The `Grade` Class

The `Grade` class has been designed and only needs to be implemented.

Attributes

The `Grade` class must contain the following attributes:

```
private Double        value;
private String        key;
```

Methods

The `Grade` class must contain the following methods. (Note: The `toString()` method has already been implemented.) It may contain other methods as well.

```
/**
 * Construct a Grade with the given key and value.
 *
 * @param key    The key used to identify this Grade object
 * @param value The value of this Grade
 * @throws       IllegalArgumentException if key is null or empty
 */
public Grade(String key, Double value) throws IllegalArgumentException


/**
 * Compare this Grade object with the specified
 * Grade object for order (required by Grade)
 *
 * @param other  The object to compare to
 * @return       -1/0/1 depending on whether this </==/> other
 */
public int compareTo(Grade other)
```

```java
/**
 * Get the key associated with this Grade object.
 *
 * @return    The key
 */
public String getKey()


/**
 * Get the numeric value associated with this Grade object.
 *
 * @return    The numeric value (or null for missing)
 */
public Double getValue()


/**
 * A factory method for constructing an ArrayList of Grade objects
 * from a comma-delimited String representation. (This method
 * assumes that each individual Grade is represented as required
 * by parseGrade(String)).
 *
 * @param s    The String representation
 */
public static ArrayList<Grade> parseGrades(String s)
    throws IllegalArgumentException


/**
 * A factory method that creates a Grade from a String representation.
 *
 * @param s  The String representation
 */
public static Grade parseGrade(String s) throws IllegalArgumentException


/**
 * A utility method for constructing a String representation of
 * an ArrayList of Grade objects.
 *
 * @param grades    The ArrayList of Grade objects
 */
public static String toString(ArrayList<Grade> grades)
```

```java
/**
 * Create a String representation of this Grade object.
 *
 * @return  The String representation
 */
public String toString()
{
    if (getValue() == null)
        return String.format("%s:%5s",   key, "NA");
    else
        return String.format("%s:%5.1f", key, getValue());
}
```