



Lecture (02)

Serialization

Dr. Ahmed ElShafee

Serialization

Serialization is a process of writing an object to a byte stream.

Providing a way for objects to be written as a stream of bytes and then later recreated from that stream of bytes.

Serialization (cont,..)

ObjectInputStream

The ObjectInputStream class enables you to read Java objects from InputStream's instead of only bytes.

You wrap an InputStream in a ObjectInputStream and then you can read objects from it.

```
ObjectInputStream input = new ObjectInputStream(  
    new FileInputStream("object.data"));  
  
MyClass object = (MyClass) input.readObject();  
//etc.  
  
input.close();
```

Serialization (cont,..)

For this example to work the object you read must be an instance of MyClass, and must have been serialized into the file "object.data" via an ObjectOutputStream.

Serialization (cont,..)

ObjectOutputStream

The ObjectOutputStream class enables you to write Java objects to OutputStream's instead of only bytes.

You wrap an OutputStream in a ObjectOutputStream and then you can write objects to it.

```
ObjectOutputStream output = new ObjectOutputStream(  
    new FileOutputStream("object.data"));  
  
MyClass object = new MyClass();  
  
output.writeObject(object);  
//etc.  
  
output.close();
```

Serialization (cont,..)

This serialized object can now be read via an `ObjectInputStream`.

Serialization (cont,..)

The `Serializable` interface is a marker interface your classes must implement if they are to be serialized / deserialized, like shown the examples with [ObjectInputStream](#) and [ObjectOutputStream](#).

The **Serializable** interface eliminates the drawbacks of sending objects across streams.

Each object to be sent has to implement this interface

Serialization (cont,..)

What is interface, and class implements?

- In its most common form, an interface is a group of related methods with empty bodies.
- A bicycle's behavior, if specified as an interface, might appear as follows:

```
interface Bicycle  
{ void changeCadence(int newValue); // wheel revolutions per  
minute  
void changeGear(int newValue);  
void speedUp(int increment);  
void applyBrakes(int decrement); }
```


Serialization (cont,..)

- To implement this interface, the name of your class would change (to a particular brand of bicycle, for example, such as ACMEBicycle), and you'd use the implements keyword in the class declaration:

```
class ACMEBicycle implements Bicycle  
{ // remainder of this class implemented as before }
```

Thanks,
See you next Week, isA