Interfaces{

1. **Interfaces**
2. **Java interface Serialization**
3. **Java interface Iterable**
4. **Java interface Comparable (and Comparator)**

Gruppe: Andreas Lind, Wajid, Ali.
Dato: 27-03-2019

}

Interface{

What is an interface?

An interface is a reference type in Java. It is similar to class. It is a collection of abstract methods.
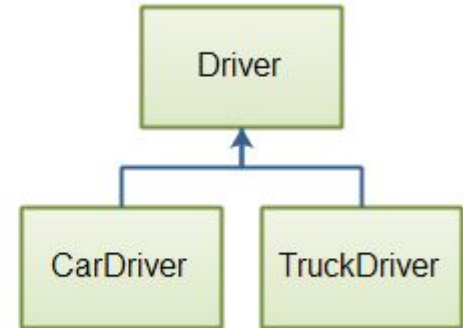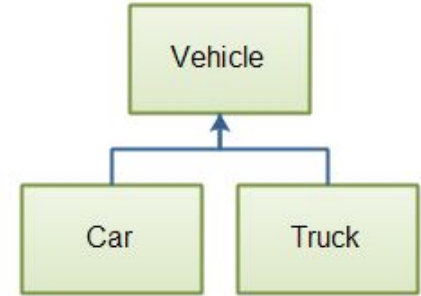
Interface methods does not have a body - the body is provided by the "implement" class

On implementation of an interface, you must override all of its methods

Interface methods are by default abstract and public

Interface attributes are by default public, static and final

An interface cannot contain a constructor

}

Serialization{

Serialization   **De-Serialization**

What is Serialization?

Java provides a mechanism, called "object serialization" where an object can be represented as a sequence of bytes that includes the object's data as well as information about the object's type and the types of data stored in the object.

Why using it?

After a serialized object has been written into a file, it can be read from the file and deserialized that is, the type information and bytes that represent the object and its data can be used to recreate the object in memory.
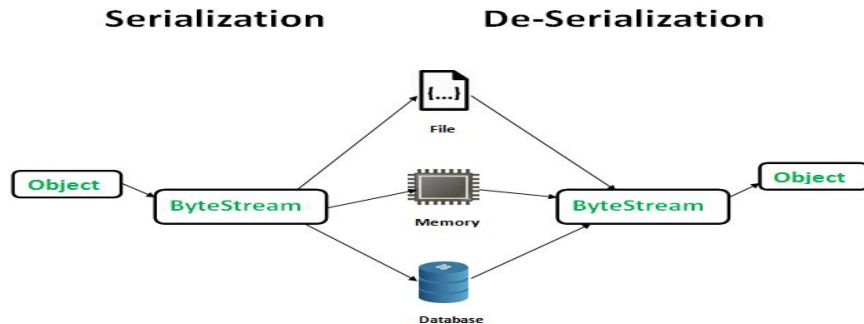
How do you use it?

Classes ObjectInputStream and ObjectOutputStream are high-level streams that contain the methods for serializing and deserializing an object.

Why is it smart?

You can store objects in bytes that includes the objects data and information

}

Iterable{

What is Iterable

A class that implements the Java Iterable interface can be iterated with the **Java for-each loop**. By iterating It means that its internal elements can be iterated.

Why using it?

Basically iterators allow iteration over any Collection

How do you use it?

required to use Java's for-each control statement.

```java
public interface Iterable<T> {
    public Iterator<T> iterator();
}
```

Why is it smart?

If you have a complicated data set, but you *don't care* how it's structured internally, you just want to get all elements one by one, you get it to return an iterator.
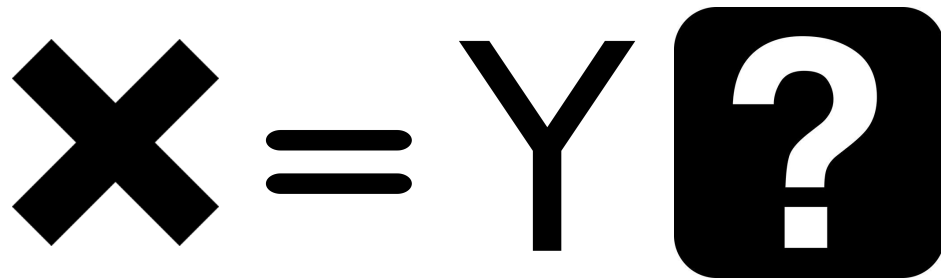
}

Comparable (and comparator){

What is comparable?

Comparable is a build in interface in java which is use to compare one thing to another which is good for sorting purposes.

Why using it?

More complex sorting.

How do you use it?

compareTo() implements comparator();

}