Turbold Chuluunbaatar

Sheikh Shamsuddin

Object-Oriented Programming CPSC-24500-004

16 November 2021

Serialization

“Serialization is a mechanism of converting the state of an object into a byte stream. Deserialization is the reverse process where the byte stream is used to recreate the actual Java object in memory. This mechanism is used to persist the object” (GeeksforGeeks). The byte stream created is platform independent. So, the object serialized on one platform can be deserialized on a different platform. To make a Java object serializable we implement the java.io.Serializable interface. The ObjectOutputStream class contain writeObject() method for serializing an Object. The ObjectInputStream class contain readObject() method for deserializing an object. The advantages of serialization are “the serialized stream can be encrypted, authenticated, and compressed, supporting the needs of secure Java computing and can be used as a mechanism for exchanging objects between Java and C++ libraries, using third party vendor libraries within C++” (Ravi). However, on the other hand, “it should not be used with large-sized objects, as it offers significant overhead. Large objects also significantly increase the memory requirements of your application since the object input/output streams cache live references to all objects written to or read from the stream until the stream is closed or reset. Consequently, the garbage collection of these objects can be inordinately delayed” (Ravi).

Works Cited

*Serialization and deserialization in Java with example*. GeeksforGeeks. (2021, October 7). Retrieved November 17, 2021, from https://www.geeksforgeeks.org/serialization-in-java/.

Ravi, M. (n.d.). *What is advantage and disadvantage of serialization*. www.javatpoint.com. Retrieved November 17, 2021, from https://www.javatpoint.com/q/1940/what-is-advantage-and-disadvantage-of-serialization.