DATE: __/__/__ PRACTICAL : 21 PAGE No: ____

Aim : Creat, debug and run Java program based on object Serialization

Theory :

Serialization is a machanism of convert-ing the state of an object into a byte stream. Desrilization is the reserive process where the byte stream is used to recreate the actual Java object in memory. This machanism is used to persist the object

Serialization in Java is machanism of writing the state of an object into a byte stream. It is mainly used in Hibernate, RMI, JPA, EJB and JMS technologies.

The reverse opration of serialization is called desperialization where byte-stream is converted into an object The serialization and desenilization process a platform-Converted It mean you can serialize an object in a platform and deserilze in different platform.

DATE: __/__/__ PAGE No: ____

For Serializing the object, we call the writeobject() method ObjectOutputStream, and for deserialization call the readobject() method of ObjectInputstream class.

We, must have to implement the Serializable interface for serializing the object

Conclusion: Hence, we successfully Create debug and run Java program based on object Serilization.

Program :-

① program :-

```
import java.io.Serializable;
public class Student implements Serializable {
    int id;
    String name;
    public Student (int id, String name) {
        this.id = id;
        this.name = name;
    }
}
```

② program :-

```
import java.io.*;
class persist {
public static void main (String args []) throws
    Exception {
  Student S1 = new Student (211, "ram");
FileOutputStream fout = new FileOutputStream ("f.txt");
ObjectOutputStream out = new ObjectOutputStream (fout);
Out.writeObject (S1);
Out.flush ();
System.out.println ("Succes");
}
}
```

③ program :-

```
import java.io.*;
class Deposit {
    public static void main (String args []) throws
        Exception {
    ObjectInputStream in = new ObjectInputStream
                (new FileInputStream
                ("f.txt") );
    Student s = (Student) in.readObject ();
    System.out.println (s.id + " " + s.name);
        in.close ();
}
}
```