Interview Theory Questions

1. **What is a multi-dimensional array in Java?**
   A multi-dimensional array in Java is an array that contains other arrays. It can be thought of as an array of arrays, where each element of the main array is itself an array.

2. **Can a multi-dimensional array have different lengths for its inner arrays?**
   Yes, a multi-dimensional array can have inner arrays of different lengths. This is known as a jagged array.

3. **What are wrapper classes in Java?**
   Wrapper classes are classes that provide a way to use primitive data types as objects. They encapsulate primitive values and provide utility methods for conversion and manipulation.

4. **What is boxing in Java?**
   Boxing is the process of converting a primitive data type into its corresponding wrapper class object. For example, converting an int to an Integer object.

5. **What is autoboxing and unboxing in Java?**
   Autoboxing is the automatic conversion of a primitive data type to its corresponding wrapper class object.
   unboxing is the automatic conversion of a wrapper class object back to its corresponding primitive data type.

6. **Name some commonly used wrapper classes in Java.**
   Integer for int
   Double for double
   Boolean for boolean
   Character for char

7. **What are wrapper conversion utilities in Java?**
   Wrapper conversion utilities are static methods provided by wrapper classes that allow you to create wrapper objects from various data sources, such as strings or primitive values.

8. **How can you create a wrapper object from a string using wrapper conversion utilities?**

You can use the valueOf() method of the wrapper class to create a wrapper object from a string.

9. **What happens if the string passed to valueOf() cannot be parsed into the desired data type?**
   If the string cannot be parsed into the desired data type, a NumberFormatException will be thrown.

10. **Can you create a wrapper object from a primitive value using autoboxing?**
    Yes, autoboxing automatically converts primitive values to their corresponding wrapper objects and vice versa.