* Installare e configurare il PATH per Java JDK 1.8
* Installare e configurare Eclipse Oxygen for Java EE developers
  + Download da <http://www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/oxygen/3a/eclipse-jee-oxygen-3a-win32-x86_64.zip>
  + Scompattare l’archivio e salvare la cartella nel percorso desiderato
  + Nel caso il programma dia errore all’avvio per virtual machine mancante, aggiungere la riga di configurazione

“

-vm

<<percorso virtual machine>>

“

Come prima istruzione del file di configurazione eclipse.ini presente nella stessa cartella del file eseguibile

* Installare e configurare Tomcat 9.0 in Eclipse
  + Download Tomcat da <https://tomcat.apache.org/download-90.cgi> selezionando la versione corretta a seconda del Sistema operativo
  + Scompattare in una cartella il file compresso appena scaricato
  + In Eclipse, andare in Window -> preferences -> Servers -> Runtime environments
  + Cliccare su Add…
  + Selezionare Apache -> Apache Tomcat v9 e cliccare Next
  + Inserire la cartella in cui è stato scompattato tomcat e cliccare su Finish
  + Aprire la view Servers andando in Window -> Show view -> Other -> Server -> Servers
  + Aggiungere un nuovo Server con l’installazione di Tomcat preconfigurata cliccando sul testo evidenziato nella finestra e cliccando Finish nella finestra che si aprirà
* Se non presente ------ Installare il plugin Maven2Eclipse
  + Andare in Help -> Install new software…
  + Cliccare su Add…
  + Inserire come nome Maven2Eclipse
  + Inserire come indirizzo <http://download.eclipse.org/technology/m2e/releases>
  + Cliccare ok
  + Selezionare Maven Integration For Eclipse
  + Completare l’installazione
* Aprire la view Git Repositories
  + Window -> Show view -> Other -> Git -> Git Repositories
* Clonare I progetti RMLDataRetrivier e RMLDataRetrieverTest (eseguire per ogni progetto)
  + Cliccare su Clone a Git repository
  + Inserire i dati della repository e cliccare Next due volte
  + Spuntare la voce “Import all existing Eclipse projects… “
  + Cliccare Finish e il progetto verrà clonato e importato nel workspace
* Inserire i file di configurazione della libreria nelle directory predefinite
  + Copiare la cartella config, presente tra i sorgenti del progetto RMLDataRetriever, all’interno della directory home del proprio profilo (diversa a seconda del sistema operativo)

Il path dovrà essere <<user-home>>/RMLDataRetriever/config

* + Verificare che la cartella config contenga fitbit.xml e nokia-health.xml
  + Modificare i path delle directory che utilizzerà la libreria, all’interno dei file xml di configurazione e salvare
* Configurare ed effettuare la build di RML Data Retriever
  + Aprire le proprietà del progetto ed assicurarsi che tutte le librerie presenti nel build path puntino ai percorsi giusti (soprattutto la JRE di Java, che deve puntare alla directory jdk e non a quella jre), nel caso modificare
  + Cliccare con il tasto destro sul progetto -> Run as… -> Maven build
  + Inserire i Goals “clean install” e cliccare Run
  + Effettuare un refresh dopo la build e verificare che nella cartella /target ci sia il file .jar compilato della libreria
* Configurare ed avviare RML Data Retriever Test con Tomcat
  + Modificare, nel buildpath del progetto Test, i percorsi che puntano alle librerie esterne servlet-api (inserire il path dell’installazione di tomcat) e RMLDataRetriever (inserire il path della cartella “target” dove è presente il .jar della libreria appena creato)
  + Cliccare con il tasto destro sul progetto RML Data Retriever Test
  + Run as… -> Run on server