

Core Java

IO and Object Serialization

Copyright © 2007, SEED Infotech Ltd. Pune, India. All rights reserved.

Objectives

- Identify the need for persisting objects
- State the Serializable interface
- Explain Object Serialization
- Explain Object Deserialization
- Problems in Object serialization

Copyright © 2007, SEED Infotech Ltd. All rights reserved.

2

Need for Persisting Objects

- Persistence : Storing the state of an object.
- Reason? For future reuse.
- If your data will be used by only the Java program that generated it :
 - ♦ Use serialization
- If your data will be used by other programs:
 - ♦ Write a plain text file

Object Serialization

- Serializable interface

Object Serialization

- Create file output stream
- Create object output stream
- Connect them
- Call writeObject()

Object Deserialization

- Create file input stream
- Create object input stream
- Connect them
- Call readObject()

`Obj = (type of obj)objin.readObject();`

Example

```
class Student implements Serializable{
    int rollno;
    String name;
    public Student(int r, String s){
        rollno = r;
        name = s;
    }
    public String toString()
    { return "roll :" + r + "name : " + name;
    }
}
```

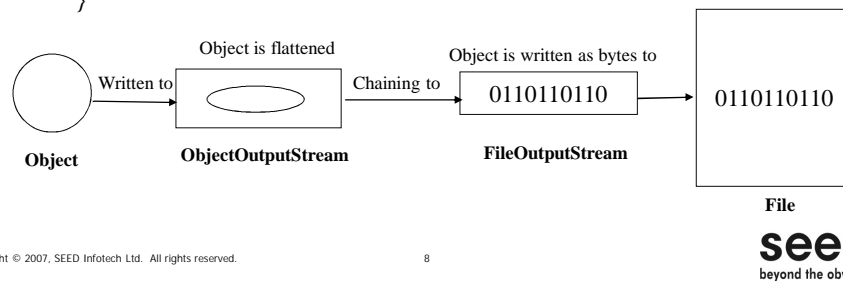
Copyright © 2007, SEED Infotech Ltd. All rights reserved.

7

seed
beyond the obvious

Serializing

```
try{
    Student s1 = new Student(1,"aaa");
    System.out.println(s1);
    FileOutputStream fos = new
        FileOutputStream("stud-details");
    ObjectOutputStream oos = new
        ObjectOutputStream(fos);
    oos.writeObject(s1);
    oos.flush();
    oos.close();
}
```



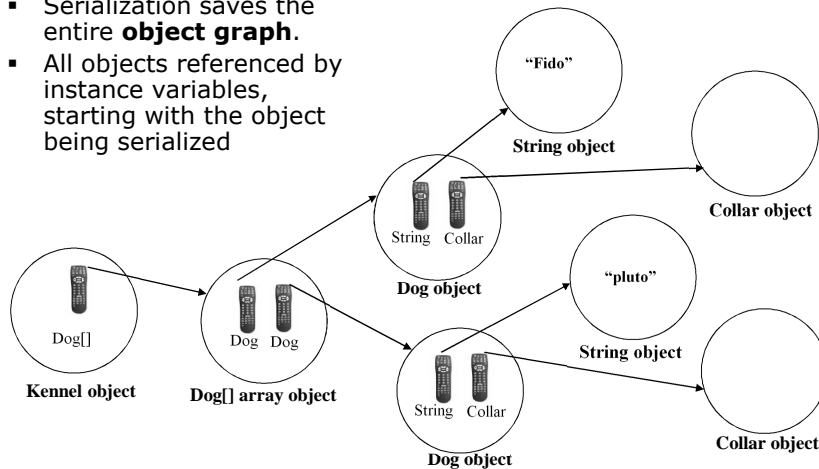
Copyright © 2007, SEED Infotech Ltd. All rights reserved.

8

seed
beyond the obvious

Serializing

- Serialization saves the entire **object graph**.
- All objects referenced by instance variables, starting with the object being serialized



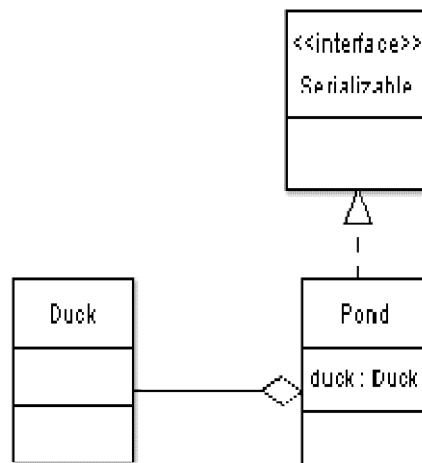
Copyright © 2007, SEED Infotech Ltd. All rights reserved.

9

seed
beyond the obvious

Serialization

- Serialization is all or nothing
- You can't serialize a Pond object if its Duck instance variable refuses to be serialized (by not implementing Serializable)
- If you want an instance variable to be skipped by the serialization process, mark the variable with the **transient** keyword, e.g Network connection, file object etc...



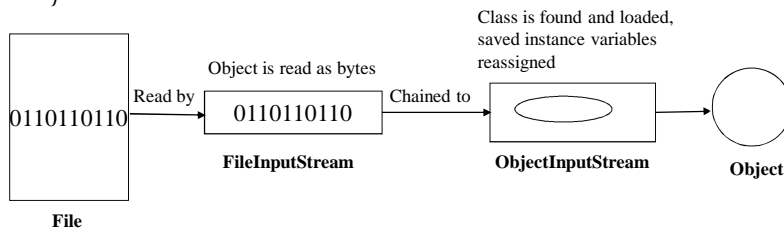
Copyright © 2007, SEED Infotech Ltd. All rights reserved.

10

seed
beyond the obvious

Deserializing

```
Student s2;  
FileInputStream fis = new  
    FileInputStream("stud-details");  
ObjectInputStream ois = new  
    ObjectInputStream(fis);  
s2 = (Student) ois.readObject();  
ois.close();  
System.out.println(s2);  
}
```



Note:- Present a paper for happenings during deserialization?