Table of Contents

[RESOURCES 3](#_Toc384371317)

[Access GitHub’s IHTSDO/ISAAC project 3](#_Toc384371318)

[Ensure have JDK 1.8 installed on system 3](#_Toc384371319)

[Ensure have Maven 3.0.5 installed on system 3](#_Toc384371320)

[Download the entire IHTSDO/ISAAC project from GitHub: 3](#_Toc384371321)

[Berkeley Database 3](#_Toc384371322)

[General Setup 4](#_Toc384371323)

[Setup your settings.xml file 4](#_Toc384371324)

[Setup Project in IDE 5](#_Toc384371325)

[Eclipse 5](#_Toc384371326)

[NetBeans 5](#_Toc384371327)

[IntelliJ 5](#_Toc384371328)

[Command Line 5](#_Toc384371329)

[Create a Git local repository of ISAAC 5](#_Toc384371330)

[Desktop Client 5](#_Toc384371331)

[Command Line 7](#_Toc384371332)

[Import local Git repository into IDE 7](#_Toc384371333)

[Eclipse 7](#_Toc384371334)

[NetBeans 11](#_Toc384371335)

[IntelliJ 11](#_Toc384371336)

[Command Line 11](#_Toc384371337)

[Build project 11](#_Toc384371338)

[Eclipse 11](#_Toc384371339)

[NetBeans 12](#_Toc384371340)

[IntelliJ 12](#_Toc384371341)

[Command Line 12](#_Toc384371342)

[Deploy Berkeley Database 12](#_Toc384371343)

[Eclipse 12](#_Toc384371344)

[NetBeans 13](#_Toc384371345)

[IntelliJ 13](#_Toc384371346)

[Command Line 13](#_Toc384371347)

[Run application 13](#_Toc384371348)

[Eclipse 13](#_Toc384371349)

[NetBeans 16](#_Toc384371350)

[IntelliJ 16](#_Toc384371351)

[Command Line 16](#_Toc384371352)

# RESOURCES

## Access GitHub’s IHTSDO/ISAAC project

1. Create a GitHub account at <http://github.com>
2. Access IHTSDO/ISAAC project at: <https://github.com/IHTSDO/ISAAC>

## Ensure have JDK 1.8 installed on system

1. Download [Java SE Development Kit 8](http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html).
2. Install as instructed.

## Ensure have Maven 3.0.5 installed on system

1. Download Maven 3.0.5 from <http://maven.apache.org/download.cgi>
2. Install as instructed.

## Download IHTSDO/ISAAC project from GitHub

1. ISAAC (top level folder)
   1. Contains the multi-module parent pom file.
2. data-model
   1. Contains shared application data models
3. gui-util
   1. Contains shared GUI utilities
4. import-export
   1. Contains Import/Export view code
5. info-model-view
   1. Contains a that displays data-model specific refsets
6. isaac-app
   1. Contains application window and menus.
7. isaac-app-interfaces
   1. Contains interfaces for application modules
8. isaac-workflow
   1. Contains workflow tooling
9. lego-view
   1. Contains LEGO view code
10. list-view
    1. Contains a list / batch view implementation
11. otf-util
    1. Contains utility code for accessing OTF APIs
12. refset-view
    1. Contains a refset view implementation
13. refex-view
    1. Contains a refset view implementation based on dynamic refsets
14. resources
    1. Contains documents & resources
15. search-view
    1. Contains search view code
16. taxonomy-view
    1. Contains taxonomy view code

## Download IHTSDO/ISAAC-PA project from GitHub

1. ISAAC-PA (top level folder)
   1. Contains the multi-module parent pom file.
2. app
   1. Contains the Maven POM to assemble an ISAAC application bundle.
3. config
   1. Contains the configuration for the ISAAC application.

## Berkeley Database

1. Log into VA-Archiva at <https://va.maestrodev.com/archiva/>
2. Download <https://va.maestrodev.com/archiva/repository/data-files/gov/va/isaac/db/isaac-sct/2014.01.31-build-1/isaac-sct-2014.01.31-build-1.zip>

# General Setup

## Setup your settings.xml file

1. Pull sample settings.xml file from <https://github.com/IHTSDO/ISAAC/tree/master/resources> folder.
2. Replace <username> and <password> with your VA-Archiva credentials:

…..

<servers>

<server>

<id>va-maestro</id>

<username>VA-ARCHIVA\_USER\_NAME</username>

<password>VA-ARCHIVA\_USER \_PASSWORD</password>

</server>

</servers>

…..

1. Define your repositories

…..

<profiles>

<profile>

<id>my-profile</id>

<activation>

<activeByDefault>true</activeByDefault>

</activation>

<repositories>

<repository>

<id>va-maestro</id>

<name>VA Maestro</name>

<url>https://va.maestrodev.com/archiva/repository/all/</url>

</repository>

</repositories>

<pluginRepositories>

<pluginRepository>

<id>va-maestro</id>

<name>VA Maestro</name>

<url>https://va.maestrodev.com/archiva/repository/all/</url>

</pluginRepository>

</pluginRepositories>

</profile>

</profiles>

…..

# Setup Project in IDE

## Eclipse

1. Via **Windows-Preferences**, ensure **Java-Installed JREs** version is using Java 8 JDK.
2. Via **Windows-Preferences**, ensure **Maven-Installations** version is using local maven 3.0.5
3. Via **Windows-Preferences**, ensure **Maven-User Settings** has the User Settings pointing to the proper settings.xml file updated in General Setup portion of this document
4. IF using Kepler (Eclipse 4.3.2, earlier versions won’t work) after configuring Java 8, you will need to install these two things
   1. <http://marketplace.eclipse.org/content/java-8-support-eclipse-kepler-sr2#.U21zK1c3fkA>
   2. (from eclipse marketplace): Java 8 support for m2e for Eclipse Kepler SR2

Then, to get rid of warnings you may need to

* 1. For each project disable project-specific compiler settings
  2. Set the corresponding compiler warning setting to “Ignore”.

## NetBeans

## IntelliJ

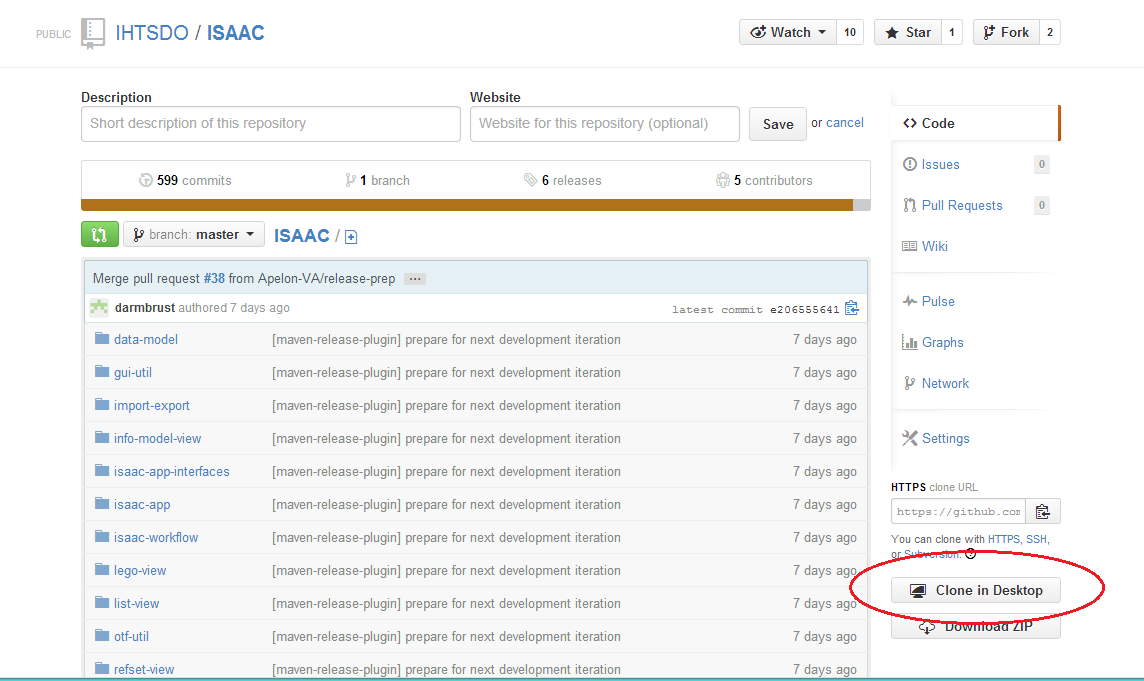
## Command Line

# Create a Git local repository of ISAAC

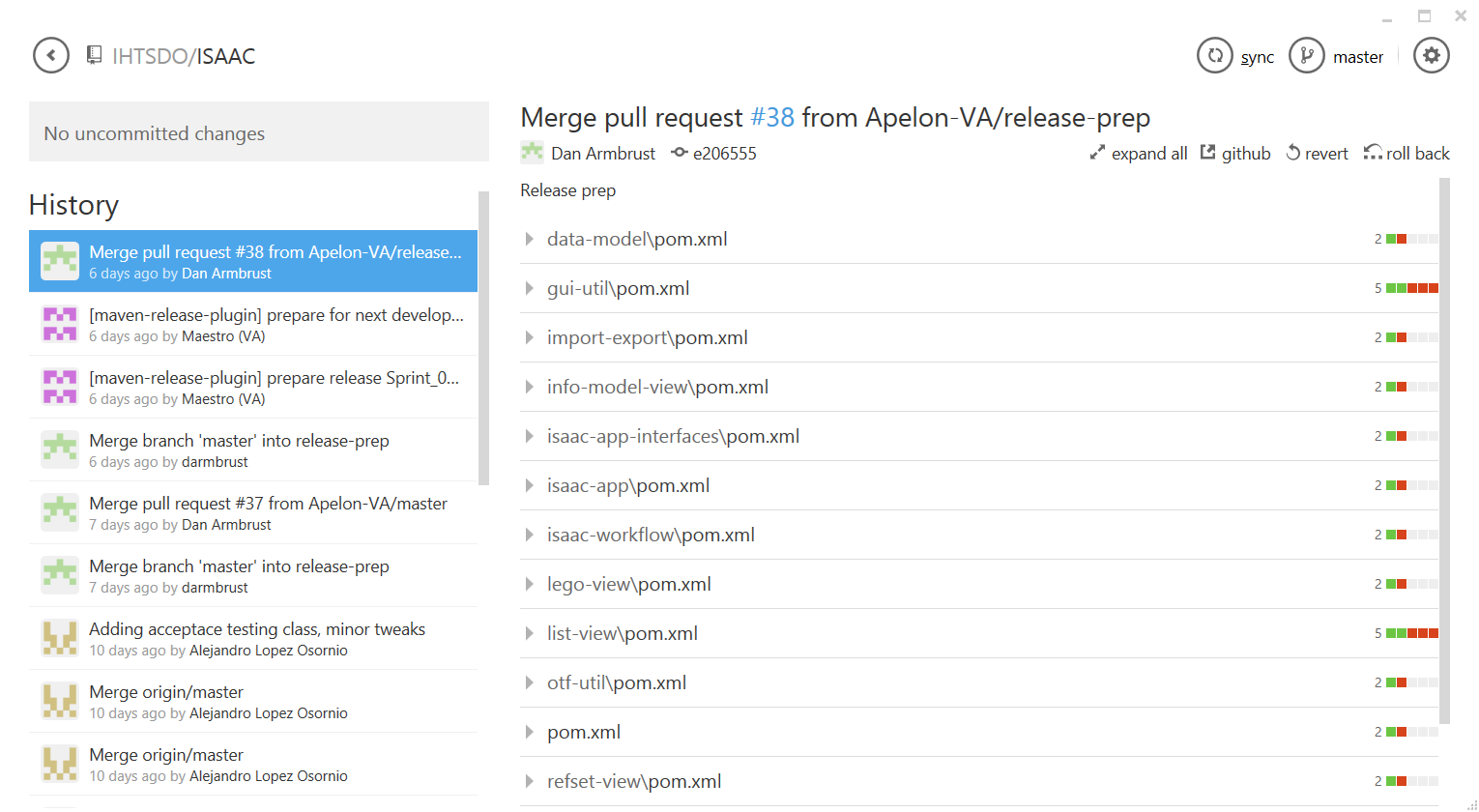
## Desktop Client

Demonstrated for Windows via Windows GitHub client

1. Download & install Windows GitHub client from <https://windows.github.com/>
2. Go to <https://github.com/Apelon-VA/ISAAC>
3. Select ‘Clone in Desktop’ button at lower-right



1. Opens GitHub client
2. Content will download
3. Once done, GitHub client will look something like this:



1. Similarly, go to <https://github.com/Apelon-VA/ISAAC-PA> and ‘Clone in Desktop’ as before.
2. Locate local repositories on computer
   1. In Windows GitHub client, click on settings button
   2. Select **Open in explorer** option. Explorer opens in directory containing local Git repository.
   3. Make note of directory for later.

## Command Line

Demonstrated in a Unix/Linux terminal

1. Change directory to the location you wish to clone the repositories.

cd /Projects

1. Download the projects with the Git “clone” command.

git clone https://github.com/Apelon-VA/ISAAC

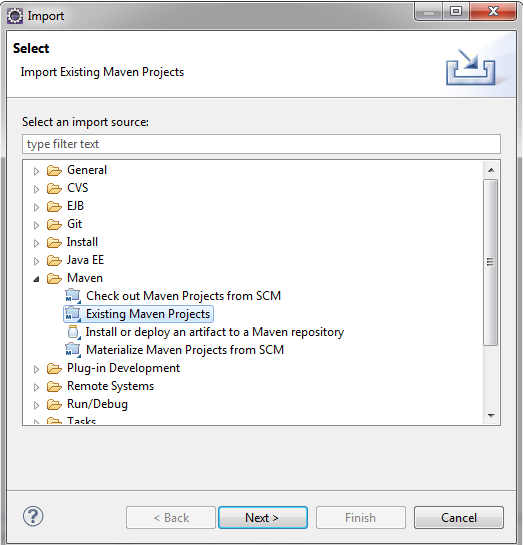
git clone https://github.com/Apelon-VA/ISAAC-PA

1. Make note of directory for later.

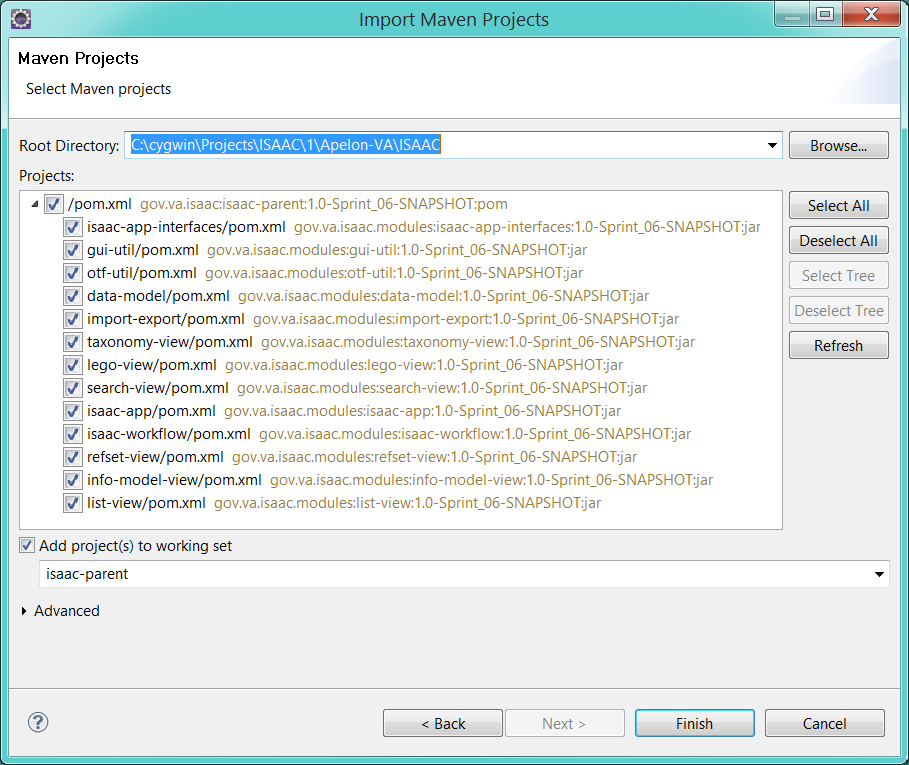
# Import local Git repository into IDE

## Eclipse

1. Select **File-Import**
2. Select **Maven-Existing** Maven Project option



1. Point **Root Directory** to the “ISAAC” folder to the local GitHub repository you used above, and select all the sub-projects.



1. Click the **Finish** button when done.
2. Repeat above steps to import the "ISAAC-PA" sub-projects from the local GitHub repository you used above.

## NetBeans

## IntelliJ

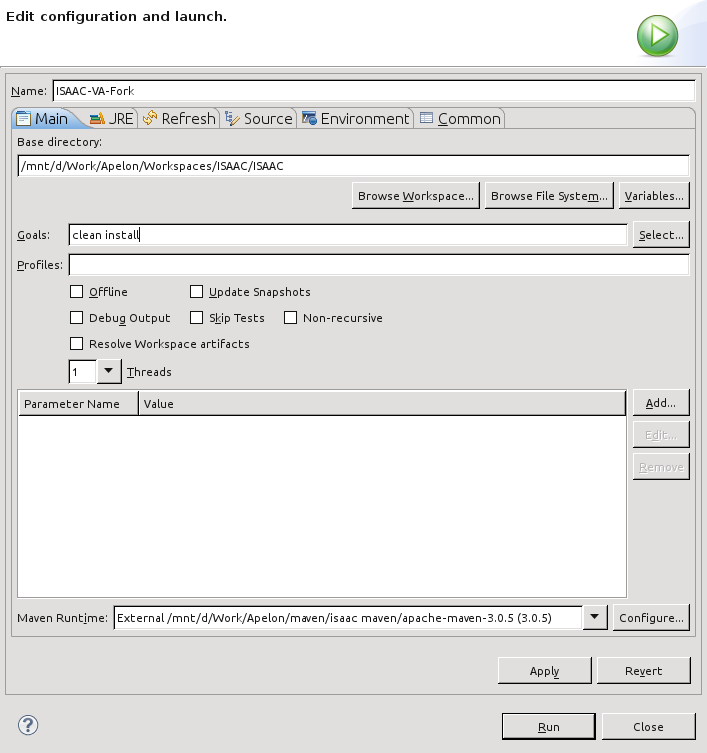
## Command Line

# Build project

The following sections describe how to build the ISAAC application.

## Eclipse

1. Create a Run Configuration for the project
   1. Right-click the “isaac-parent” project and select **Run As-Maven Build…** from the menu
   2. Type “clean install” in the **Goals** field of the **Edit Configuration** dialog:



1. Click the **Run** button to build the ISAAC GUI core and all its modules.
2. Repeat the above steps for the "isaac-pa-parent" project to build the ISAAC-PA project and assemble an ISAAC application bundle.

## NetBeans

## IntelliJ

## Command Line

Demonstrated in a Unix/Linux terminal

1. Change directory to the location containing the ISAAC repository cloned in [Create a Git local repository of ISAAC](#_Create_a_Git).

cd /Projects/ISAAC/

1. Build the project with the Maven as usual.

mvn clean install

1. Next, change directory to the location of the ISAAC-PA repository cloned above.

cd /Projects/ISAAC-PA/

1. Build the project with the Maven as usual.

mvn clean install

1. The ISAAC application bundle built into a ZIP file in the “app/target” folder.

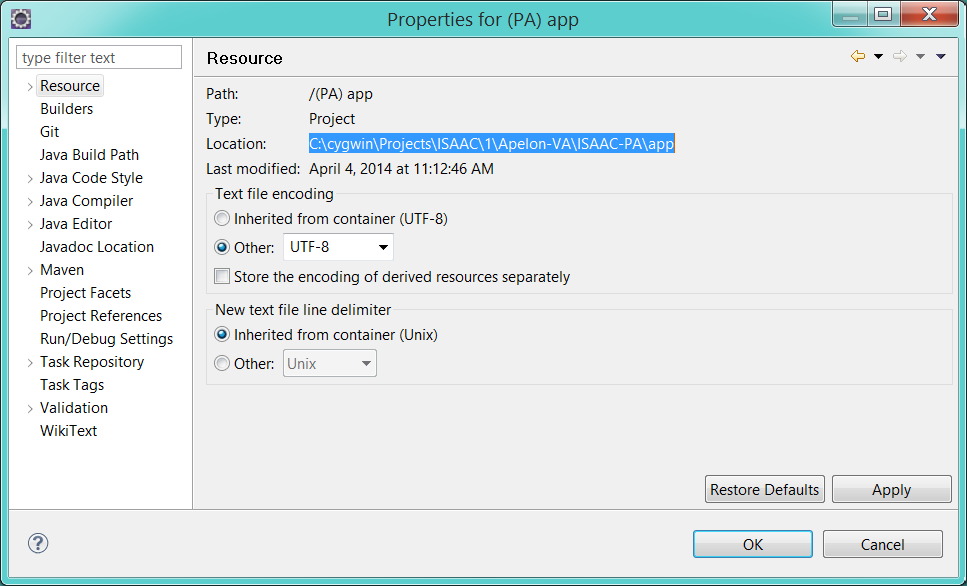
# Deploy Berkeley Database

The ISAAC application will need a database to connect to. The following steps describe how to deploy the database described in the [Berkeley Database](#_Berkeley_Database) section.

1. Locate the isaac-sct-2014.01.31-build-1.zip file you downloaded.
2. Extract the ZIP file into a directory called isaac-sct-2014.01.31-build-1.
3. Note the extracted “berkeley-db” folder inside. This contains the database content.

## Eclipse

1. Right-click the “app” project and select **Properties**.
2. Select the **Resources** group on the left.
3. In the Properties dialog, make a note of the **Location** field:



1. Copy the “berkeley-db” folder you extracted above into this directory.

## NetBeans

## IntelliJ

## Command Line

Demonstrated in a Unix/Linux terminal

1. Return to the directory containing the “target” folder of the “app” project.

cd /Projects/ISAAC-PA/app/target

1. Copy the “berkeley-db” folder you extracted above into this directory.

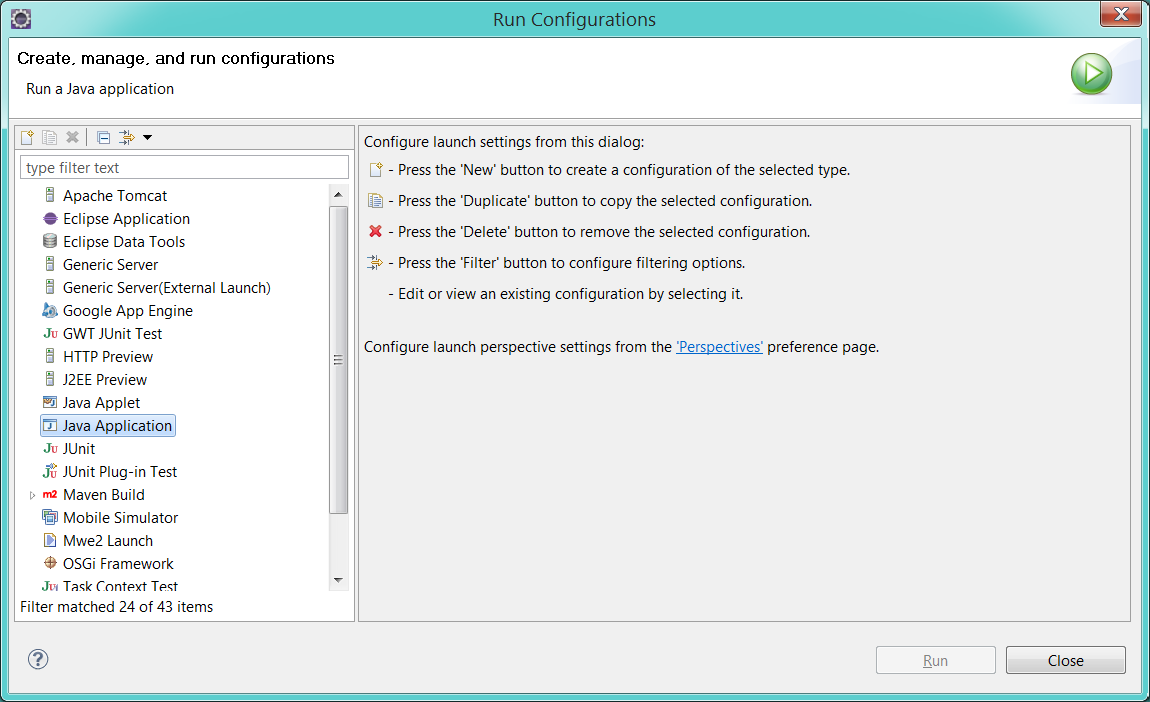
cp -r /isaac-sct-2014.01.31-build-1/berkeley-db .

# Run application

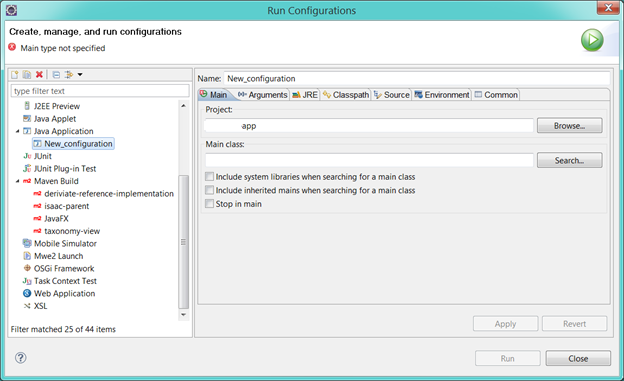
The following sections describe how to run the ISAAC application.

## Eclipse

1. Create a Run Configuration for the app
   1. Right-click the “app” project and select **Run As > Run Configurations …** from the menu. You should see something like this:



* 1. Right-click the “Java Application” item and select **New**.
  2. A new run configuration will be created. You should see something like this:



* 1. Change the new run configuration whatever you like.
     1. For example, enter “App” into the **Name** field.
  2. In the **Main class** field, enter “gov.va.isaac.gui.App”.

1. Click the **Run** button to start the ISAAC application.

## NetBeans

## IntelliJ

## Command Line

Demonstrated in a Unix/Linux terminal

1. Change directory to the location containing the “target” folder of the “app” project you built in [Build project](#_Build_project).

cd /Projects/ISAAC-PA/app/target

1. Unzip the ISAAC App ZIP file.

jar xvf isaac-app.zip

1. This will extract several scripts and a lib folder of JAR files into the “target” folder. Locate the appropriate script and execute it.

./isaac-app64.sh