# Serializacja

To zamiana obiektu w strumień danych, dzięki czemu można obiekty zapisywać. Obiekty zserializowane powinno się dać deserializować.

Żeby obiekt był serializowalny w Java'ie MUSI implementować interfejs **Serializable**.

Interfejs ten nie posiada żadnych metod do zaimplementowania. To jest tylko informacja dla JVM, że obiecujemy zapewnić tę serializowalność. Mimo, że obiekty nadklasy może i są dobrze serializowane nie znaczy to, że obiekty klasy dziedziczącej też będą w praktyce serializowane. Dopiero w runtime, a nie compiletime jesteśmy w stanie sprawdzić czy serializacja tak naprawdę działa. Wszystkie elementy klasy są serializowane chyba, że oznaczymy jest **@Transient.** Przed serializacją lub deserializacją należy upewnić się, że klasa bazowa klasy serializowanej ma bezparametrowy konstruktor.

Wymagania, aby obiekt był serializowany:

* musi implementować Serializable (lub jedna z jego superklas musi),
* wszystkie pola składowe muszą być albo:

1. serializowalne ,
2. być typami prostymi,
3. być oznaczone jako transient, (by je zapisać trzeba dostarczyć własny mechanizm serializacji, lub pole takie będzie zignorowane.)

<https://github.com/idzikpro/JavaBasics/blob/master/src/main/java/pl/idzikpro/serialization/SerializationMain.java>

String file = "file.bin";  
Person person=new Person("thomas",22);  
try (ObjectOutputStream objectOutputStream = new ObjectOutputStream(new FileOutputStream(new File(file)))){  
 objectOutputStream.writeObject(person);  
} catch (IOException e) {  
 e.printStackTrace();  
}

## Deserializacja

<https://github.com/idzikpro/JavaBasics/blob/master/src/main/java/pl/idzikpro/serialization/DeserializationMain.java>

String file = "file.bin";  
try (ObjectInputStream objectInputStream=new ObjectInputStream(new FileInputStream(new File(file)))){  
 Object person=(Person)objectInputStream.readObject();  
 System.*out*.println(person);  
} catch (IOException e) {  
 e.printStackTrace();  
} catch (ClassNotFoundException e) {  
 e.printStackTrace();  
}

Cały proces deserializacji to nic innego jak interpreter, a interpretery są niebezpieczne. Serializacja w którejś kolejnej wersji Javy ma zostać usunięta.

**UWAGA:**

Klasa serializowana powinna mieć pole

private static final long *serialVersionUID* = 12312;

które mówi o wersji obiektu klasy. Jeśli nie będzie tego pola, to JVM doda je sama. Wartość tego pola jest sprawdzana w trakcie deserializacji. Ta wartość oznacza przecież wersję klasy.