

```

import java.io.Serializable;

public class Empleado implements Serializable
{
    private int NoEmpleado;
    private String nombre;
    private double sueldo;

    public Empleado(int noEmpleado, String nombre, double sueldo) {
        super();
        NoEmpleado = noEmpleado;
        this.nombre = nombre;
        this.sueldo = sueldo;
    }

    public int getNoEmpleado() {
        return NoEmpleado;
    }

    public void setNoEmpleado(int noEmpleado) {
        NoEmpleado = noEmpleado;
    }

    public String getNombre() {
        return nombre;
    }

    public void setNombre(String nombre) {
        this.nombre = nombre;
    }

    public double getSueldo() {
        return sueldo;
    }

    public void setSueldo(double sueldo) {
        this.sueldo = sueldo;
    }
}

public class Gerente extends Empleado
{
    private String departamento;

    public Gerente(int noEmpleado, String nombre, double sueldo, String
departamento)
    {
        super(noEmpleado, nombre, sueldo);
        this.departamento = departamento;
    }

    public String getDepartamento() {
        return departamento;
    }
}

```

```

    }

    public void setDepartamento(String departamento) {
        this.departamento = departamento;
    }

}

public class Secretaria extends Empleado
{
    private String oficina;
    private Gerente jefe;

    public Secretaria(int noEmpleado, String nombre, double sueldo, String
oficina, Gerente jefe)
    {
        super(noEmpleado, nombre, sueldo);
        this.oficina = oficina;
        this.jefe = jefe;
    }

    public String getOficina() {
        return oficina;
    }

    public void setOficina(String oficina) {
        this.oficina = oficina;
    }

    public Gerente getJefe() {
        return jefe;
    }

    public void setJefe(Gerente jefe) {
        this.jefe = jefe;
    }

}

```

```

public class Intendente extends Empleado
{
    private String asignacion;

    public Intendente(int noEmpleado, String nombre, double sueldo, String
asignacion)
    {
        super(noEmpleado, nombre, sueldo);
        this.asignacion = asignacion;
    }
}

```

```

    public String getAsignacion() {
        return asignacion;
    }

    public void setAsignacion(String asignacion) {
        this.asignacion = asignacion;
    }
}

public class tester
{

    public static void main(String[] args)
    {
        Empleado [] misEmpleados = new Empleado[5];

        Gerente elJefe = new Gerente(1,"Carlos Slim",5678.00,"Gerencia
General");

        misEmpleados[0] = elJefe;
        misEmpleados[1] = new Secretaria(2,"Maria la del
Barrio",234.50,"Compras",elJefe);
        misEmpleados[2] = new
Secretaria(3,"Marimar",225.60,"Ventas",elJefe);
        misEmpleados[3] = new Intendente(4,"Roberto Gomez",123.45,"Baños");
        misEmpleados[4] = new Intendente(5,"Florinda
Meza",145.67,"Cocina");

        //Serializacion
        Utils util = new Utils();

        //util.serializa(misEmpleados, "misEmpleados.ser");

        //Deserializacion
        Empleado [] recuperados = util.deserializa("misEmpleados.ser");
        util.imprimeEmpleados(recuperados);
    }
}

import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.ObjectInputStream;
import java.io.ObjectOutputStream;

public class Utils
{

    public void serializa(Empleado [] emp, String arch)
    {
        try
        {

```

```

        FileOutputStream fileOut =
        new FileOutputStream(arch);
        ObjectOutputStream out = new ObjectOutputStream(fileOut);
        out.writeObject(emp);
        out.close();
        fileOut.close();
        System.out.printf("Datos serializados");
    }
    catch(IOException i)
    {
        i.printStackTrace();
    }
}

public Empleado [] deserializa(String arch)
{
    Empleado[] emp = null;
    try
    {
        FileInputStream fileIn = new FileInputStream(arch);
        ObjectInputStream in = new ObjectInputStream(fileIn);
        emp = (Empleado[]) in.readObject();
        in.close();
        fileIn.close();
    }
    catch(IOException i)
    {
        i.printStackTrace();
    }
    catch(ClassNotFoundException c)
    {
        System.out.println("Clase Empleado no encontrada");
        c.printStackTrace();
    }

    return emp;
}

public void imprimeEmpleados(Empleado [] emp)
{
    for(int i=0; i< emp.length; i++)
    {
        System.out.println("NoEmp " + emp[i].getNoEmpleado() + "
Nombre " + emp[i].getNombre() + " Sueldo " + emp[i].getSueldo());
    }
}
}

```