

sPearCat-Ready Subclass Template

```
//
// ${entity.classNameWithoutPackage}.java
//
// copyright ${copyrightYear} ${copyrightBy}

#if ($entity.packageName)
package $entity.packageName;
#end

import com.webobjects.eocontrol.EOEditingContext;
import com.webobjects.foundation.NSArray;

/**
 * Java Class ${entity.classNameWithOptionalPackage}.java generated for entity ${entity.name}
 * <p>
 * copyright ${copyrightYear} ${copyrightBy}
 * </p>
 */
public class ${entity.classNameWithoutPackage} extends ${entity.prefixClassNameWithOptionalPackage} {
    // private static org.apache.log4j.Logger logger = org.apache.log4j.Logger.getLogger(${entity.
    classNameWithoutPackage}.class);

    /**
     * Sole constructor for entity ${entity.name}
     */
    public ${entity.classNameWithoutPackage}() {
        super();
    }

    /**
     * Returns a list of all objects of entity "${entity.name}" in the shared editing context.
     *
     * @return list of ${entity.name}
     */
    @SuppressWarnings("unchecked")
    public static NSArray<${entity.classNameWithoutPackage}> all${entity.classNameWithoutPackage}Objects() {
        return (NSArray<${entity.classNameWithoutPackage}>)_all${entity.classNameWithoutPackage}Objects();
    }

    /**
     * Returns a list of all objects of entity "${entity.name}" by fetching with the "FetchAll"
    fetchSpecification
     *
     * @param context
     *         editing context to fetch in.
     * @return list of ${entity.name}
     */
    @SuppressWarnings("unchecked")
    public static NSArray<${entity.classNameWithoutPackage}> all${entity.classNameWithoutPackage}Objects
    (EOEditingContext context) {
        return (NSArray<${entity.classNameWithoutPackage}>)_${entity.prefixClassNameWithOptionalPackage}.
        objectsForFetchAll(context);
    }

    /**
     * (non-Javadoc)
     *
     * @see com.sPearWay.catalogAccess.NamedObject#awakeFromInsertion(com.webobjects.eocontrol.
    EOEditingContext)
     */
    @Override
    public void awakeFromInsertion(EOEditingContext anEditingContext) {
        super.awakeFromInsertion(anEditingContext);
        // Set default value for the object after its creation
    }
}
```

```
/*
 * (non-Javadoc)
 *
 * @see com.sPearWay.catalogAccess.NamedObject#awakeFromFetch(com.webobjects.eocontrol.EOEditingContext)
 */
@Override
public void awakeFromFetch(EOEditingContext anEditingContext) {
    super.awakeFromFetch(anEditingContext);
    // Set default value for the object after its fetch
}
}
```