# **Metadata in Java**

# Metadata is data about your data - let’s say you had a DB table full of entries. The rows or tuples are the data, and the columns would be the metadata; they define what each value is for any given row.

# **Annotation in Java**

# Annotations are code snippets you put in at the java class or method level that further define data about the given code, but without changing any of the actual coded logic. If you annotate a method, the execution of that method does not change, but you have specified extra data about the method other java apps or the jvm might use in different situations.

# Annotation are metadata that we able provide in our java programs in order to make available data up to .java, .class and runtime of our applications.

# @Override is a very common annotation, you would put that at the method level to indicate you are either replacing or extending a method that was already defined.

# **XML based Technologies** **Annotation Based Technologies**

# Up to JDK 1.4 version JDK 5.0 version onward

# Jdbc 3.X Jdbc 4.0

# Servelet 2.5 Servelet 3.X

# Struts 1.X Struts 2.X

# JSF 1.X JSF 2.X

# Hibernate 3.2.4 Hibernate 3.2.5

# Spring 2.X Spring 3.X

# Annotations are executed by predefined annotation tool called APT (Annotation processing tool)

# **Annotation Syntax type**

# **Declaration syntax**

# @interface Annotation\_name

# {

# Members (data-type member\_Name () [default value]}; default value is optional.

# }

# **Utilization syntax**

# We can use annotation for Variables, methods, class and almost all programing elements we can utilize annotations.

# @Annotation\_Name(memeber1=value1, members2=value2….)

# Programming elements(Above)

# **On the basis of annotations members, Annotation divided into three types:**

# **Marker Annotation**

# It is like Marker Interface. Any interface without any abstract methods.

# Marker annotation is the annotation without any member. @Override or @WebListners

# **Single Valued Annotations**

# It contains only one member. @SupressWarnnings(“Uncked”) i.e Value = “Unchecked”

# **Multi value annotation**

# It contains more than one members.

# @WebServlet(name=“ms”, urlPatterns={“/abc”,”xyz”}

# **Classifications of Annotation in Java**

# **Standard Annotation**

# Java Provided annotation. Java API contains this annotation.

# Common default super type is java.lang.annotation.Annotation

# All the annotations are Interfaces

# **General-Purpose Annotation**

# Use for normal, common and frequent purposes.

# Avail in java.lang Package

# @Override

# @SupressWarning

# @Depricated

# @FunctionalInterface (Only Available in Java 1.8)

# **Meta-Annotation**

# Annotation about annotations. They are used to give nature to our or user define annotations.

# Available in java.lang.annotation

# @Documented

# @Inherited

# @Target

# @Retention

# **Custom Annotation**

# Programmer defined annotations.