**Модификатор transient**

[Java Core](https://javarush.ru/quests/QUEST_JAVA_CORE)

[Уровень 10](https://javarush.ru/quests/lectures?quest=QUEST_JAVA_CORE&level=10), Лекция 5

— Привет, Амиго! Хотела тебе порассказывать одно маленькое дополнение к сериализации.

Допустим наш класс содержит ссылку на какой-нибудь **InputStream**, тогда его нельзя сериализовать, ведь так?

— Да. Ты же сама говорила, что потоки сериализовать нельзя. А сериализовать объект, у которого есть несериализуемые данные – тоже нельзя.

— Да. Именно так. Но что, если класс хранит данные, которые не играют значащей роли в его состоянии, но мешают считаться ему сериализуемым классом? Мало ли что класс может у себя хранить ненужного. Возможно, он может выбросить эти данные в любой момент или даже так и делает постоянно.

Для таких случаев разработчики Java придумали специальное слово – **transient**. Его можно написать перед переменной класса и она не будет учитываться при сериализации. Ее состояние не будет ни сохраняться, ни восстанавливаться. Как будто и нет ее вовсе. Как раз для таких ситуаций, как мы только что рассмотрели.

Помнишь кеширование и модификатор **volatile**? **Нет правил без исключений.**

Вот тебе один примерчик такого счастья:

Пример «кота» с невидимой для сериализации переменной — in:

Код

class Cat implements Serializable

{

public String name;

public int age;

public int weight;

transient public InputStream in = System.in;

}