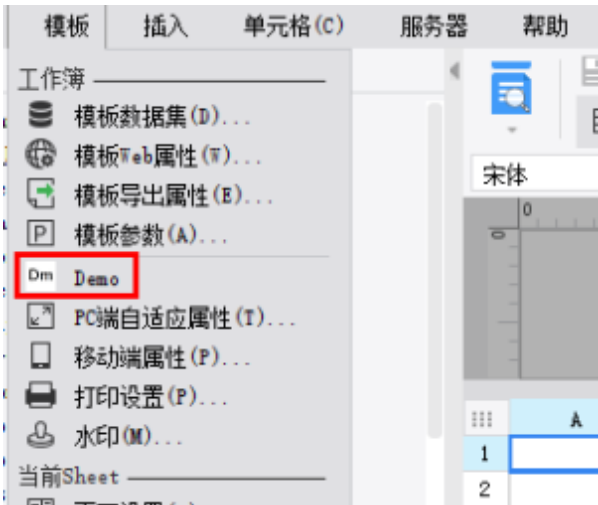


com.fr.design.fun.MenuHandler

-
-
-
-
-
-
-
-

// MenuHandler



MenuHandler.java

```
package com.fr.design.fun;

import com.fr.design.mainframe.toolbar.ToolBarMenuDockPlus;
import com.fr.design.menu.ShortCut;
import com.fr.stable.fun.mark.Mutable;

/**
 * @author richie
 * @date 2015-04-01
 * @since 8.0
 */
public interface MenuHandler extends Mutable {

    String MARK_STRING = "MenuHandler";

    int CURRENT_LEVEL = 1;

    int LAST = -1;
    int HIDE = -2;

    String HELP = "help";
```

```

String SERVER = "server";
String FILE = "file";
String TEMPLATE = "template";
String INSERT = "insert";
String CELL = "cell";
String BBS = "bbs";

/**
 *
 *
 * @param total
 * @return -1
 */
int insertPosition(int total);

/**
 *
 *
 * @return
 */
boolean insertSeparatorBefore();

/**
 *
 *
 * @return
 */
boolean insertSeparatorAfter();

/**
 *
 *
 * @return
 */
String category();

/**
 *
 *
 * @return
 */
Shortcut shortcut();

/**
 *
 *
 * @param plus
 * @return
 */
Shortcut shortcut(ToolBarMenuDockPlus plus);
}

```

Shortcut.java

```

package com.fr.design.menu;

import com.fr.stable.fun.impl.AbstractProvider;
import com.fr.stable.fun.mark.API;
import com.fr.stable.fun.mark.Mutable;

import javax.swing.*;

/**
 * MenuDefToolBarDef
 * august:ShortcutXMLabled
 * Menudef
 */
@API(level = Shortcut.CURRENT_LEVEL)
public abstract class Shortcut extends AbstractProvider implements Mutable {

```

```

public static final String TEMPLATE_TREE = "TemplateTreeShortCut";

public static final int CURRENT_LEVEL = 1;

public int currentAPILevel() {
    return CURRENT_LEVEL;
}

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

private MenuKeySet menuKeySet = null;

/**
 * menu1
 * @param menu menu
 */
public abstract void intoJPopupMenu(JPopupMenu menu);

/**
 *
 * @param toolBar
 */
public abstract void intoJToolBar(JToolBar toolBar);

public abstract void setEnabled(boolean b);

/**
 *
 * @return
 */
public abstract boolean isEnabled();

/**
 *
 * @param scs
 * @return
 */
public static final ToolBarDef asToolBarDef(Shortcut[] scs) {
    ToolBarDef def = new ToolBarDef();
    def.addShortcut(scs);

    return def;
}

public MenuKeySet getMenuKeySet() {
    return menuKeySet;
}

public void setMenuKeySet(MenuKeySet menuKeySet) {
    this.menuKeySet = menuKeySet;
}

/**
 *
 * @param isAuhtority
 */
public void notifyFromAuhtorityChange(boolean isAuhtority) {
}
}

```

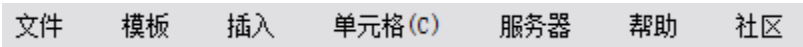
FR	8.0		
FR	9.0		
FR	10.0		

plugin.xml

```
<extra-designer>
    <MenuHandler class="your class name"/>
</extra-designer>
```

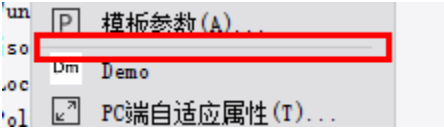
```
Set<MenuHandler> set = ExtraDesignClassManager.getInstance().getArray(MenuHandler.MARK_STRING);
MainDesignerToolBarMenuDockinsertMenu
```

MenuHandler



categoryFILETEMPLATEINSERTCELLSERVERHELPBBS

insertSeparatorBeforeinsertSeparatorAfter



```
ShortCut shortcut();ShortCut shortcut(ToolBarMenuDockPlus plus);categorydemo
```

ShortCutShortCutUpdateActionShortCut Action

[demo-menu-handler](#)