**UE20CS352-OOADJ**

**Lab Assignment-7**

**Name**: Vishwa Mehul Mehta

**SRN**: PES2UG20CS389

**Section**: F

**Date**: 23-03-2023

**Summary:**

**Serialization:**

Itis the process by which we convert an object into a stream of bytes and store these bytes in file systems/databases or put them on the network to move from one location to another. To implement serialization and deserialization we use the **java.io.Serializable** interface in java. To get the file as an output stream we use **java.io.FileOutputStream**, and to accept the object output stream we use **java.io.ObjectOutputStream.**

**Deserialization:**

It is the reverse process of serialization. Deserialization consists of retrieving the objects from the byte stream.

**HashMap:**

A HashMap stores items in key/value pairs, and we can access them by an index of another type (such as a string). It is implemented using **java. util.HashMap** in java.

**Code:**

**Serializing.java:**

import java.io.FileOutputStream;

import java.io.IOException;

import java.io.ObjectOutputStream;

import java.util.HashMap;

import java.io.File;

import java.util.Scanner;

public class Serializing {

public static void main(String[] args)

{

HashMap<String, String> config = new HashMap<>();

try {

File file = new File("config.cfg");

if (!file.exists()) {

file.createNewFile();

FileOutputStream myFileOutStream = new FileOutputStream(file);

ObjectOutputStream myObjectOutStream = new ObjectOutputStream(myFileOutStream);

config.put("Path:", null);

config.put("Version:", null);

config.put("System\_Name:", null);

myObjectOutStream.writeObject(config);

}

else {

FileOutputStream myFileOutStream = new FileOutputStream(file);

ObjectOutputStream myObjectOutStream = new ObjectOutputStream(myFileOutStream);

String path;

String ver;

String sysname;

Scanner sc = new Scanner(System.in);

System.out.println("Enter the path:");

path = sc.next();

System.out.println("Enter the version:");

ver = sc.next();

System.out.println("Enter the system name:");

sysname = sc.next();

config.put("Path:", path);

config.put("Version:", ver);

config.put("System\_Name:", sysname);

//config.put("Path:", "Vishwa/Documents/Vishwa\_PES/Sem6/352\_OOAD/PES2UG20CS389");

//config.put("Version:", "17.0.6");

//config.put("System\_Name:", "Acer-Vishwa");

myObjectOutStream.writeObject(config);

myObjectOutStream.close();

myFileOutStream.close();

}

}

catch (IOException e) {

e.printStackTrace();

}

}

}

**Deserializing.java:**

import java.io.FileInputStream;

import java.io.IOException;

import java.io.ObjectInputStream;

import java.util.HashMap;

import java.util.Iterator;

import java.util.Map;

import java.util.Set;

public class Deserializing {

public static void main(String[] args)

{

HashMap<String, String> newHashMap = null;

try {

FileInputStream fileInput = new FileInputStream("config.cfg");

ObjectInputStream objectInput = new ObjectInputStream(fileInput);

newHashMap = (HashMap)objectInput.readObject();

objectInput.close();

fileInput.close();

}

catch (IOException obj1) {

obj1.printStackTrace();

return;

}

catch (ClassNotFoundException obj2) {

System.out.println("Class not found");

obj2.printStackTrace();

return;

}

Set set = newHashMap.entrySet();

Iterator iterator = set.iterator();

while (iterator.hasNext()) {

Map.Entry entry = (Map.Entry)iterator.next();

System.out.print(entry.getKey() + " ");

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

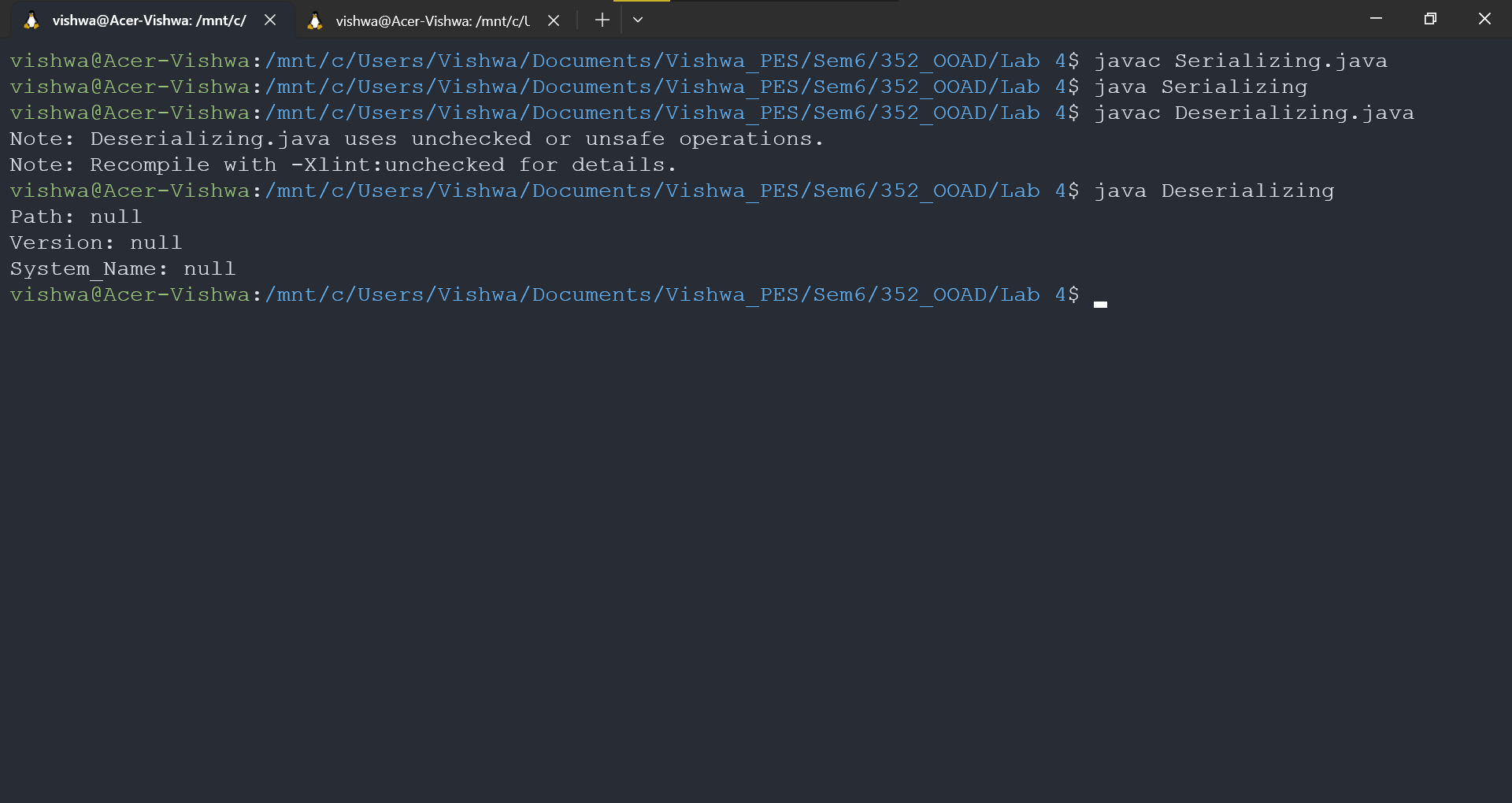
}

}

}

**Screenshots:**

**1. When config.cfg exixts:**



**2. When config.cfg exists:**

