Bean Types

T **@Stateless**(name="")
T **@Stateful**(name="")
M **@Init**(init = "")
M **@Remove**(retainIfException=false)
T **@MessageDriven**(name="", ActivationConfigProperty[] activationConfig={})
@ActivationConfigProperty(propertyName="", propertyValue="")
T **@Entity**(name="", access=[*PROPERTY*,FIELD], entityType=[*CMP*,BMP], version=3)
T **@Remote**
T **@Local**
M **@BusinessMethod**

Transaction and Security

T **@TransactionManagement**(value=[*CONTAINER*,BEAN])
TM **@TransactionAttribute**(value=[MANDATORY, *REQUIRED*, REQUIRES_NEW, SUPPORTS, NOT_SUPPORTED, NEVER])
TM **@MethodPermissions**(String[] roles)
TM @PermitAll
M **@DenyAll**
T **@RunAs**(String rolaname)
T **@DeclareRoles**(value={})
TM **@RolesAllowed**(value={})

Callbacks

T **@CallBackListener**(String classname)
M **@PostConstruct**
M **@PreDestroy**
M **@PostActivate**
M **@PrePassivate**
Te **@EntityListener**(String classname)
Me **@PrePersist**
Me **@PostPersist**
Me **@PreRemove**
Me **@PostRemove**
Me **@PreUpdate**
Me **@PostLoad**

Resources

TMF **@Resource**(name="", type=Object.class, AuthenticationType=[*CONTAINER*,Application], shareable=true, resourceType="", mappedName="")
T **@Resources**(Resource[] value)
TMF **@EJB**(name="", businessInterface=Object.class, mappedName="", description="")
T **@EJBs**{EJB[] value)

Interceptors, Exceptions

T **@Interceptors**(Class[] value)
M **@AroundInvoke**
TM **@ExcludeDefaultInterceptors**
M **@ExcludeClassInterceptors**
T **@ApplicationException(rollback=false)**

EJB 2.1 and Ealier Client View

Ts **@RemoteHome**(Class value)
Ts **@LocalHome**(Class value)

Queries

TM **@FlushMode**(value=[COMMIT, AUTO, NEVER])
PTM **@NamedQuery**(name="", queryString, resultType, ejbInterfaceType[*LOCAL*, REMOTE])
PTM **@NamedQueries**(NamedQuery[] value)

ResultSetMapping

TM **@SqlResultSetMapping**(String name, EntityResult[] entities={}, ColumnResult[] columns={})
@EntityResult(String name, FieldResult[] fields = {})
@FieldResult(String name, column)
@ColumnResult(String name)

Object-Relational-Mapping

T **@Table**(name="", catalog="", schema="", UniqueConstraint{}={}, specified=true)
T **@SecondaryTable**(name="", catalog="", schema="", JoinColumn[] join={}, UniqueConstraint[] uniqueConstraints={})
T **@SecondaryTables**(SecondaryTable[] value)
@UniqueConstraint(primary=false, String[] columnNames)
TMF **@Column**(name="", primaryKey=false, unique=false, nullable=true, insertable=true, updatable=true, columnDefinition="", secondaryTable="", length=255, precision=0, scale=0, specified=true)
TMF **@JoinColumn**(name="", referencedColumn="", primaryKey="", unique=false, nullable=true, insertable=true, updatable=true, columnDefinition="", secondaryTable="")
TMF **@JoinColumns**(JoinColumn[])
MF **@GeneratedValue**(generationType=*[NONE*, TABLE, SEQUENCE, IDENTITY, AUTO), generator="")
MF **@Id**
MF **@EmbeddedId**(AttributeOverride[]={})
@AttributeOverride(name, Column[]={})
@AssociationOverride(name, JoinColum={})
T **@IdClass**
MF **@Transient**
MF **@Version**
MF **@Lob**(fetch=[*LAZY*, EAGER], type=[*BLOB*,CLOB])
MF **@Serialized**(fetch=[LAZY, *EAGER*])

Embedded Values

T **@Embeddable**
MF **@Embedded**(AttributeOverride[] override={})

Relations

MF **@ManyToOne**(targetEntity=void.class, cascade={ALL,MERGE, PERSIST,REFRESH,REMOVE}, fetch=[LAZY, *EAGER*], optional=true)
MF **@OneToOne**(targetEntity=void.class, cascade={...}, fetch=[...],optional=true, mappedBy="", usePKasFK=false)
MF **@OneToMany**(targetEntity=void.class , cascade={...}, fetch=[...], mappedBy="")
MF **@AssociationTable**(table=@Table(specified=false), joinColumns={}, inverseJoinColumns={})
MF **@ManyToMany**(targetEntity=void.class, cascade={...}, fetch=[...], mappedBy="")

Inheritance

T **@Inheritance**(strategy=[*SINGLE_TABLE*, TABLE_PER_CLASS, JOINED], AttributeOverride[] inheritedAttributes={})
T **@DiscriminatorColumn**(name="DTYPE",discriminator Type=[*STRING*,CHAR,INTEGER],columnDefinition="", length=31)
T **@DiscriminatorValue**(String value)
T **@InheritanceJoinColumn**(name="", referencedColumnNames="", columnDefinition="")
T **@InheritanceJoinColumn**(InheritanceJoinColumn[] columns)
T **@DiscriminatorColumn**(name="", nullable=false, columnDefinition="", length=10)
T **@MappedSuperclass**

Generators

MF **@GeneratedValue**(name="", strategy=[*AUTO*,IDENTITY,SEQUENCE,TABLE], pkColumnName="", valueColumnName="")
PTMF **@SequenceGenerator**(String name, sequenceName="", initialValue=0, allocationSize=50)
PTMF **@TableGenerator**(String name, table="", catalog="", schema="", pkColumnName="", valueColumnName="", pkColumnValue="", initialValue=*0* allocationSize=*50,* uniqueConstraints={})

Legend

PMTF Annotation applies to Package, Method, Type, Field
Me Annotation applies to Entity Bean Methods
Ts Annotation applies to Session Bean
VALUE Default Value

EJB 3.0 Annotations Cheat Sheet

©2005,2006 Philipp Meier – www.fnogol.de
Version 1.2.2, based on Java EE 5 API Doc
Report errors to meier@fnogol.de

This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 2.0 Germany License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/2.0/de/ or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.