

AuthorityInheritTriggerProvider

•
•
•

()

AuthorityInheritTriggerProvider

```
package com.finebi.common.api.provider;

import com.fr.stable.fun.mark.Mutable;

/**
 * @author gaattc
 * @version 6.0
 * Created by gaattc on 2022/8/30
 */
public interface AuthorityInheritTriggerProvider extends Mutable {

    String XML_TAG = "AuthorityInheritTriggerProvider";

    /**
     * analysis
     */
    boolean originTableInheritTriggerEnable();

}
```

AbstractAuthorityInheritTriggerProvider

```
package com.finebi.fun.impl;

import com.finebi.common.api.provider.AuthorityInheritTriggerProvider;
import com.finebi.constant.Constants;
import com.fr.stable.fun.mark.API;

/**
 * @author gaattc
 * @version 6.0
 * Created by gaattc on 2022/8/30
 */
@API(level = Constants.Extension.NOT_STACKABLE_API_LEVEL)
public abstract class AbstractAuthorityInheritTriggerProvider implements AuthorityInheritTriggerProvider {

    @Override
    public int currentAPILevel() {
        return Constants.Extension.NOT_STACKABLE_API_LEVEL;
    }

    @Override
    public String mark4Provider() {
        return getClass().getName();
    }

}
```

ActiveAuthorityInheritTriggerProvider

```
package com.finebi.plugin.bi.authority.inherit;

import com.finebi.fun.impl.AbstractAuthorityInheritTriggerProvider;
import com.fr.intelli.record.Focus;
import com.fr.intelli.record.Original;
import com.fr.record.analyzer.EnableMetrics;

/**
 * @author gaattc
 * @version 6.0
 * Created by gaattc on 2022/8/30
 */
@EnableMetrics
public class ActiveAuthorityInheritTriggerProvider extends AbstractAuthorityInheritTriggerProvider {

    @Override
    @Focus(id = "com.finebi.plugin.bi.authority.inherit.ActiveAuthorityInheritTriggerProvider", text = "BI",
source = Original.PLUGIN)
    public boolean originTableInheritTriggerEnable() {
        return true;
    }
}
```

com.finebi.activate.extension.BIConfManagerActivator#registerAuthorityInheritTriggerProvider

```
private void registerAuthorityInheritTriggerProvider() {
    FineLoggerFactory.getLogger().info("---AuthorityInheritTriggerProvider---init start---");
    Set<AuthorityInheritTriggerProvider> providers = ExtraClassManager.getInstance().getArray
(AuthorityInheritTriggerProvider.XML_TAG);
    if (CollectionUtils.isNotEmpty(providers)) {
        providers.stream().findAny().ifPresent(i ->
            OriginTableInheritTriggerManager.INSTANCE.registerOriginTableInheritTriggerProvider(i.
originTableInheritTriggerEnable());
        );
    }
    FineLoggerFactory.getLogger().info("---AuthorityInheritTriggerProvider---init end---");
}
```