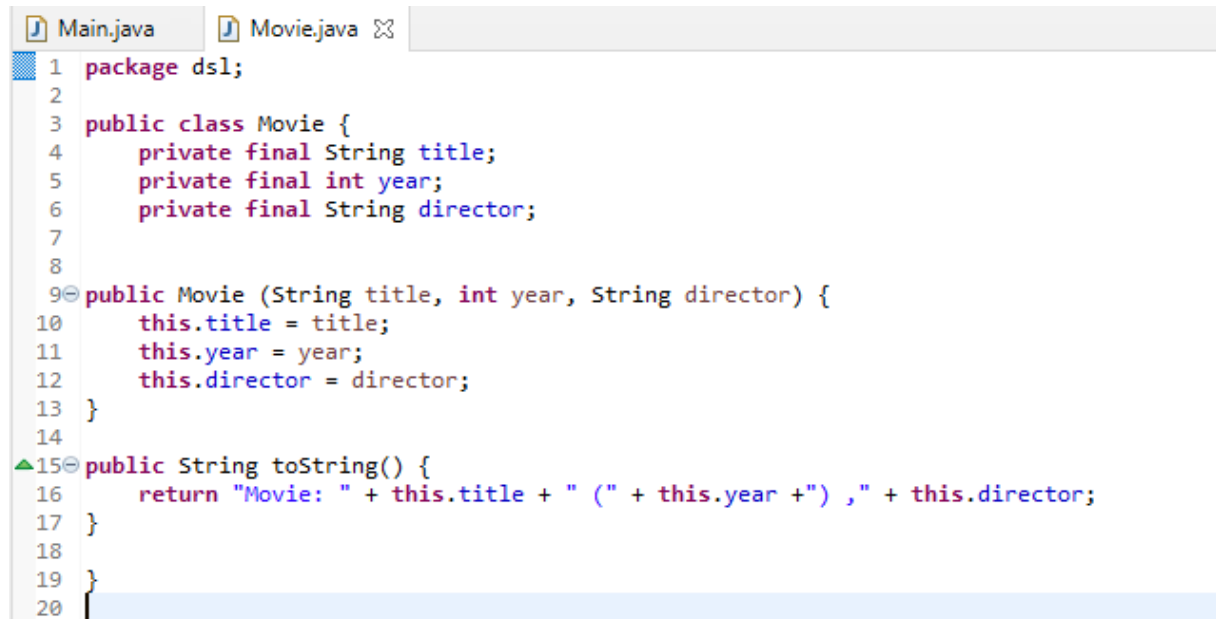


Domain Specific Languages

Für die Einsendeaufgabe „Domain Specific Languages“ wurde in der Entwicklungsumgebung *Eclipse* eine *DSL* am Beispiel einer Watchlist für Filme erstellt. Im Vordergrund der Anwendung steht eine geordnete Ausgabe von Meta-Daten der hinzugefügten Filme. Das Objekt *Movie* wird über die drei Metadaten **Title**, **Year** und **Director** beschrieben.



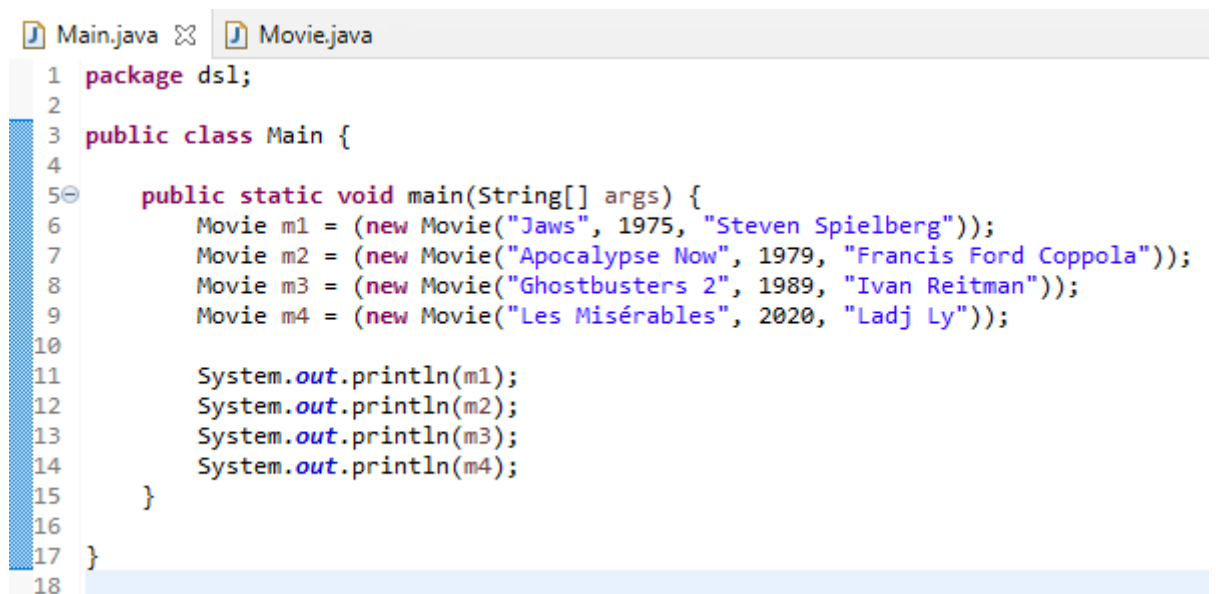
```

1 package dsl;
2
3 public class Movie {
4     private final String title;
5     private final int year;
6     private final String director;
7
8
9     public Movie (String title, int year, String director) {
10         this.title = title;
11         this.year = year;
12         this.director = director;
13     }
14
15     public String toString() {
16         return "Movie: " + this.title + " (" + this.year + ") ," + this.director;
17     }
18
19 }
20

```

Abbildung 1: Java *Fluent-Interface* für die Verkettung von Film Metadaten am Beispiel einer Watchlist

Durch das *Fluent-Interface* wird eine festgelegte „Grammatik“ vorgegeben, um die gewünschten Filme in der Watchlist abzulegen.



```

1 package dsl;
2
3 public class Main {
4
5     public static void main(String[] args) {
6         Movie m1 = (new Movie("Jaws", 1975, "Steven Spielberg"));
7         Movie m2 = (new Movie("Apocalypse Now", 1979, "Francis Ford Coppola"));
8         Movie m3 = (new Movie("Ghostbusters 2", 1989, "Ivan Reitman"));
9         Movie m4 = (new Movie("Les Misérables", 2020, "Ladj Ly"));
10
11         System.out.println(m1);
12         System.out.println(m2);
13         System.out.println(m3);
14         System.out.println(m4);
15     }
16
17 }
18

```

Abbildung 2: Java *Main*-Klasse für die Ausgabe der Watchlist

Werden in der Klasse *Main* nun Filme mit der im Objekt *Movie* angelegten Grammatik hinzugefügt, wird über die Konsole eine einheitliche Watchlist ausgegeben.

```
<terminated> Main [Java Application] C:\Users\morit\p2\pool\plugins\org.
Movie: Jaws (1975) ,Steven Spielberg
Movie: Apocalypse Now (1979) ,Francis Ford Coppola
Movie: Ghostbusters 2 (1989) ,Ivan Reitman
Movie: Les Misérables (2020) ,Ladj Ly
```

Abbildung 3: Ausgabe der Watchlist in der Konsole

Im nächsten Schritt wird über das Open-Source-Framework *Xtext* eine logische Grammatik für eine Watchlist erzeugt. Hierbei wurde die Funktionsweise des erstellten *Java Fluent-Interfaces* als Vorlage genommen.

```
1 grammar org.xtext.example.test.Test with org.eclipse.xtext.common.Terminals
2
3 generate test "http://www.xtext.org/example/test/Test"
4
5 Model:
6     action+=Action*;
7
8 Action:
9     addMovie | removeMovie;
10
11 addMovie:
12     'add' movie;
13
14 removeMovie:
15     'remove' movie;
16
17 movie:
18     STRING "("INT")";
19
```

Abbildung 4: *Xtext*-Datei mit einer festgelegten Grammatik zum Anlegen einer Watchlist

Nach erfolgreichem Aufbau der Grammatik in der *Test.xtext* Datei werden die *Xtext Artifacts* generiert, um die Grammatik für eine Watchlist zugänglich zu machen.

```
<terminated> GenerateTest.mwe2 [Mwe2 Launch] C:\Users\morit\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.f
0 [main] INFO text.xtext.generator.XtextGenerator - Initializing Xtext generator
8 [main] INFO lipse.emf.mwe.utils.StandaloneSetup - Adding generated EPackage 'org.ecl
147 [main] INFO lipse.emf.mwe.utils.StandaloneSetup - Registering project org.xtext.exam
147 [main] INFO lipse.emf.mwe.utils.StandaloneSetup - Registering project org.xtext.exam
148 [main] INFO lipse.emf.mwe.utils.StandaloneSetup - Registering project org.xtext.exam
148 [main] INFO lipse.emf.mwe.utils.StandaloneSetup - Registering project org.xtext.exam
148 [main] INFO lipse.emf.mwe.utils.StandaloneSetup - Registering project org.xtext.exam
156 [main] INFO lipse.emf.mwe.utils.StandaloneSetup - Using resourceSet registry. The re
376 [main] INFO clipse.emf.mwe.utils.GenModelHelper - Registered GenModel 'http://www.ec
381 [main] INFO clipse.emf.mwe.utils.GenModelHelper - Registered GenModel 'http://www.ec
394 [main] INFO clipse.emf.mwe.utils.GenModelHelper - Registered GenModel 'http://www.ec
395 [main] INFO clipse.emf.mwe.utils.GenModelHelper - Registered GenModel 'http://www.ec
578 [main] INFO text.xtext.generator.XtextGenerator - Generating org.xtext.example.test.
1234 [main] INFO nerator.ecore.EMFGeneratorFragment2 - Generating EMF model code
1260 [main] INFO clipse.emf.mwe.utils.GenModelHelper - Registered GenModel 'http://www.xt
3140 [main] INFO text.xtext.generator.XtextGenerator - Generating common infrastructure
3160 [main] INFO .emf.mwe2.runtime.workflow.Workflow - Done.
```

Abbildung 5: Konsolenausgabe nach Generierung der *Xtext Artifacts*



```

Watchlist.test
add "Jaws" (1975)
add "Apocalypse Now" (1976)
add "Pulp Fiction" (1994)

```

Abbildung 6: Watchlist.test-Datei unter Nutzung der generierten .xtext-Datei

Durch die in *xText* festgelegte Grammatik können der Watchlist neue Filme nach einem vorbestimmten Schema hinzugefügt werden. Der Befehl `add` fügt einen neuen Film hinzu, sofern dieser nach der Grammatik `STRING + (INT)` benannt wird. → Filmtitel + Erscheinungsjahr

Diese Grammatik kann ebenfalls im *Xtext Syntax Graph* nachvollzogen werden.

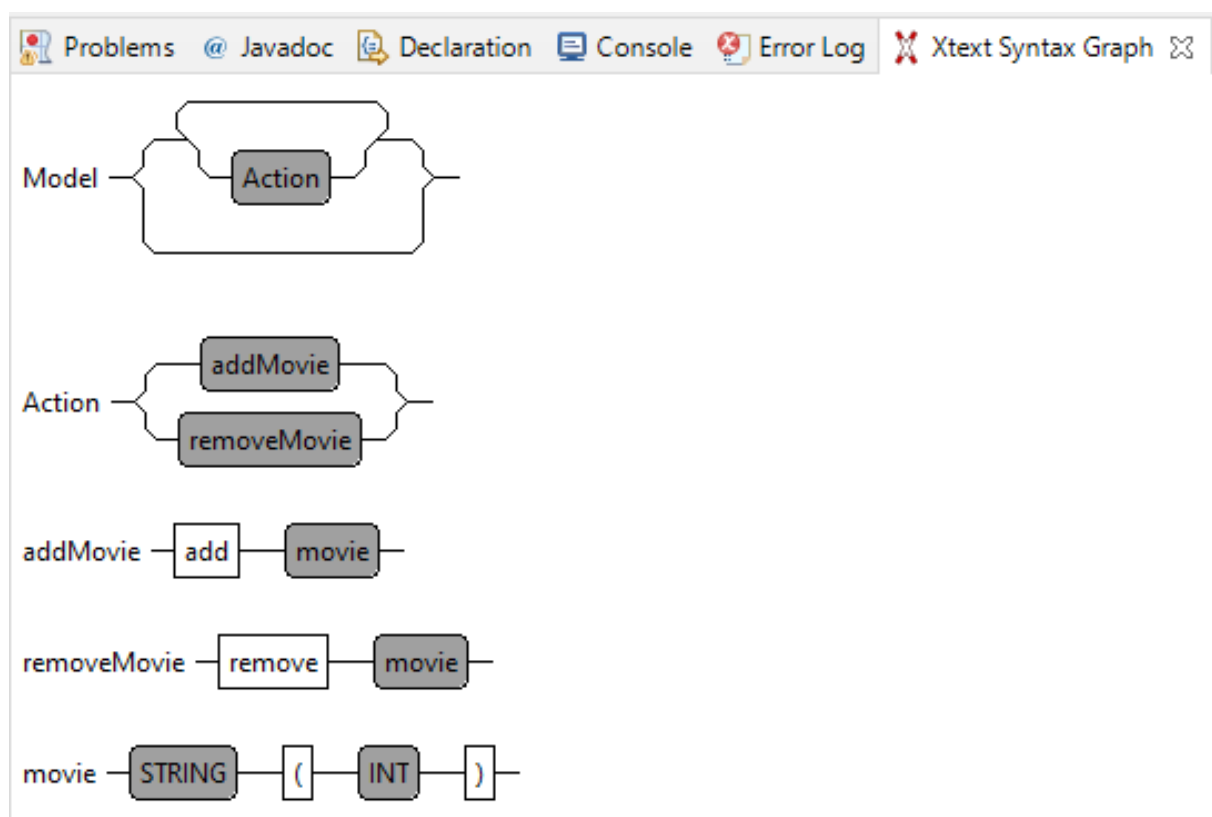


Abbildung 7: Xtext Syntax Graph stellt die Grammatik und Konzept der Watchlist dar