT+5:30) Asia/Calcutta
Ignore orphan files in FAT file systems
Select Local Disk
Select a local disk:
Disk Name
Disk Size
Drive 0
465.8 GB
Local Disk (D:)
221.6 GB
OK
Cancel
Refresh Local Disks
< Back
Next >
Finish
Cancel
Help

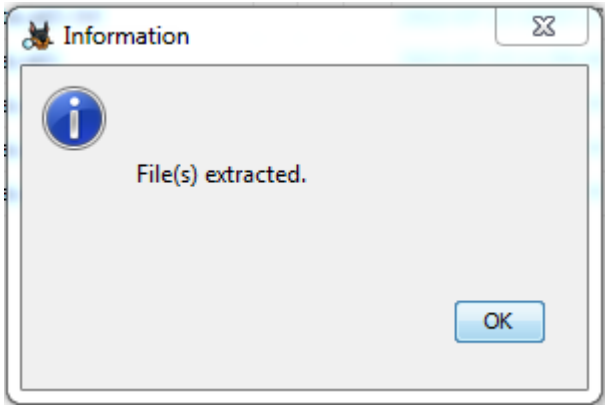

Add Data Source
Steps
1. Select Type of Data Source To Add
2. Select Data Source
3. Configure Ingest Modules
4. Add Data Source
Select Data Source
Local Disk: Local Disk (D:)
Select Disk
Timezone: (GMT+5:30) Asia/Calcutta
Ignore orphan files in FAT file systems
(faster results, although some data will not be searched)
Make a VHD image of the drive while it is being analyzed
Files\ModuleOutput\Image Writer\Local Disk (D) 1657694218639.vhd
Browse
Update case to use VHD file upon completion
Note that at least one ingest module must be run to create a complete copy
Sector Size: Auto Detect
< Back
Next >
Finish
Cancel
Help

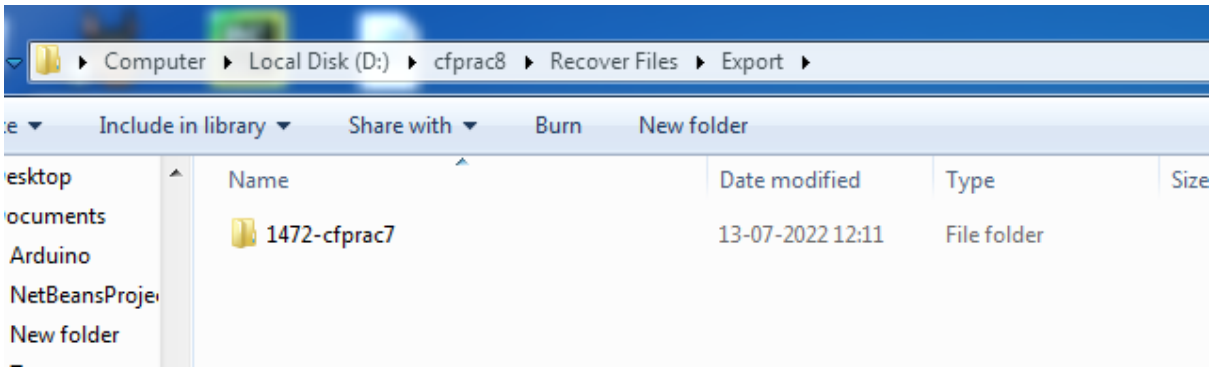

Add Data Source
Steps
1. Select Type of Data Source To Add
2. Select Data Source
3. Configure Ingest Modules
4. Add Data Source
Configure Ingest Modules
Run ingest modules on:
All Files, Directories, and Unallocated Space
Recent Activity
Hash Lookup
File Type Identification
Extension Mismatch Detector
Embedded File Extractor
Picture Analyzer
Keyword Search
Email Parser
Encryption Detection
Interesting Files Identifier
Central Repository
PhotoRec Carver
Virtual Machine Extractor
Data Source Integrity
Select All
Deselect All
History
The selected module has no per-run settings.
Extracts recent user activity, such as Web browsing, recently us...
Global Settings
< Back
Next >
Finish
Cancel
Help

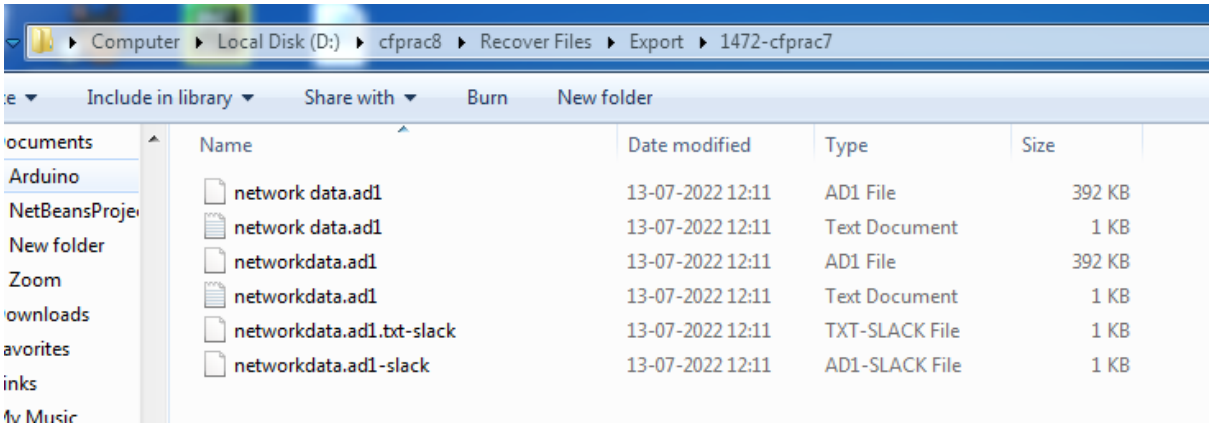

Add Data Source
Steps
1. Select Type of Data Source To Add
2. Select Data Source
3. Configure Ingest Modules
4. Add Data Source
Add Data Source
Data source has been added to the local database. Files are being analyzed.
< Back
Next >
Finish
Cancel
Help

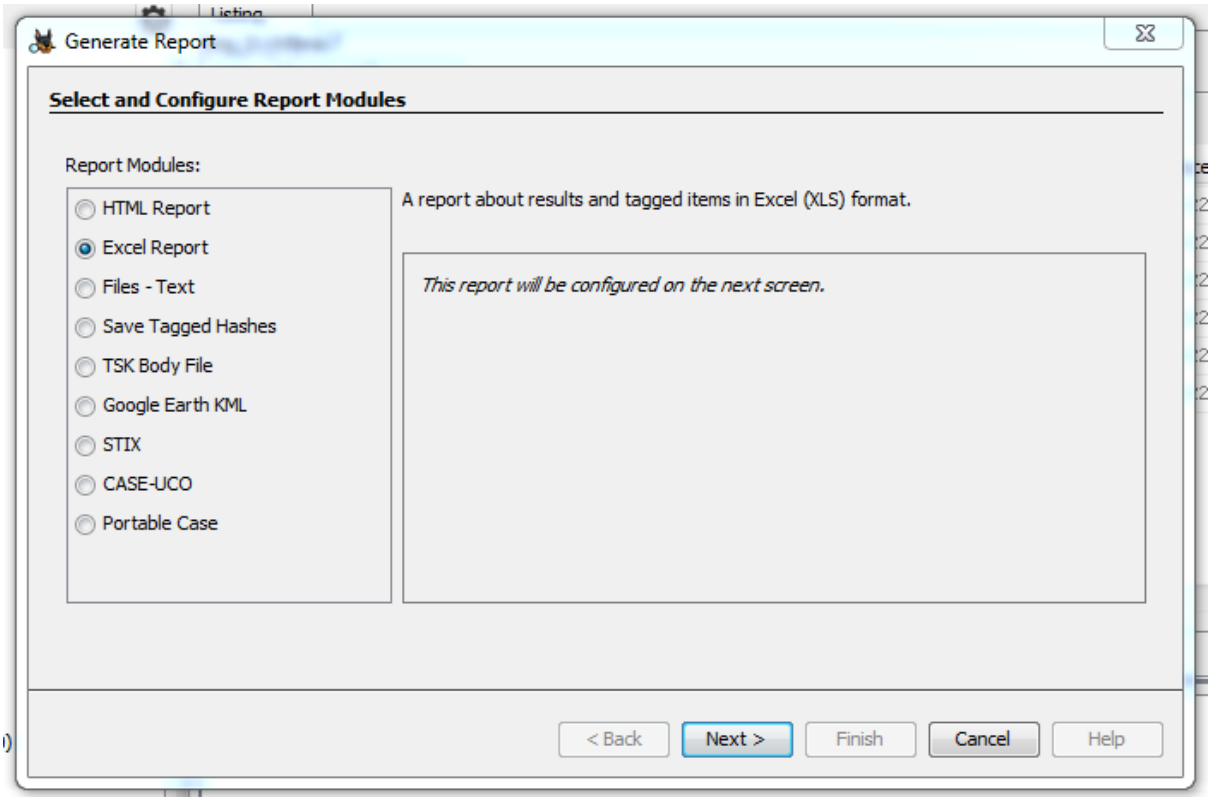

Recover Files - Autopsy 4.17.0
Case View Tools Window Help
Add Data Source
Images/Videos
Communications
Geolocation
Timeline
Discovery
Generate Report
Close Case
Keyword Lists
Keyword Search
Data Sources
D:
$OrphanFiles (334)
$Extend (6)
$RECYCLE.BIN (3)
$Unalloc (222)
25_cfprac7 (4)
cfprac7 (6)
cfprac8 (3)
DELL Vostro driver (7)
New folder (19)
System Volume Information (3)
Views
File Types
Deleted Files
File Size
Results
Extracted Content
Metadata (1)
Recycle Bin (4)
Keyword Hits
Single Literal Keyword Search (0)
Single Regular Expression Search (0)
Hashset Hits
E-Mail Messages
Interesting Items
Accounts
Tags
Listing
/img_D:
23 Results
Table
Thumbnail
Summary
Save Table as CSV
Name
S
C
O
Modified Time
Change Time
Access Time
Created Time
$Extend
$RECYCLE.BIN
$Unalloc
[current folder]
25_cfprac7
cfprac7
cfprac8
DELL Vostro driver
New folder
System Volume Information
Hex
Text
Application
File Metadata
Context
Results
Annotations
Other Occurrences
Name /img_D://cfprac7
Type Virtual Directory
MIME Type null
Size 0
File Name Allocation Allocated
Metadata Allocation Allocated
Analyzing files from D:
24%

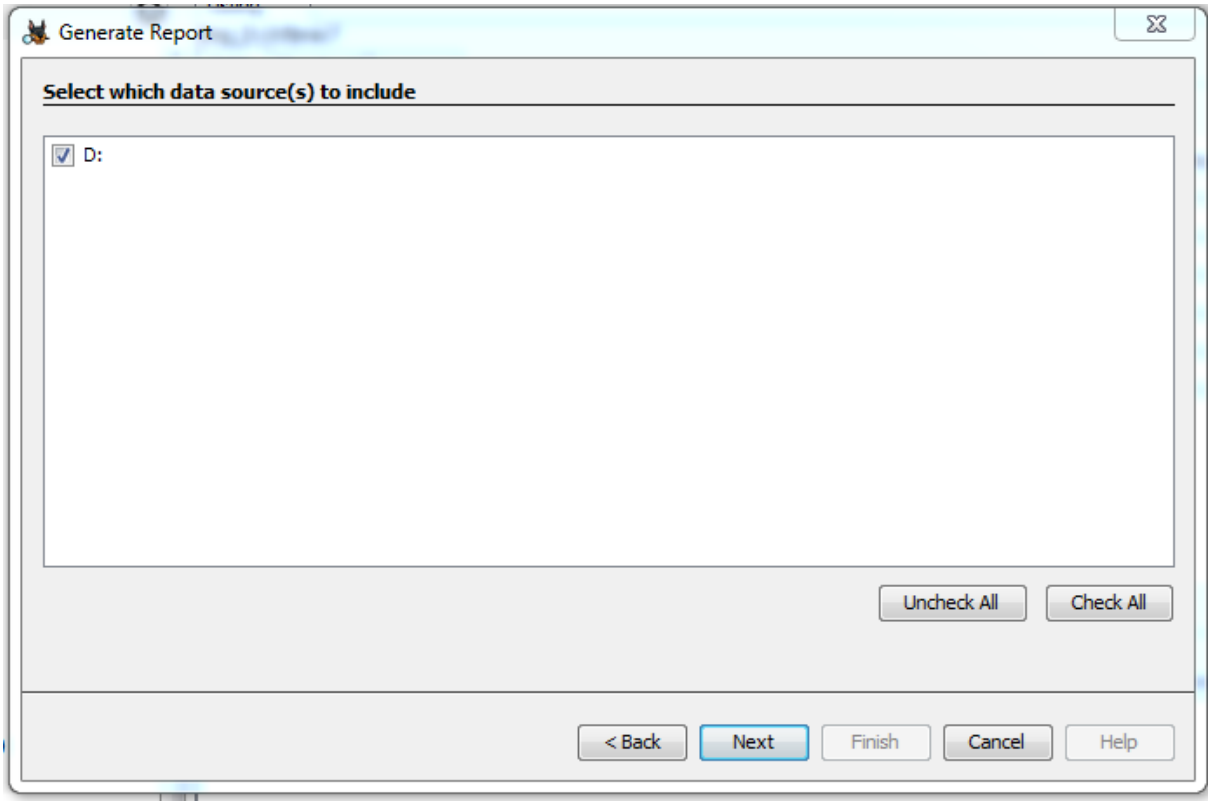

Save
Save in:
Export
Recent
Desktop
Documents
Computer
Network
Folder name:
D:\cfprac8\Recover Files\Export
Files of type:
All Files
Save
Cancel

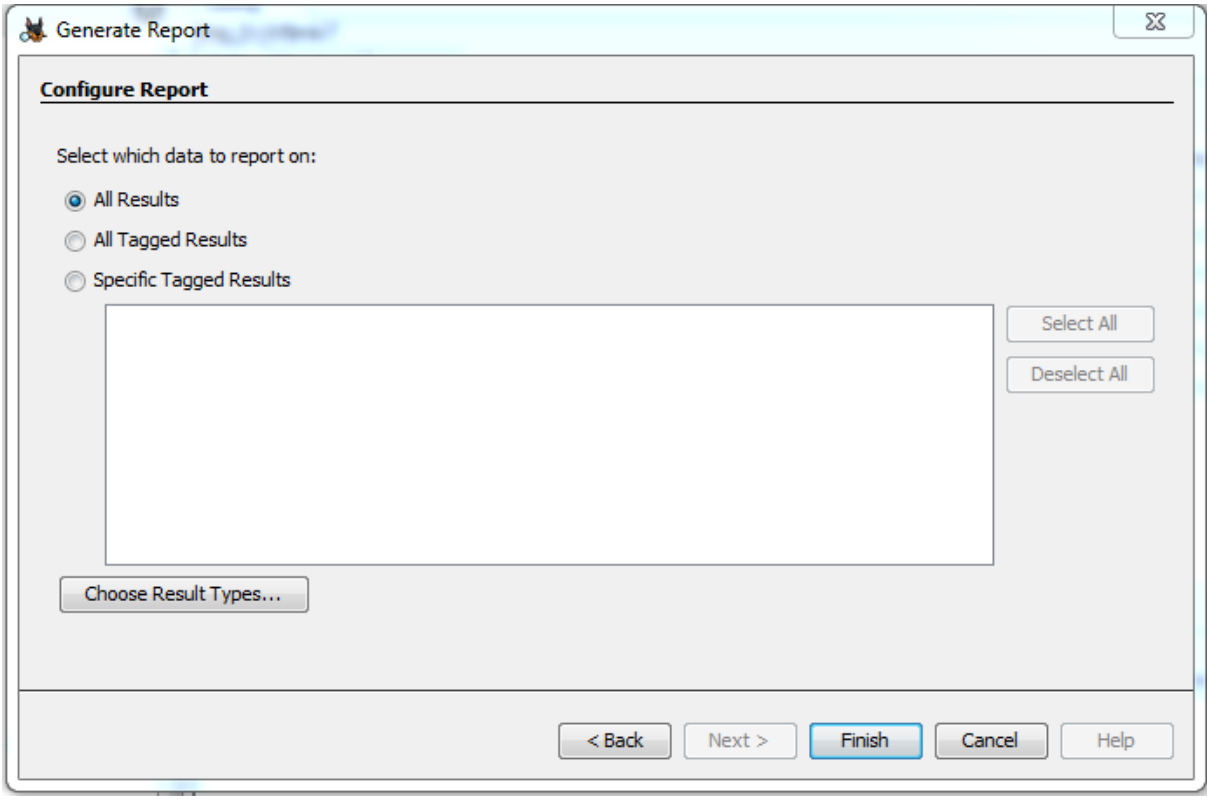

Information
File(s) extracted.
OK

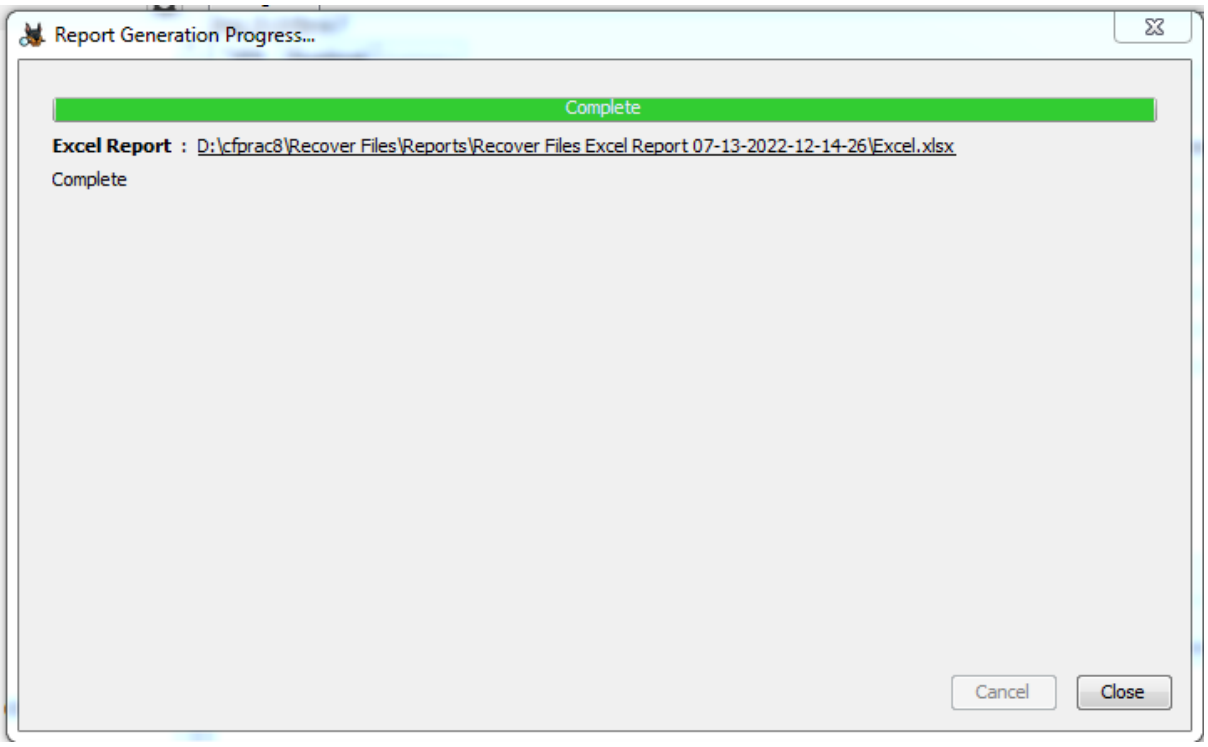

Computer ▸ Local Disk (D:) ▸ cfprac8 ▸ Recover Files ▸ Export ▸
Include in library
Share with
Burn
New folder
esktop
ocuments
Arduino
NetBeansProje
New folder
Name
Date modified
Type
Size
1472-cfprac7
13-07-2022 12:11
File folder


Computer ▸ Local Disk (D:) ▸ cfprac8 ▸ Recover Files ▸ Export ▸ 1472-cfprac7
Include in library
Share with
Burn
New folder
ocuments
Arduino
NetBeansProje
New folder
Zoom
ownloads
avorites
inks
Name
Date modified
Type
Size
network data.ad1 13-07-2022 12:11 AD1 File 392 KB
network data.ad1 13-07-2022 12:11 Text Document 1 KB
networkdata.ad1 13-07-2022 12:11 AD1 File 392 KB
networkdata.ad1 13-07-2022 12:11 Text Document 1 KB
networkdata.ad1.txt-slack 13-07-2022 12:11 TXT-SLACK File 1 KB
networkdata.ad1-slack 13-07-2022 12:11 AD1-SLACK File 1 KB


Generate Report
Select and Configure Report Modules
Report Modules:
HTML Report
Excel Report
Files - Text
Save Tagged Hashes
TSK Body File
Google Earth KML
STIX
CASE-UCO
Portable Case
A report about results and tagged items in Excel (XLS) format.
This report will be configured on the next screen.
< Back
Next >
Finish
Cancel
Help


Generate Report
Select which data source(s) to include
D:
Uncheck All
Check All
< Back
Next
Finish
Cancel
Help


Generate Report
Configure Report
Select which data to report on:
All Results
All Tagged Results
Specific Tagged Results
Select All
Deselect All
Choose Result Types...
< Back
Next >
Finish
Cancel
Help


Report Generation Progress...
Complete
Excel Report : D:\cfprac8\Recover Files\Reports\Recover Files Excel Report 07-13-2022-12-14-26\Excel.xlsx
Complete
Cancel
Close

Computer ▸ Local Disk (D:) ▸ cfprac8 ▸ Recover Files ▸ Reports ▸ Recover Files Excel Report 07-13-2022-12-14-26

Include in library ▾ Share with ▾ Burn New folder

Name	Date modified	Type	Size
Excel	13-07-2022 12:14	Microsoft Office E...	7 KB

Clipboard Font Align

A1 Summary

	A	B	C
1	Summary		
2			
3	Case Name:	Recover Files	
4	Case Number:	26	
5	Number of data sources in case:	1	
6	Case Notes:	recovery of deleted data	
7	Examiner:	Michael Winston	
8			
9			
10			
11			