Explain the differences between Java and Javascript

- Compile-time errors will not show in javascript, and will only show on runtime, therefore better code completion.
- Languages that are compiled are normally faster.
- JS is a bit more flexible in terms of syntax as compared to java which is very strict.
- JS doesn't have typecheck.
- JS is focused around functions where java is focused on objects

- Difference in object creation:
- JavaScript: Var a = {} OR Var b = new Person()
- Java: Object objectName = new Object();

Function Closures and the JavaScript Module Pattern

A closure is a special kind of object that combines two things:

- A function.
- The environment in which that function was created. The environment consists of any local variables that were in-scope at the time that the closure was created.
-

JavaScript does not, like Java, provide a native way of providing private methods. But it is possible to emulate this using closures.

Example:
See **ReusableModules.js**

JavaScript Prototyping

Prototypes in JavaScript are kind of like instances of objects, and with each prototype you can add/remove/change elements which only affects the given prototype, it does not affect other prototypes of the same object, nor the object itself.

Example:
See **Objects.js**

User defined Callback Functions

Functions that takes a callback (another function) as an argument, are known as callback functions.

Example:
See **Callbacks.js**

Explain generally about node.js and NPM.

Node.js

“Node.js is a platform built on Chrome's JavaScript runtime for easily building fast, scalable network applications.
It uses an event-driven, non-blocking I/O model that makes it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices
Keywords:

- Based on Googles V8 Engine.
- Event Driven.
- Highly targeted against async. Programming.
- Non-Blocking I/O.
- Easily Scalable”.

Source: http://js2016.azurewebsites.net/node1/NodeIntro.html#2

NPM

NPM is a tool for version control and sharing code between developers. A developer can use the tool to share packages of code with others, so they can implement that into any project they’re working on.

Provide examples of user defined reusable modules implemented in Node.js