**Evaluation of sources:**

**Ivor Horton's Beginning Java, Java 7 Edition:**

**Source 1**

This book is a complete guide to beginning Java. It contains theory, syntax, and how it differs from other languages. It also covers in detail most aspects of logic. I used this book whenever I got a strange error as well as the whole theory behind Java's entities and objects, as Java handles code in a different way to most other languages. It does however lack in specifics which I needed, since it wasn't based around any form of game design. Because of this I didn't use the book that much after I had the basics of the game to set up.

It often matched with Source 3 in how implement certain things yet wrote about them in a more clear way, particularly about key listeners. It was written by a well known author, who also writes books for other languages. Because of this, I believe the book is a reliable source, yet not overly useful due to the niche nature of my project.

**Java 2D Game Tutorial:**

**Source 2**

This website is a tutorial for how to make simple games in Java, it has a step by step description of what was needed to be done. I used this source as the starting block for my entire artefact, it explained in a clear manner how to set up the background, player movement, and entities. It was incredibly useful as it offered step by step guides with code examples.

The source however was old, it was last update on 2010/10/07, so there have been a lot of updates to the code base and practices since then. This was not inherently a problem as everything still worked, it it meant that it didn't necessarily use the most efficient process, as well as formatted code in a weird way. The tutorial was also written by someone not affiliated with Oracle, the company that make Java, at all, nor were there any references to their experience with the language, so a large portion of the code may not be reliable. However, it when I implemented their sections of code, it was mainly bug free, and therefore far more useful and reliable than other sources.

**Official Java Docs, Oracle:**

**Source 3**

This is a website which covers in detail every part of the Java code base. It gives a lot of detail and list commands for every aspect of programming with Java. I used it for a variety of reasons: from sounds to how to rotate the player's sprite. While in depth, it is incredibly confusing and far above my level. Many times I only used it for the specific commands or for the basic idea of what is happening. The source is updated regularly, but often in a very text heavy format, so it's readability is very low. The guides that are included in the documents are also often not helpful.

It also contradicted Source 2 about how to set up a JFrame, the window in which the game displayed. I could not get this version to work though, so I decided to keep with my earlier version. Despite all of this, it is an official page so should be very reliable, albeit confusing. Because of this, it wasn't very useful in progressing my game.

**Inside My Laptop:**

**Source 4**

This source contains lots of information about how to fix a recover data from a broken laptop. I used this website when my laptop died and I needed to recover all of the data. It was useful as it offered step by step solutions with pictures, which helped show me what to do. It significantly increased my practical knowledge about computers as well.

However, I was unable to find who wrote the website, so it is not reliable despite the solution offered working.

**Java game programming; Playing sound:**

**Source 5**

“JavaSE, via Java Sound API (in packages javax.sound), supports two types of audio:

* Sampled Audio: Sampled audio is represented as a sequence of time-sampled data of the amplitude of sound wave. It is supported in package javax.sound.sampled. The supported file formats are: "wav", "au" and "aiff". The samples can be either 8-bit or 16-bit, with sampling rate from 8 kHz to 48 kHz.
* Musical Instrument Digital Interface (MIDI): MIDI music is synthesized from musical notes and special sound effects, instead of time-sampled, like a recipe for creating musical sound. MIDI is supported in package javax.sound.midi.”

This website explains in depth how to use both types of audio. It offers explanations and commented code examples.

The sampled audio was the type I used in my game. When I tried to implement the code however, I could not get it to work at all. It explains how to use clips in a similar way that Source 3 does, so it is likely to be corrected as it uses the same framework as the official Java Docs. Because of this, it is likely to be reliable. I did however find it confusing, and just directly implementing some of the code gave me errors without even modifying it.

Source 8 explains the same problem only in a much clearer way, as well as I got sound in the game to work using Source 8, so I believe it does not have as stronger claim to reliability as that source does.

**Java2D: Have Fun With Affine Transform:**

**Source 6**

“Affine transformations are a concept based in euclidean algebra, and help us define (through the use of matrices) ways to modify coordinate spaces so that 'parallelism' and 'perpendicularity' are preserved. Affine transformations can help provide translations (shifting on the x and y axes), rotations, scaling (zooming), and shearing (another, more mutative form of scaling).”

This source explains the types of transformations you can do with the AffineTransform class, the theory behind them, and demonstrates with explanations and commented code how to implement them.

I used this source to help me implement the rotation of all of my sprites in game, both the player and the enemies. Despite not used in the actual game, this was incredibly useful for debugging the player's movement as well as that of the enemies. This source simplifies what is shown in Source 3 and explains it in a much more clear manner. Because it matches source 3 as well as the source works, I would say that this source is very reliable and useful.

**Edu4Java:**

**Source 7**

This source is a website which explains how to code using Java.

I visited the website for help with sound implementation, however I decided not to follow it's tutorial. It was not as in depth as Source 3 or Source 5, nor did it explain it as clearly as in Source 8. The source also had information that wasn't useful to me at that point in time, such as how to create some test sounds. On top of this, I identified bad practises in the example code. For example, it uses the sleep method when a sound is playing. This freezes the program until the sound has stopped, so would be completely useless in a game.

The sources also isn't very reliable, I was unable to identify who wrote it or when it was written, and it contradicted with the Source 3, the official documents, at certain points. This source was not useful at all.

**Codecall Forum Post: Playing Simple (sampled) Audio in Java.**

**Source 8**

This source is a forum post describing how to implement sampled sound in an easy way. Brief and to the point, it explains in a clear and concise manner.

It was incredibly easy to follow as it had short sections of code interjected with explanations because of this it was extremely useful in helping me set up the entire sound system which underpins my project. It is reliable as it references the Source 3, using extracts from there which it then increased readability. This shows that the post, despite being written by an unknown poster is reliable. The source also is very similar to that of Source 5, so it matches up there as well. The poster himself has written over 150 posts, so is prolific and generally regarded as helpful.

**StackOverflow Selection of Forum Posts:**

**Source 9**

StackOverflow is a forum completely based around answering questions about coding. Similar to WikiAnwers, only a lot more reputable, it covers everything you need to know about anything related to computers. Questions can be modified by anyone, deleted, and marked as duplicates to increase helpfulness around the site and readability.

9.1: Using a string as a variable clause in an if statement:

Clear and concise answer. Everyone else agreed with the top voted answer, who has a lot of reputation on the site. Likely to be reliable.

9.2: How can I play sound in Java?

Answer was not helpful, nor did it match any of my other sources about playing sound, such as Source 3 or Source 8. Despite the user who posted it having massive amounts of presence of the site, this was not a reliable source.

9.3: What is a Null Pointer Exception, and how do I fix it?

A long and useful answer, based around the theory of a null pointer exception, and how to track one down and fix it. I mainly had problems with this error when implementing images for the first time. This was incredibly useful as it helped me fix it. Answered by a well known user, this is a reliable post.

9.4: Java: How to use Keyboard in program

Not incredibly useful, just a reference to the official documentation with a little bit of explaining. However, it is reliable.

**Interview with Jason Mashinchi.**

**Source 10**

Jason is a developer at Cambridge Software. He is extremely knowledgeable about a wide variety of coding practises and languages.

The interview consisted of the pro's and con's of the use of projectiles for bullets or vectors. He explained to me the advantages and disadvantages of using projectiles gave me some advice on how to proceed. It was explained that projectiles were bad for processor and memory usage as you have to calculate it's moving positions for each bullet, and also that they were considerably harder to code. He suggested vectors as that is just a straight line, which happens instantly to see if it collides with a player. This would be easier to code and less intensive. However, pausing the program until the bullet hits could cause a hitch, a freeze in the game for over a second, if the bullet has a way to travel.

Due to the nature of his job, I would say he is reliable.

**Audacity Tutorial 2: Audio Editing:**

**Source 11**

This video demonstrates all of the basic features of how to use Audacity, the sound editing program which I used to create the sounds in my game.

It explains everything in a clear and slow manner, so is useful to cover the basics. It doesn't teach you about sound manipulation and audio effects though, so it's use is limited. While it works, it is quite significantly out of date; the tutorial is for an older version of Audacity, as well as the fact that it is for OSX, which is different operating system as I am using Windows 7. Because of this, it is not very reliable.