1. Explain what the RandomAccessFile seek method does.
   * Sets the file-pointer offset, measured from the beginning of this file, to where the next read or write occurs.
2. In ObjectRandomAccessFile, the read and write methods, which are invoked in readObject and writeObject, respectively, are not explicitly qualified by an object name. For example, they are simply used as read(byteArray, 0, objectSize); and write(byteArray, 0, objectSize); To what object do read and write belong?
   * They belong to the java.io.RandomAccessFile object
3. What data source is being “read” in read(byteArray, 0, objectSize);?
   * The data source that is being read is the RandomFileAccess file stream that was created when the constructor for RandomFileAccess was ran which created the stream from the student.dat file.
4. What is the destination to which data is copied in read(byteArray, 0, objectSize);?
   * The data is copied into whatever byteArray that is given in the parameters.
5. Why do the read and write methods invoked by readObject and writeObject in ObjectRandomAccessFile operate on byte arrays, not objects?
6. In the statement read(byteArray, 0, 16), what does the 0 represent?
   * The offset of where the first byte is read within the bytearray
7. In the statement read(byteArray, 0, 16), what does the 16 represent?
   * The number of bytes of data from the input stream that is read into byteArray
8. ObjectRandomAccessFile does not define the read and write methods. How is it able to use them, or put another way, where do they come from?
   * ObjectRandomAccessFile is able to use them because it extends the class RandomFileAccess which is the class that defines the read and write methods, and it gives ObjectRandomAccessFile the ability to use them.
9. What does the writeObject method of ObjectOutputStream do? Say more than it writes objects.
   * It writes the passed object to the ByteArrayOutputStream, which is converted to a byteArray then writes the byteArray to the specified file.
10. What does the readObject method of ObjectInputStream do? Say more than it reads objects.
    * Reads the specified file into a byteArray starting from the specified record number it was given, the byteArray is then read into the ObjectInputStream, and then a StudentObject is created from that ObjectInputStream.
11. Why do we need to derive ObjectRandomAccessFile from RandomAccessFile? Why not just use RandomAccessFile?
12. What effect does marking some of the data members in Student as “transient” have?
13. The Serializble interface does not require classes which implement it to define any methods. So what purpose does it serve?
14. Explain what it means to serialize an object?
15. Given the statement: Map <String, Book> isbnKeyedMap = new HashMap();

String is the class to which the \_\_\_\_\_\_\_\_\_\_\_\_ belongs, and Book is the class to which the \_\_\_\_\_\_\_\_\_\_\_\_ belongs.

1. What does a Map.Entry represent?
2. What method would you use to retrieve the key object from a record in a Map.Entry object?
3. What method would you use to retrieve the value object from a record in a Map.Entry object?
4. What does the entrySet method do?
5. What method would you use to insert a record into a Map?
6. When a program reads a file, what condition causes the EOFException to be thrown?
7. What method would you use to convert a ByteArrayOutputStream object to a byte array?
8. The loadMaps method in Driver.java does explicitly return any Map objects to the method that called it. How does the calling method receive the Map objects that loadMaps processes?
9. Write an “if” statement to test the condition empObj is an instance of the Employee class.
10. Why can’t we store primitive data types – char, int, long, double, etc. – in Java collection strucatures like ArrayLists and Maps?