Apache Isis

Annotations (org.apache.isis.applib.annotation)

Annotation	Description	
@DomainService *type*	Indicates a class is a domain service singleton that contributes *Actions* to the application's menu and/or to the REST backend.	
@DomainServiceLayout *type*	Domain service layout customization. When present, overridden by menubars.layout.xml	
@DomainObject *type*	Indicates a class that has an identity and is bookmarkable, it is either a viewmodel or an entity. Alternatively implement interfaces ViewModel, ViewModel.CloneableViaSerialization or ViewModel.CloneableViaMemento.	
@DomainObjectLayout *type*	Domain object layout customization. When present, overriden by Xxx.layout[.layoutName].xml > Xxx.layout.xml > Xxx.layout.fallback.xml.	
@Action *method, mixin*	Indicates that a method contributes an *Action*.	@Action .. placeOrder(X x, Y y, Z z) boolean **hide**PlaceOrder() String **disable**PlaceOrder() boolean **hide**{0|1|2}PlaceOrder(X|Y|Z); String **disable**{0|1|2}PlaceOrder(X|Y|Z) String **validate**{0|1|2}PlaceOrder(X|Y|Z) String **validate**PlaceOrder(X x, Y y, Z z) Coll..<X|Y|Z> **autoComplete**{0|1|2}PlaceOrder(X|Y|Z, @MinLength(3) String search) 2 args Collection<X|Y|Z> **choices**{0|1|2}PlaceOrder(X|Y|Z) X|Y|Z **default**{0|1|2}PlaceOrder() String **named**PlaceOrder() String **described**PlaceOrder() *right*: member-support methods
@ActionLayout *field, getter, mixin*	*Action* layout customization. When present, overridden by Xxx.layout[.layoutName].xml > Xxx.layout.xml > Xxx.layout.fallback.xml.	
@Property *field, getter, mixin* Typically used with @lombok.Getter @lombok.Setter	Indicates that a field or method contributes a *Property*. *right*: member-support methods	@Property .. email**, get**Email(), **set**Email(); boolean **hide**Email(); String **disable**Email(); String **validate**Email(X); Collection<..> **autoComplete**Email(), Collection<..> **choices**Email(); String **named**Email() String **described**Email()
@PropertyLayout *field, getter, mixin*	*Property* layout customization. When present, overriden by Xxx.layout[.layoutName].xml > Xxx.layout.xml > Xxx.layout.fallback.xml.	
@Collection *field, getter, mixin* Typically used with @lombok.Getter @lombok.Setter	Indicates that a field or method contributes a *Collection*, meaning a non-scalar property. *right*: member-support methods	@Collection .. orders, List<Order> **get**Orders() boolean **hide**Orders() String **disable**Orders() String **named**Orders() String **described**Orders()
@CollectionLayout *field, getter, mixin*	*Collection* layout customization. Overriden by Xxx.layout[.layoutName].xml > Xxx.layout.xml > Xxx.layout.fallback.xml.	
@Parameter @MinLength *parameter*	Action parameter constraint and behavior customization.	
@ParameterLayout *parameter*	Action parameter layout customization.	
@Domain.Include *field, method*	Indicates that a field or method must contribute to the metamodel.	
@Domain.Exclude @Programmatic *field, method, type*	Indicates that a field or method must **not** contribute to the metamodel.	

@HomePage *type*	Indicates which viewmodel should be used to render as homepage.	
@Title *field, method*	Indicates which property or properties make up the object title.	
@MemberSupport *method*	Indicates that a method supports either an *Action* or a *Property* or a *Collection*.	

Services (most common)

Service	Description
RepositoryService	access to persistence layer
MessageService	UI notifications
FactoryService	object construction
ClockService	provides virtual clock
WrapperFactory	enforce domain rules on domain objects
EventBusService	emit custom events
BookmarkService	bookmark = object identity
InteractionService	*Action* execution and *Property* modification
TransactionService	request transactions
MetaModelService	export domain model

Object Support Methods

boolean hidden()
String disabled()

Layout Support Methods

String title()
String iconName()
String cssClass()
String layout()

Object Lifecycle Callbacks

void created()

Entity Lifecycle Callbacks

void loaded()
void saved()
void saving()
void persisted()
void persisting()
void deleting()
void removing()
void updated()
void updating()

Error & Exception Handling

RecoverableException, NonRecoverableExeption

v2.0.0-M7

Value Types

Number Types	byte, Byte, short, Short, int, Integer, long, Long, float, Float, double, Double, BigInterger, BigDecimal
Boolean Types	boolean, Boolean
Text Types	char, Character, String, Password
Markup Types	Markup, AsciiDoc, Markdown
Temporal Types	java.util.Date, java.sql.{Date|Timestamp} java.time.{LocalDate|LocalDateTime} Joda Time (deprecated)
Enum Types	any
Collection Types	Collection<T>, List<T>, Set<T>, Can<T>
Additional Types	UUID, LocalResource, Url, BufferedImage, Blob, Clob, LocalResourcePath

New Mixin Programming Style

```java
@Action
@RequiredArgsConstructor
public class Customer_placeOrder {

    private final Customer target;

    @Value @Accessors(fluent = true)
    public static class Parameters {
        Product product;
        int quantity;
    }

    public Customer act(
        @Parameter Product product,
        @Parameter int quantity) {
        // ...
        return target;
    }

    // supporting methods (no action name reference required)
    public boolean hide() { ... }
    public String disable() { ... }
    public String validate(Parameters params) { ... }

    // parameter supporting methods (exemplified on first parameter)
    public boolean hideProduct(Parameters params) { ... }
    public String disableProduct(Parameters params) { ... }
    public String validateProduct(Parameters params) { ... }
    public Collection<Product> choicesProduct(Parameters params){ ... }
    // note: additional search parameter required
    public Collection<Product> autoCompleteProduct(
        Parameters params, String search) { ... }
    public Product defaultProduct(Parameters params) { ... }

    // parameter supporting methods (exemplified on second parameter)
    public boolean hideQuantity(Parameters params) { ... }
    // ...
}
```

Spring Framework (5.x)

Spring Annotations and Standard API

@SpringBootApplication @SpringBootTest @Configuration @Import @ComponentScan	Bootstrapping
@Component @Service @Repository	Indicates that a class is a "component". Such classes are considered as candidates for auto-detection when using annotation-based configuration and classpath scanning.
@EventListener	Marks a method as a listener for application events. **Domain Events** .. subscribe using Spring's @EventListener @DomainObject(xxxLifecycleEvent=….) @DomainObjectLayout(xxxUiEvent=….) @Action(domainEvent=….) @Property(domainEvent=…) @Collection(domainEvent=…)
@org.springframework.lang.Nullable	declares that annotated elements can be **null**
@javax.inject.Inject @Autowired	Marks a constructor, field, setter method, or config method as to be autowired by Spring's dependency injection facilities.
@PostConstruct	Marks a method to be executed after dependency injection is done to perform any initialization.
@PreDestroy	Marks a method as a listener for notification that the instance is in the process of being removed by the container .

Entity and Viewmodel Annotations

JPA Entities (javax.persistence)

@Entity @Table @NamedQueries @Id @Version @Column @Transient

JDO Entities (javax.jdo.annotation)

@PersistenceCapable @DatastoreIdentity @Inheritance @Discriminator @Version @Queries @Uniques @Indices @NotPersistent @Column @PrimaryKey @Join @Element

JAXB View Models (javax.xml.bind.annotation)

@XmRootElement; @XmlType @XmlAccessorType; @XmlTransient referenced entities: @XmlJavaTypeAdapter(PersistentEntityAdapter.class)