

Eclipse

Installationshinweise

Eclipse

- Voraussetzung: Java muss auf Ihrem PC installiert sein.
- XML Editor mit Unterstützung für DTD und XML-Schemas ist in praktisch allen Eclipse-Paketen integriert.
- JAXB: Eine Implementierung ist nicht mehr Teil von Standard Java und muss nachinstalliert werden.
- XPath-Plugin und XSLT-Unterstützung sind nicht Teil der „Eclipse IDE for Java Developers“.
- [Eclipse IDE for Enterprise Java and Web Development](#) verwenden.
- Im folgenden wird von der Enterprise Version ausgegangen.
- Die Schritte wurden alle ohne Hochschulproxy durchgeführt.

Inhalt

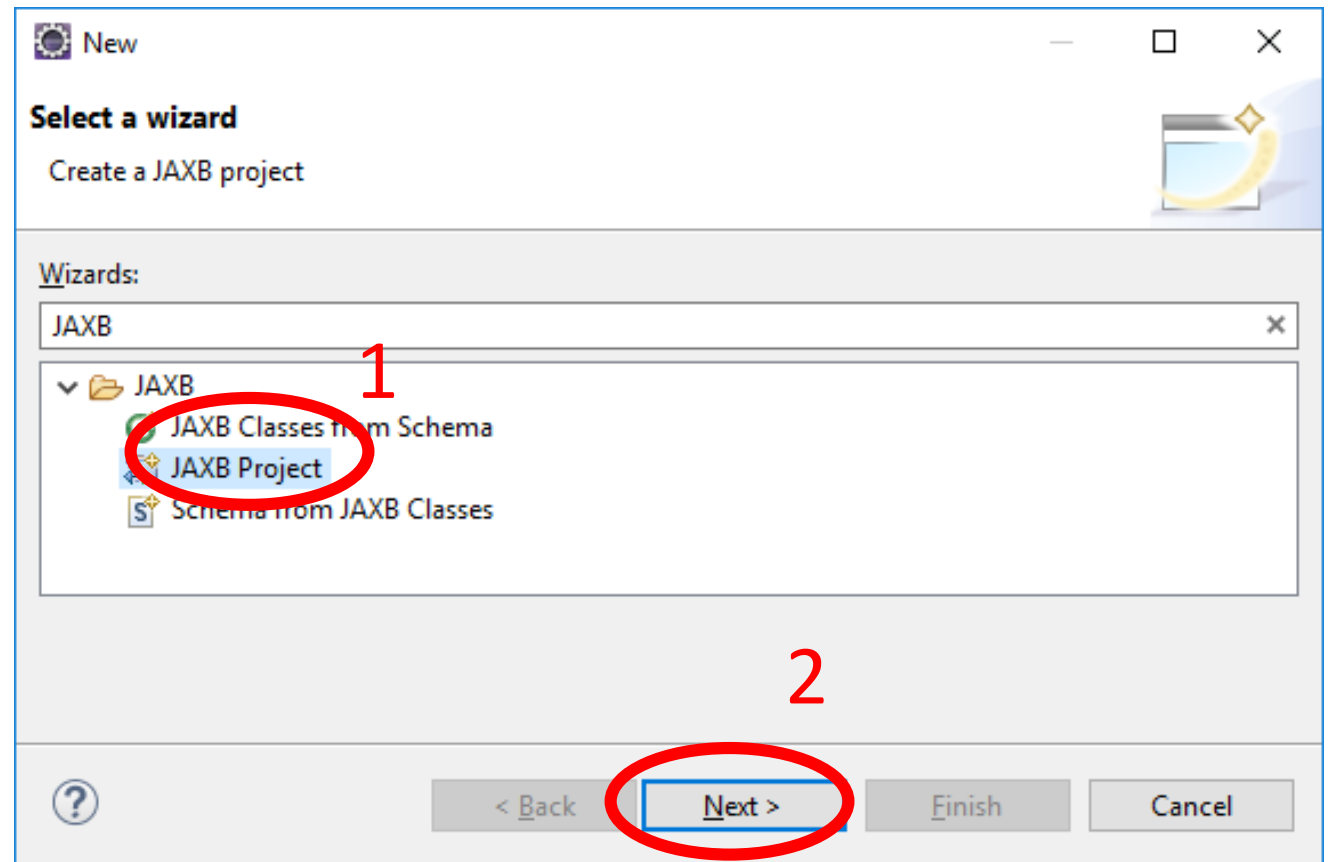
- JAXB für Aufgabe 4 installieren und Schema zu Java-Klassen-Compiler aufrufen.
- XPath-View für Aufgabe 5
- Aufrufe für Stylesheet-Programme mit internen Prozessor für Aufgabe 6.

JAXB für Eclipse installieren

- Da Java nicht mehr mit einer JAXB-Implementierung ausgeliefert wird, muss für Eclipse eine nachinstalliert werden.
- Dazu muss ein JAXB-Projekt erstellt werden. Die Eclipse eigene Implementierung kann leider im Verlauf des Wizards nicht mehr heruntergeladen werden.
- Die nächsten Folien zeigen deswegen die Schritte zur Verwendung von Jakarta JAXB.

JAXB-Projekt erstellen (1/6)

- Neues Projekt erstellen.
- Im Projekt-Wizard „JAXB Project“ anwählen (1).
- Weiter mit „Next“ (2).



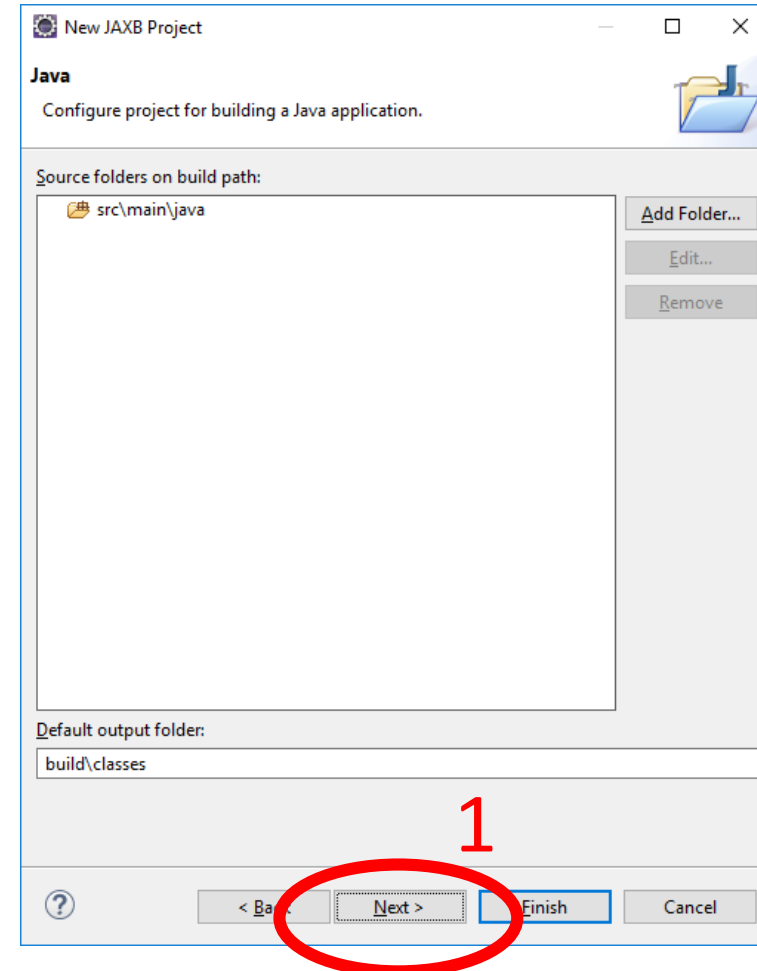
JAXB-Projekt erstellen (2/6)

- Sinnvollen Projektnamen angeben (1).
- Weiter mit „Next“ (2).

The screenshot shows the 'New JAXB Project' dialog box. The title bar says 'New JAXB Project'. Below it, the section 'JAXB Project' has the instruction 'Configure JAXB project settings.' followed by a red '1'. The 'Project name' field contains 'Movies-JAXB' and is circled in red. Below it, the 'Project location' section has a checked box for 'Use default location' and a 'Location' field showing a path on the desktop. The 'Target runtime' is set to '<None>'. The 'JAXB version' is set to '2.2'. The 'Configuration' section shows 'Default Configuration'. The 'Working sets' section has an unchecked box for 'Add project to working sets'. At the bottom, the 'Next >' button is circled in red and labeled with a red '2'. Other buttons include '< Back', 'Finish', and 'Cancel'.

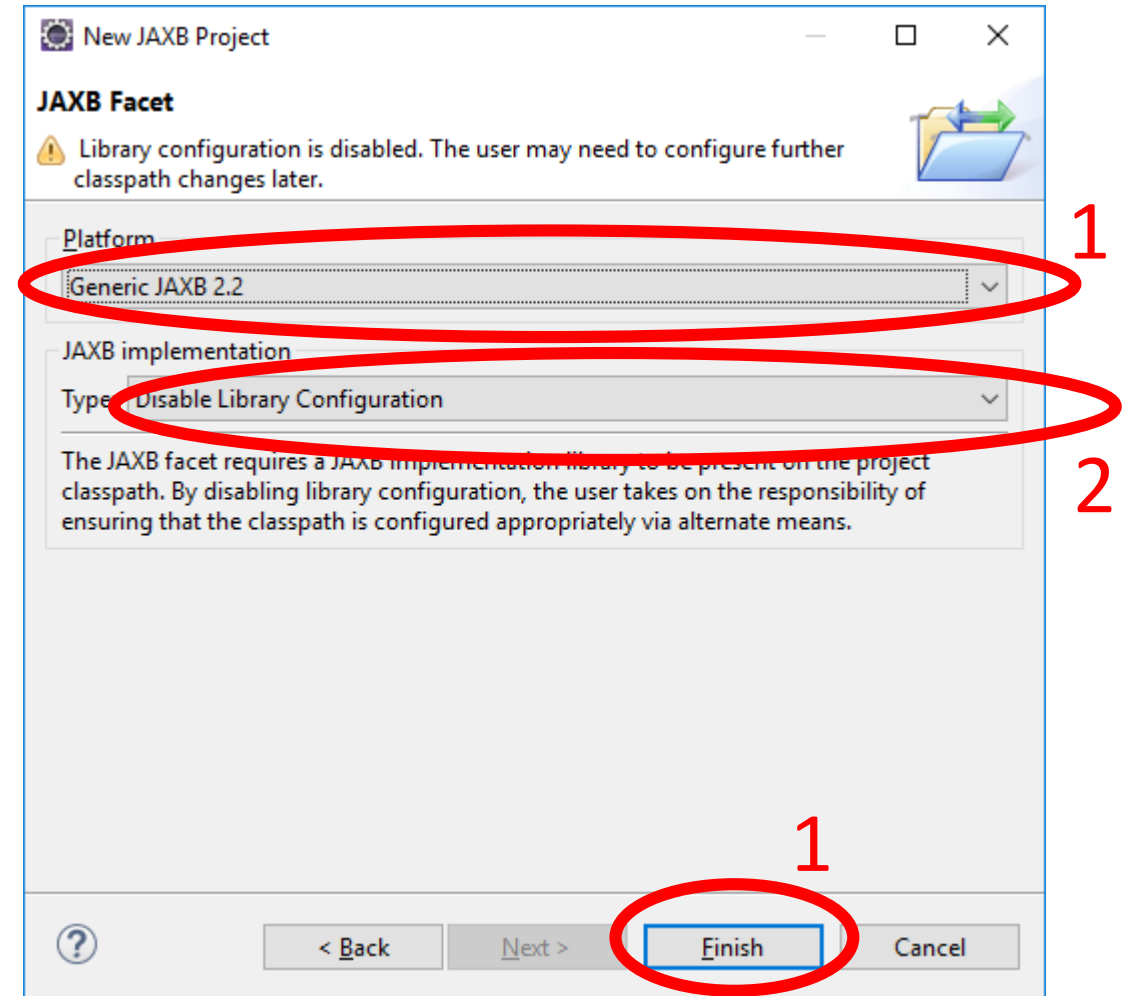
JAXB-Projekt erstellen (2/6)

- Voreinstellung übernehmen.
- Weiter mit „Next“ (1).



JAXB-Projekt erstellen (3/6)

- Bei „Platform“ „Generic JAXB 2.2“ auswählen (oder höhere Version, falls vorhanden). (1)
- Bei „JAXB implementation“ den Typ „Disable Library Configuration“ auswählen. (2)
- Weiter mit „Finish“ (3).
- Die JAXB-Bibliotheken werden später dem Projekt hinzugefügt.



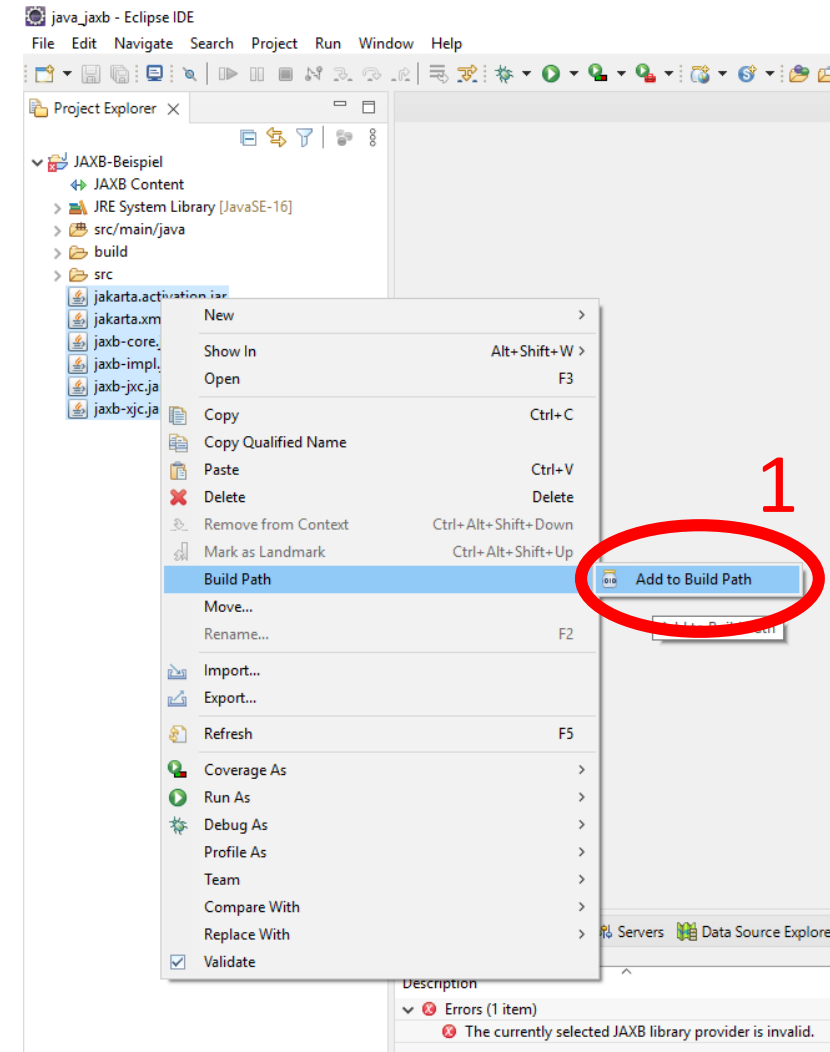
JAXB-Projekt erstellen (4/6)

- Im Projekt sind noch Fehler, weil eine JAXB-Implementierung fehlt.
- Jakarta JAXB-Implementierung herunterladen. (1)
- <https://eclipse-ee4j.github.io/jaxb-ri/>
- Das Heruntergeladene Archive entpacken und die Java-Archiv-Dateien im Ordner jaxb-ri\mod kopieren und zum Projekt hinzufügen (z.B. via Drag-and-Drop in den Projektordner)



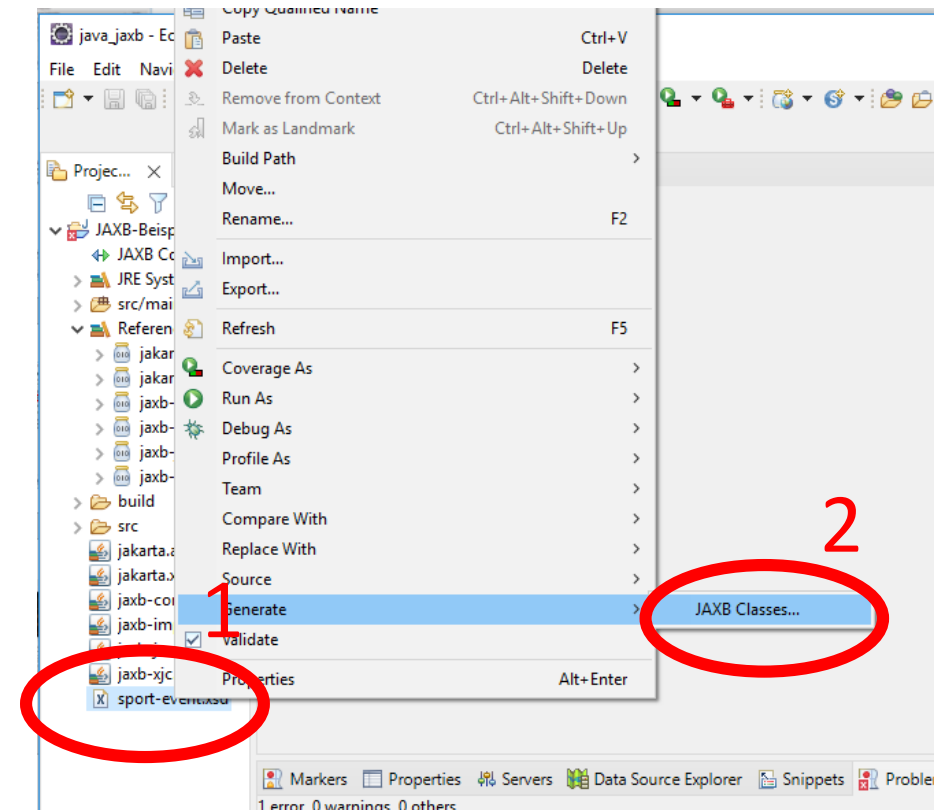
JAXB-Projekt erstellen (5/6)

- Die hinzugefügten Java-Archive anwählen und mit rechter Maustaste das Menü „Add to Build Path“ auswählen. (1)
- Der weiterhin angezeigte Fehler wird vermutlich angezeigt, weil Eclipse nur die Version 2.2 von JAXB unterstützt.
- Version 3.0 funktioniert allerdings.



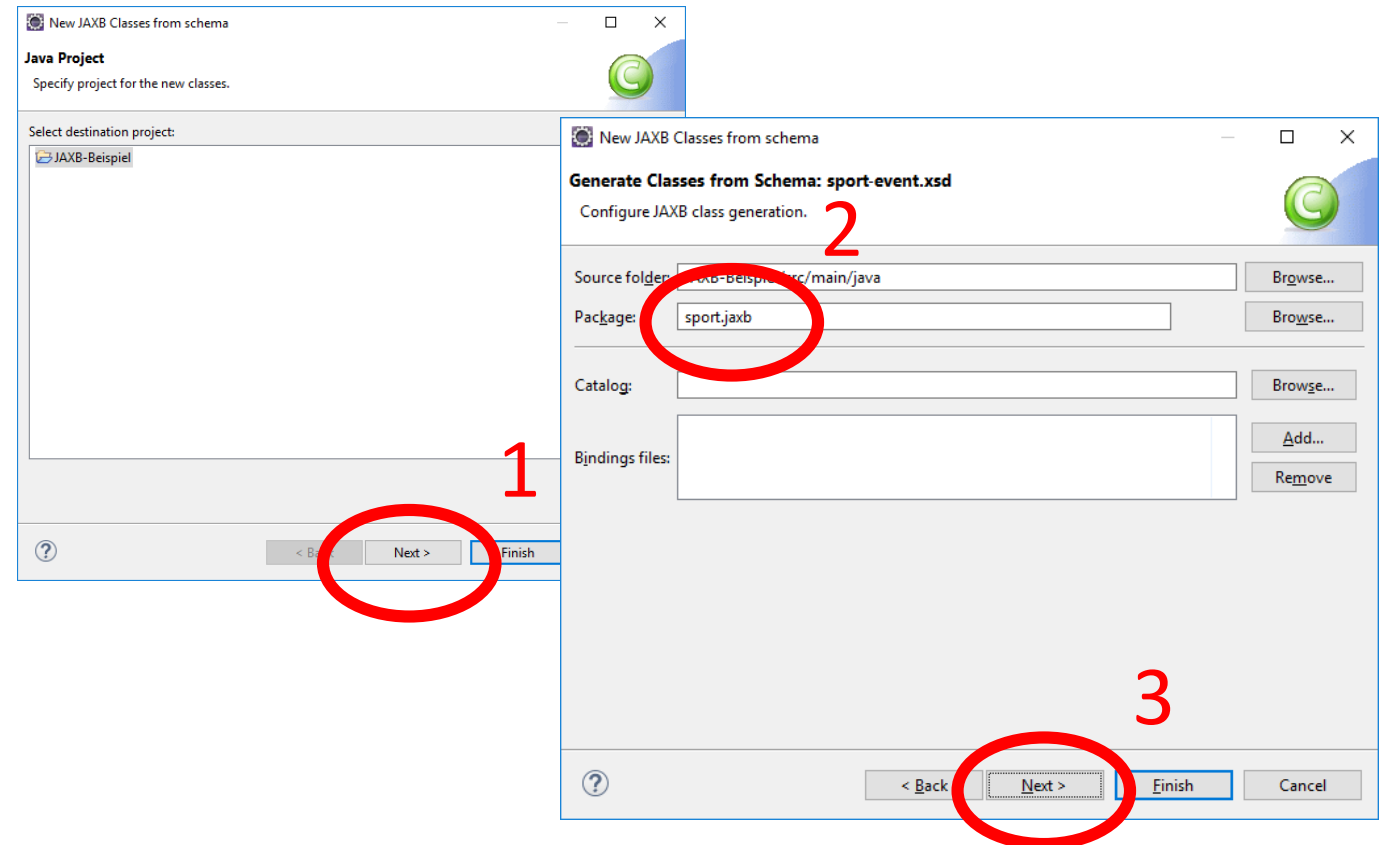
JAXB-Klassen erzeugen (6/6)

- Ein XML-Schema zum Projekt hinzufügen. (1)
- Das Schema anwählen und mit rechter Maustaste „Generate“ und „JAXB Classes“ anwählen. (2)



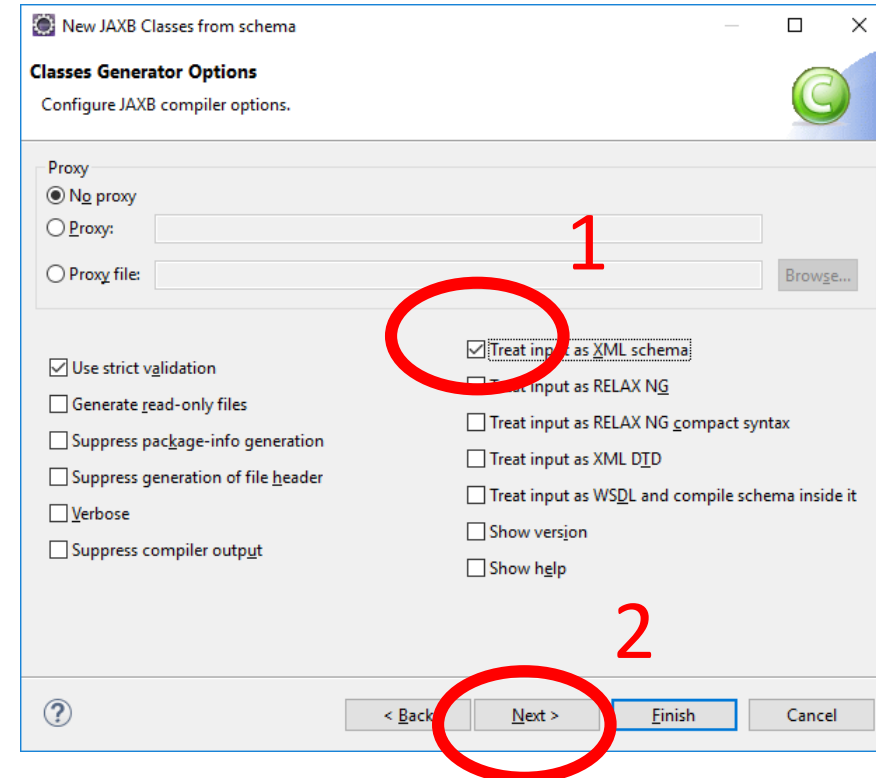
JAXB-Klassen erzeugen (1/4)

- Beim ersten Dialog „Next“ anwählen. (1)
- Beim zweiten eine sinnvollen Paketnamen angeben. (2)
- Weiter mit „Next“. (3)



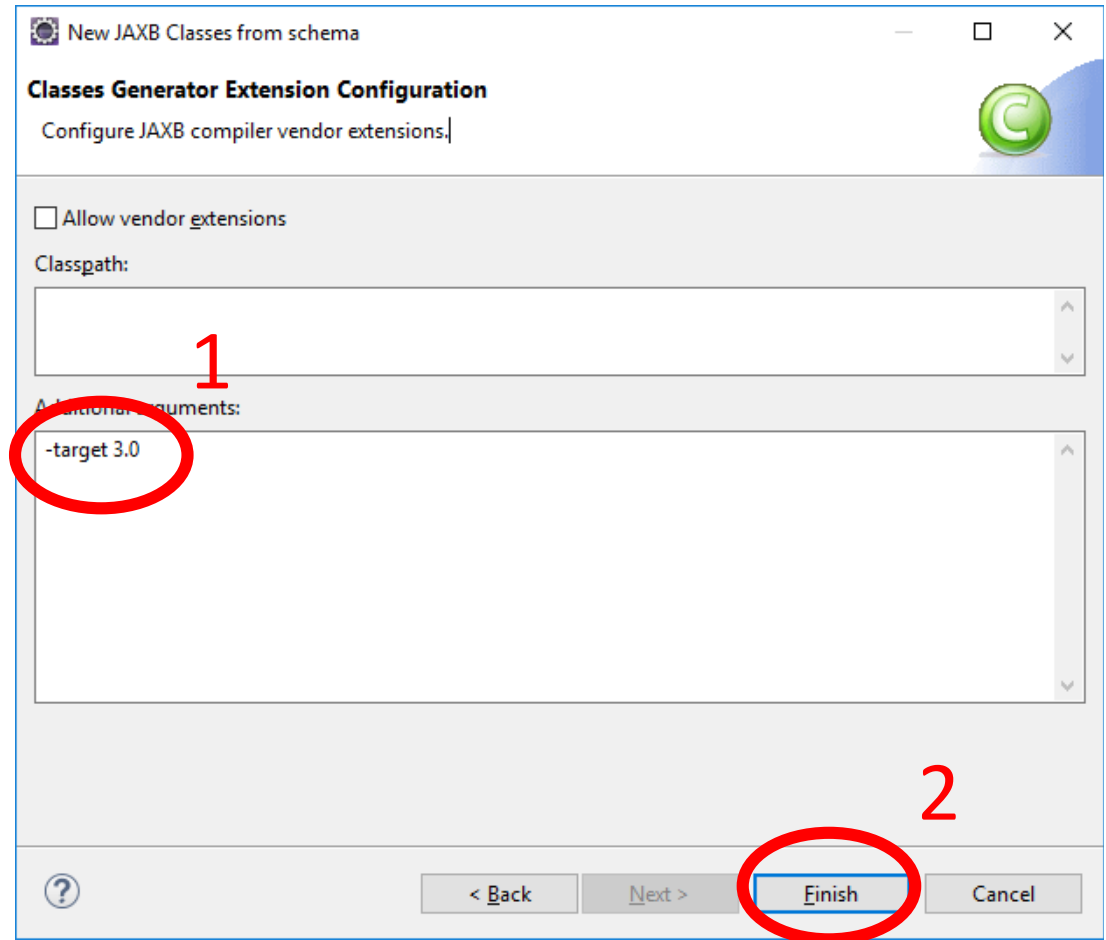
JAXB-Klassen erzeugen (2/4)

- „Treat input as XML schema“ anwählen (1).
- Weiter mit „Next“ (2).



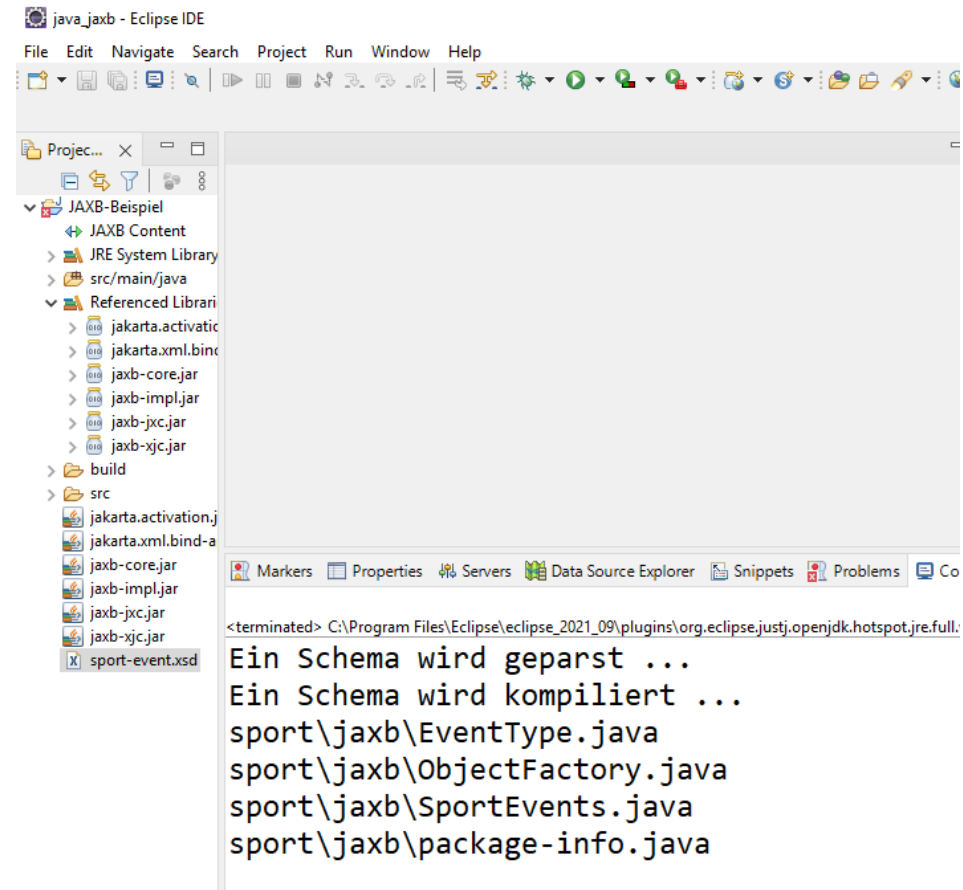
JAXB-Klassen erzeugen (3/4)

- Bei „Addition arguments“ die Version auf „3.0“ abändern (1).
- Falls eine aktuellere Version von Jakarta-JAXB verwendet wird, muss diese hier angegeben werden.
- Mit „Finish“ beenden (2).



JAXB-Klassen erzeugen (4/4)

- Der Schema zu Java-Compiler wird aufgerufen.
- Wenn das Schema korrekt war, sollte kein Fehler auftreten.
- Refresh beim Projekt (F5) ausführen, damit die erzeugten Klassen im Projektordner angezeigt werden.

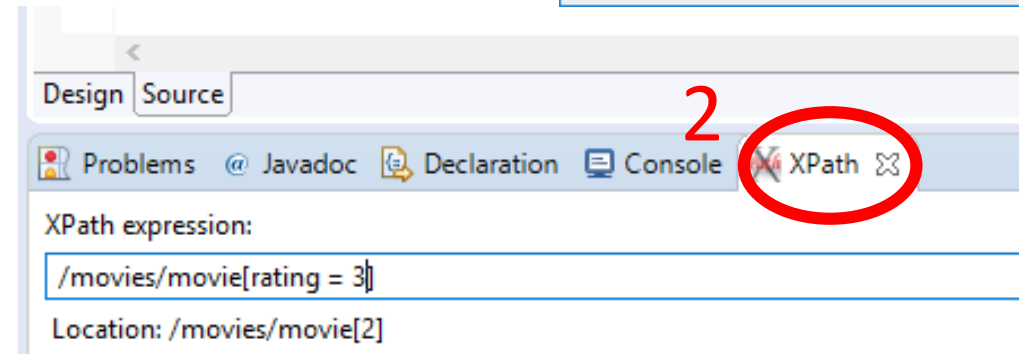
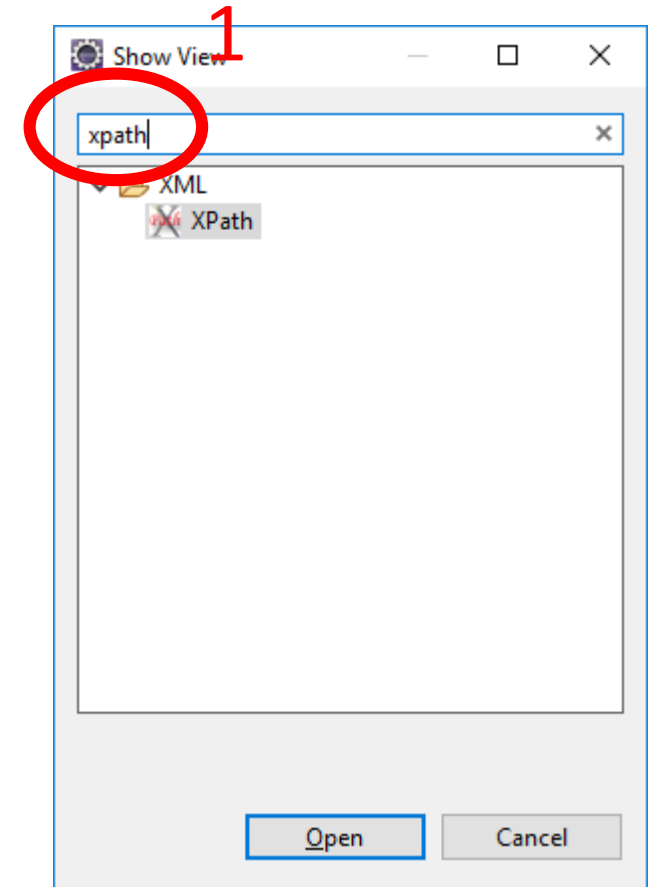


JAXB-Klassen

- Paketname wurde umbenannt von:
javax.xml.bind
- Zu:
jakarta.xml.bind
- Manuell Imports berichtigen oder diese löschen und neue Imports mit Vervollständigung erzeugen.
- Eclipse unterstützt JAXB 3.0 ggf (noch) nicht. Der entsprechende Fehler kann ignoriert werden, da es trotzdem funktioniert.

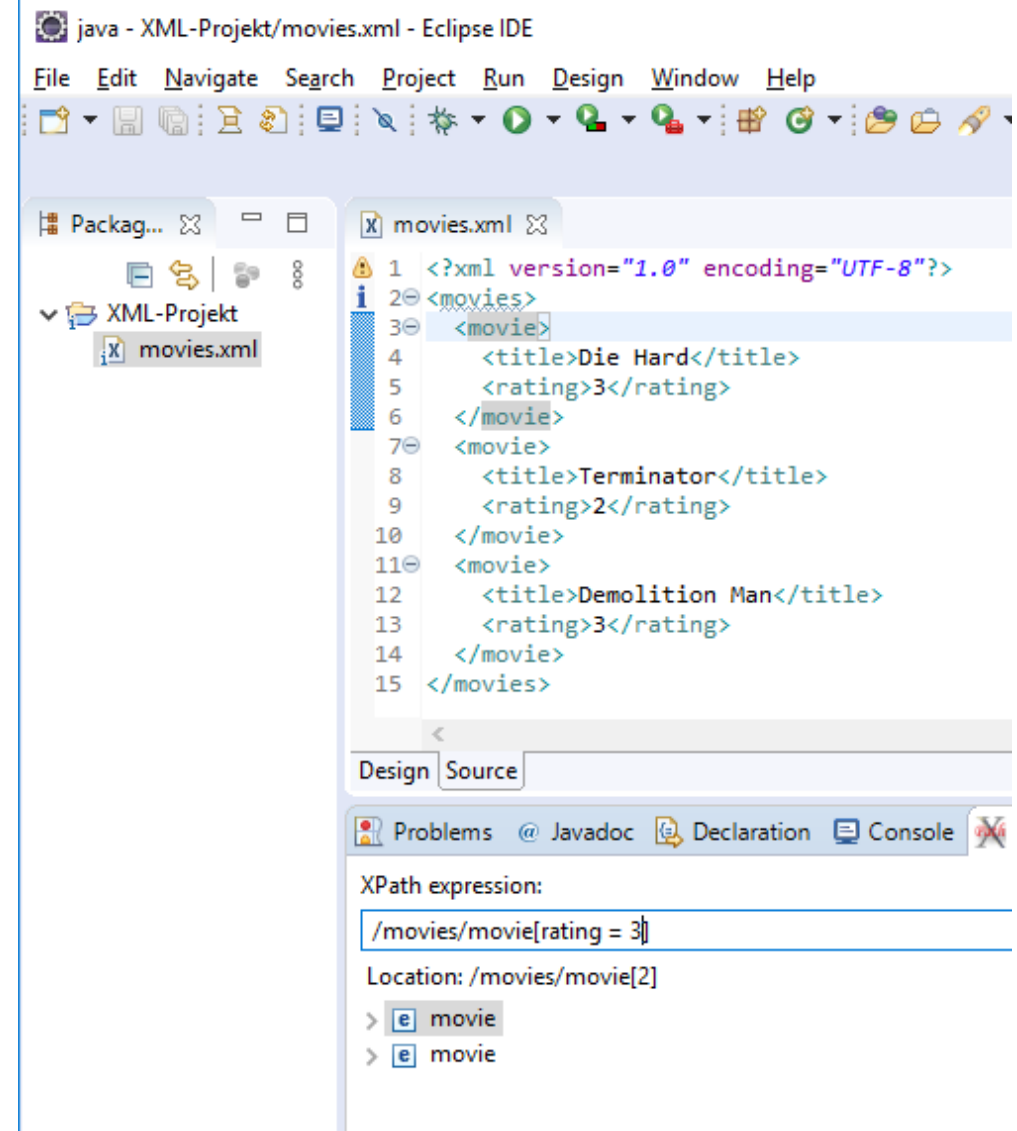
Eclipse XPath-View (1/2)

- Der View für die Eingabe von XPath-Ausdrücken muss über „Window->Show View->Other...“ und dann „XPath“ im Suchfeld eingeben, ausgewählt werden. (1)
- Der View erscheint unterhalb des Editors.
- Die Eingabe im View bezieht sich immer auf das aktive XML im Editor.
- Wenn Sie dieses Plugin für die XPath-Aufgabe verwenden, sollten im XML besser keinen Namensraum mehr deklarieren und der Verweis zum XSD löschen, da es keine Namensraum unterstützt.
- Es gibt andere XPath-Plugins im Marketplace, die auch die Angabe von Namensräumen unterstützten.



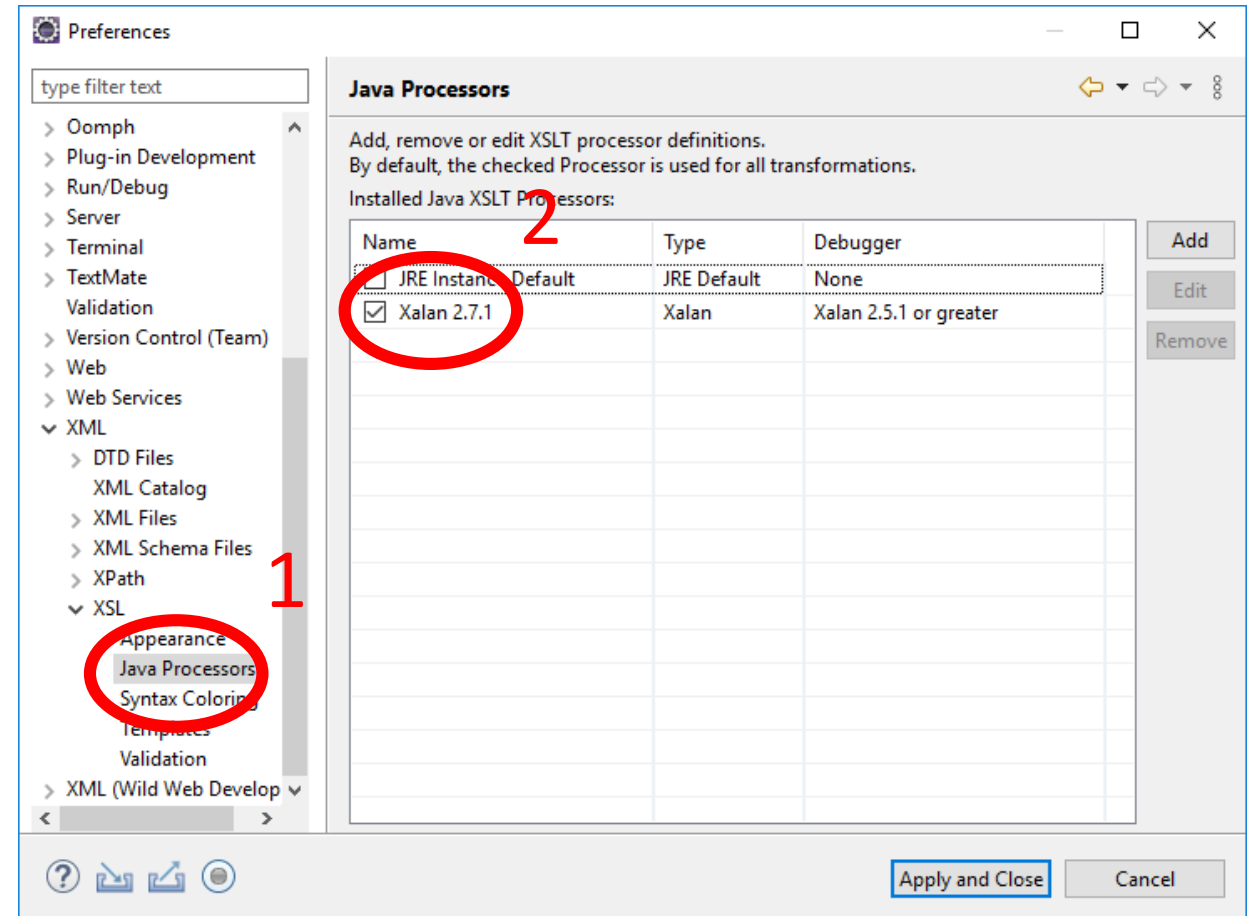
Eclipse XPath-View (2/2)

- Die Eingabe des Xpaths (1) wird auf das aktive XML-Dokument (2) angewendet
- Die Ergebnisknoten werden in einer hierarchischen Sicht im View angezeigt (3)



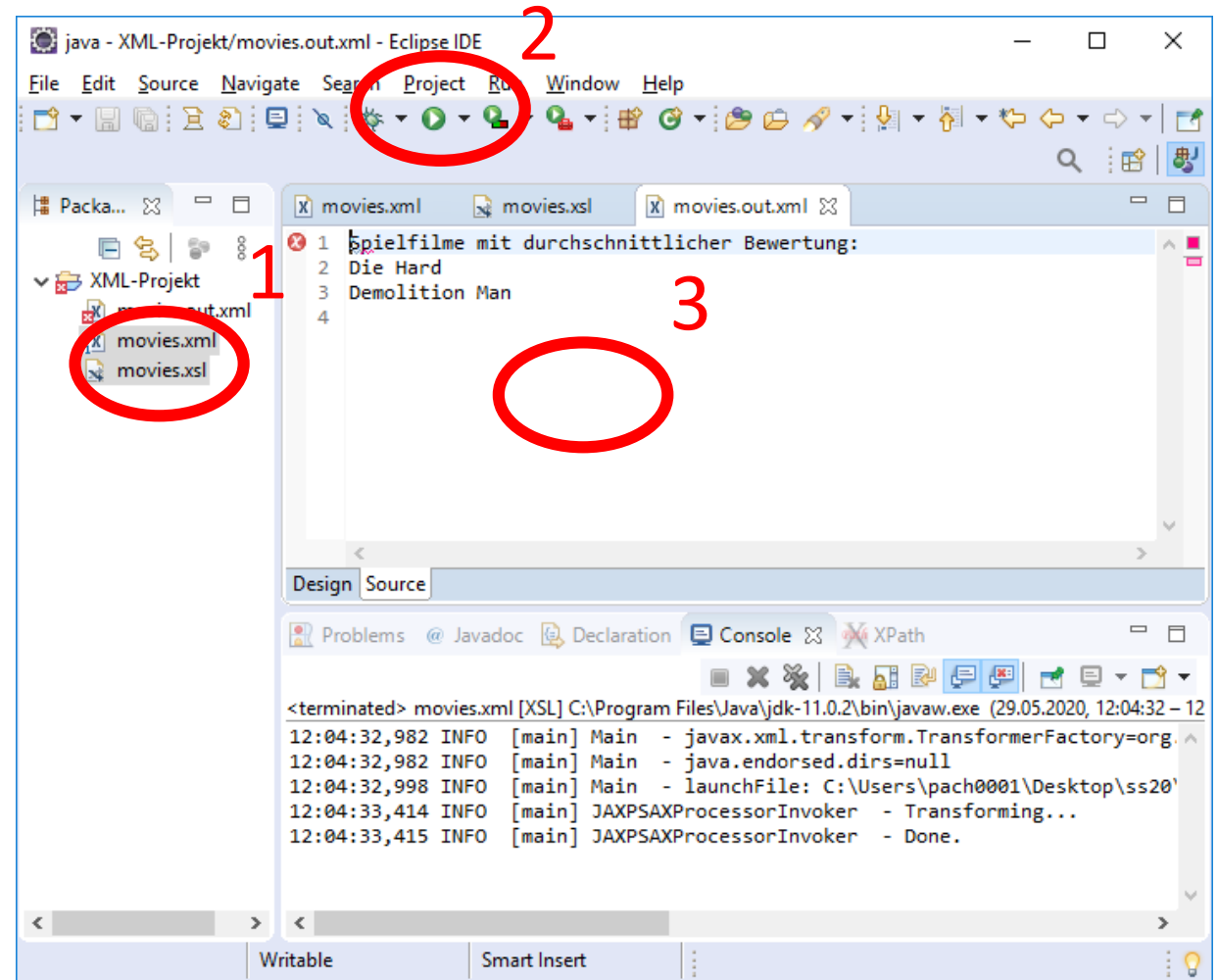
Eclipse Stylesheet Prozessor (1/4)

- Der standardmäßig eingestellte XSL-Prozessor hat nicht funktioniert (NullPointerException beim Ausführen von Stylesheets)
- Deswegen Xalan verwenden.
- Bei den „Preferences“ unter „XML-> XSL -> Java Processors“ auswählen (1).
- Xalan sollte Teil der „Eclipse IDE for Enterprise Java and Web Developers“ sein (2).
- Mit Xalan kann ein Stylesheet auch im Quelltext debuggt werden (nicht ausprobiert).



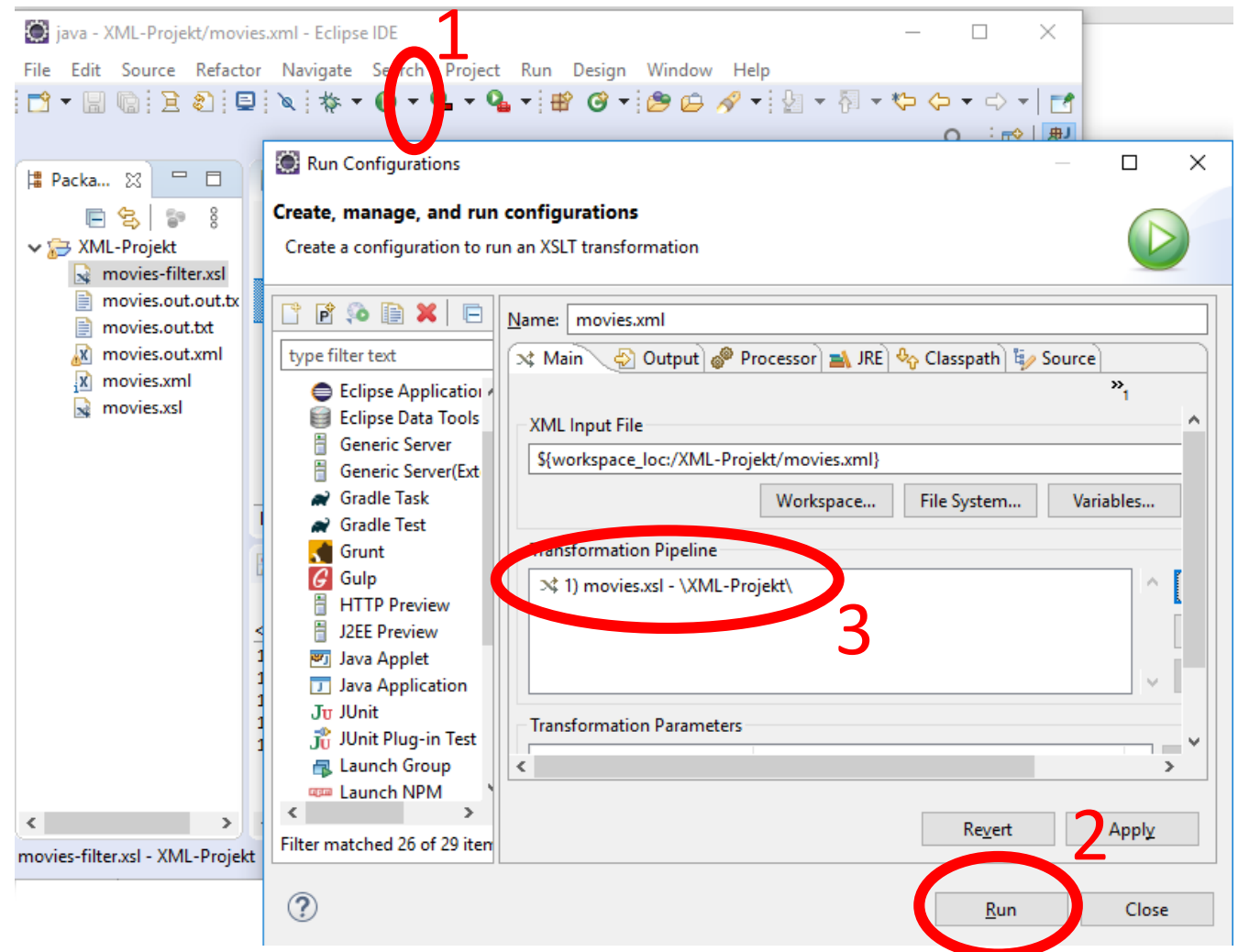
Eclipse Stylesheet Prozessor (2/4)

- XSL und XML-Datei zugleich auswählen (1) und den grünen Run-Button drücken (2).
- Alternativ: rechte Maustaste und „Runs as..“ -> „XSL-Transformation“ ausführen.
- Die Ausgabe wird als Datei erzeugt und angezeigt (3).



Eclipse Stylesheet Prozessor (3/4)

- Ein einmal ausgeführtes XSL erzeugt eine Run-Configuration.
- Diese wird wiederverwendet, neigt aber dazu fehlerhaft zu werden.
- Am besten nach dem ersten Start immer direkt über die Run-Configuration (2) starten.
- Kann über Auswahl beim Run-Button (1) angezeigt werden.
- Die „Transformation Pipeline“ (3) gibt an, in welcher Reihenfolge mehrere XSLT ausgeführt werden.
- Das Ergebnis einer Transformation wird automatisch als Eingabe der nächsten übertragen.



Eclipse Stylesheet Prozessor (4/4)

- Für die Aufgabe 6 sollen zwei Stylesheet-Transformationen hintereinander ausgeführt werden.
- Dazu muss eine Run-Configuration erstellt werden, mit einer entsprechenden „Transformation-Pipeline“.
- Mann kann eine Kopie einer existierenden Run-Configuration verwenden und anpassen.
- Ansonsten nicht vergessen bei „XML Input File“ die XML-Datei anzugeben.
- Mit „Add Files“ (1) kann eine weitere XSLT hinzugefügt werden.
- Mit den Pfeil-Tasten (2) können Sie noch verschoben werden.

