Core Java

1. Difference between java6/java7/java8?

Java 5.0 not 1.5 to The number was changed to "better reflect the level of maturity, stability, scalability and security of the J2SE.

* Generic : provide compile time (static ) type safty for collection and eliminate the need for most type cast (type conversion)
* Metadata: (Annotation) allow language construct like classes and method to be taged with some additional data which can then be processed by meta-data aware utilities.
* Autoboxing /unboxing : automatic conversion between primitive types and primitive Wrapper classes.
* Enumaration : the enum keyword creates typesafe, ordered list of values(Day.MONDAY,DAY.TUESDAY etc.), previously achived by non type safe constanat and manualy constructed classes.
* Varargs: last parameter of method can be declared as type name followed by three dots(eg. Int add(int … numbers) )
* Enhanced for-each loop : extended for each loop syntax for any Iterable.
* Improved symantics of execustion of multi-threded java program. New java memory model.
* Static import.
* Atomatic stub generation for RMI objects
* Swing : new skinnable look and feel callied synth.
* The concurancy utilities in package java.util.concurent
* Scanner class for parsing data from various input stream and buffers.
* Improved startup time and memory footprint. Sharing of read-only data between multiple running JVMs. Remote monitoring and management. A new JVM profiling API. Programmatic generation of stack traces. Support for XML 1.1 with Namespaces, XML Schema, SAX 2.0.2, DOM Level 3, and XSLT with a fast XSLTC compiler. Unicode 4.0 support.

Java 6

* Scripting language support: generic api for tight integration with scripting language and script in Mozilla javascript rohino integration .
* Performance improvement for core platform.
* Improved web service support thrue JAX-WS
* Jdbc 4.0 support
* Java compiler api : select and invocke java compiler programmatically.
* Upgrade jaxb-to 2.0 including integration of stAX parser
* SUPPort for plugbale annotation.
* Pre-installed relational data base java derby.
* Many gui improvement.
* Jvm improvement: synchronization and compiler performance optimization. New algorithem and upgrades to garbage collection algorthem.and application start up performance.

Java 7:

* Jvm suppor t for dyanamic language
* String in switch
* Automatic resource management in try statement.
* Improved type reference for generic instance creation aka the diamond operator.
* Simplified varg method declaration .
* Allowing underscore in numeric litrals.
* Catching multiple exception types and rethrowing exception with improved type checking.
* Concurancey utilities.
* New file I/o libraray adding support for multiple file system, file metadata and symbolic link new package java.nio.file, java.nio.file.attribute and java.nio.file.spi

Java 8

* Language level support for lymbda expression and default method.
* Project Nashron , a javascript runtime which allow developer to embed javascript codewithin application .
* Annotation on java types.
* Unsigned integer arthmatic
* Repeating annotaition
* Date and Time api
* Statically linked jni librararies.
* Remove permanent generation

Java 9

* Java platform module system.
* The java shell
* Ahed of time compilation
* Xml catalogs
* More concurancy update , reactive stream, Flow class.
* Variable handeles.
* jlink

Abstraction with real scenario?

In java abstraction concept is base of all other oops concept like Plolymorphism, Inheritance, Encapsulation.

Abstraction means hiding /removing complexity with respective to context.

Also Abstraction is modeling real life entity in programing word,

We often reuse abstractions when attempting to model a new concept.

Producing an abstraction of a system to be built, known as a model, is in some senses second nature to us, and yet paradoxically is one of the hardest things that software developers have to do in the life cycle of an information systems project. It’s also one of the most important.