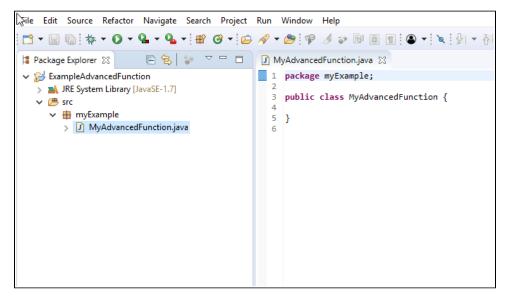
Advanced Function Basics

This section depicts a basic process of how an advanced function is created to be used in a Yellowfin report.

Note: Our example uses Eclipse to create an advanced function.

1. Create a new Java project and class.



 Add the required jar files to your build path. In order to compile an advanced function, you will need to include i4-core.jar and i4-mi.jar files into the build path. These can be found in your < YellowfinDirectory>/appserver/webapps/ROOT/WEB-INF/lib/ folder.

Properties for ExampleAdvand	redFunction	— 🗆 X
type filter text	Java Build Path	▼ ▼
 > Resource Builders Coverage Java Build Path > Java Code Style > Java Compiler > Java Editor Javadoc Location Project References Run/Debug Settings > Task Repository Task Tags > Validation WikiText 	 Source Projects Libraries Order and Expo JARs and class folders on the build path: id+core.jar - C:\Users\schro\Documents\Yellowfin Insta id+mi.jar - C:\Users\schro\Documents\Yellowfin Insta iAF. System Library [JavaSE-1.7] 	tt Add JARs Add External JARs Add Variable Add Library Add Class Folder Add Class Folder Edit <u>E</u> dit <u>R</u> emove <u>Migrate JAR File</u>
	< >>	
		Apply
?	Apply a	and Close Cancel

3. Set your class to extend AnalyticalFunction and add the unimplemented methods.

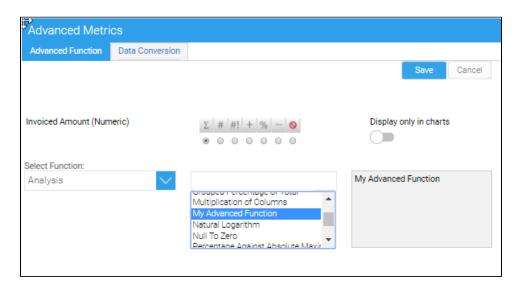
- Code your advanced function. See example of a basic accumulative total advanced function.
 Compile this into a jar file. For compatibility purposes, we recommend that you compile this with Java 7 JDK. If your function uses dependent libraries, you will need to place all of the jar files into a single folder, compress it into a .zip format, and then change the extension of that file to . yfp.

JAR Export			_		×	
JAR File Specification				\$		
Define which resources should be exported into the JAR.				-		
Select the resources to expo	t:					
> 🔳 🔂 ExampleAdvanc	edFunction	.clas 🕅 🕅 .clas				
Export generated class files and resources						
Export all output folders	for checked proje	cts				
Export Java source files a	ind resources					
Export refactorings for checked projects. Select refactorings						
Select the export destination				D		
JAR file: C:\Yellowfin\my-advanced-function.jar V Browse						
Options:						
Compress the contents of the JAR file						
Add directory entries						
Overwrite existing files without warning						
٢	r De els	Martin	Finish	6-		
0	< Back	Next >	Finish	Cance	ei	

6. Load your advanced function through Yellowfin's plug-in manager.

MyAdvancedFunction		×
MyAdvancedFunction		
Description		
	/	
imy-advanced-function.jar	×	
	+ Add more	
0.1		
Submit		

7. Once the advanced function has loaded, it will be immediately available for use in a report.



Previous topic: Overview Next topic: Advanced function creation