**Cave of programming 46 – Serialization**

* “Introduction to serialization” below the code

**1st class – WriteObjects.java:**

**import** java.io.FileNotFoundException;

**import** java.io.FileOutputStream;

**import** java.io.IOException;

**import** java.io.ObjectOutputStream;

**public** **class** WriteObjects {

**public** **static** **void** main(String[] args){

System.*out*.println("Writing objects...");

Person mike = **new** Person(543, "Mike");

Person sue = **new** Person(123, "Sue");

System.*out*.println(mike);

System.*out*.println(sue);

**try**(FileOutputStream fs = **new** FileOutputStream("people.bin")){

ObjectOutputStream os = **new** ObjectOutputStream(fs);

os.writeObject(mike);

os.writeObject(sue);

os.close();

} **catch** (FileNotFoundException e){

e.printStackTrace();

} **catch**(IOException e){

e.printStackTrace();

}

}

}

**2nd class – ReadObjects.java:**

**import** java.io.FileInputStream;

**import** java.io.FileNotFoundException;

**import** java.io.IOException;

**import** java.io.ObjectInputStream;

**public** **class** ReadObjects {

**public** **static** **void** main(String [] args){

System.*out*.println("Reading objects...");

**try**(FileInputStream fi = **new** FileInputStream("people.bin")){

ObjectInputStream os = **new** ObjectInputStream(fi);

Person person1 = (Person)os.readObject();

Person person2 = (Person)os.readObject();

os.close();

System.*out*.println(person1);

System.*out*.println(person2);

} **catch** (FileNotFoundException e) {

e.printStackTrace();

} **catch** (IOException e) {

e.printStackTrace();

} **catch** (ClassNotFoundException e) {

e.printStackTrace();

}

}

}

**3rd class – Person.java:**

**import** java.io.Serializable;

**public** **class** Person **implements** Serializable{

**private** **static** **final** **long** *serialVersionUID* = 4801633306273802062L;

**private** **int** id;

**private** String name;

**public** Person(**int** id, String name){

**this**.id = id;

**this**.name=name;

}

**public** String toString() {

**return** "Person [id=" + id + ", name=" + name + "]";

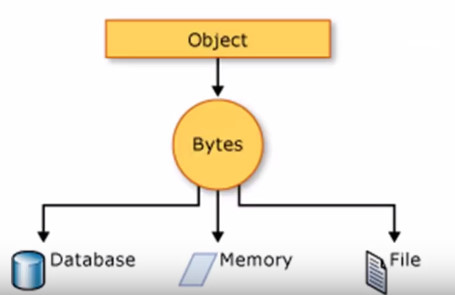
}

}

**Introduction to serialization**

Serialization is the process of **converting an object** into **a stream**.

Deserialization is the process of **reconstructing an object** that has been **serialized**.



**Why serialize?**

1. **Store** data/state of an object
2. **Transmit** data/state of an object
3. **Clone** an object without overriding clone

To store a serialized object in a file, create an ObjectOutputStream with a FileOutputStream.

* FileOutputStream //sequence of bytes -> File

Writes the binary data to a file

* ObjectOutputStream //object -> sequence of bytes

Converts a Java object (or a primitive type) to a sequence of bytes and writes it to an OutputStream (e.g. FileOutputStream, ByteArrayOutputStream)

**Serialize**

new **ObjectOutputStream**(new **FileOutputStream**(“clients.ser”));

Or in simper way:

**try**(**FileOutputStream** fs = **new** **FileOutputStream**("people.bin")){

**ObjectOutputStream** os = **new** **ObjectOutputStream**(fs);

os.close();

}

**Object -> sequence of bytes -> file**