Implement different types of InputStreams using inheritance to achieve code reuse InputStreamBase + read() + init() ObjectInputStream FileInputStream + close() + available() + read() + read() ByteArrayInputStream FilterInputStream #in: InputStream + read() + read() + close() + close() + available() Potential issue DataInputStream BufferedInputStream Lets take a look at FilterInputStream children. BufferedInputStream inherits implementation of read() and available() from the parent. In the same time, + read() + close() DataInputStream needs a different implementation of + close() read() that is be completely the same as in ByteArrayInputStream. But, DataInputStream can't reuse read() of the ByteArrayInputStream because they are not on the same branch of the class hierarchy tree. Inheritance is losing over composition since it doesn't allow to reuse code horizontally in class hierarchy.