Context: A Movie List

In this chapter, we'll start with an example program and improve it little by little, without adding any new functionality. This important programming task is called refactoring.

> Our initial program is about recent Batman movies. The data comes under the form of an array of objects, with each object describing a movie.

```
const movieList = [
  {
    title: "Batman",
    year: 1989,
    director: "Tim Burton",
    imdbRating: 7.6
  },
  {
    title: "Batman Returns",
    year: 1992,
    director: "Tim Burton",
    imdbRating: 7.0
  },
  {
    title: "Batman Forever",
    year: 1995,
    director: "Joel Schumacher",
    imdbRating: 5.4
  },
  {
    title: "Batman & Robin",
    year: 1997,
    director: "Joel Schumacher",
    imdbRating: 3.7
  },
  {
    title: "Batman Begins",
    year: 2005,
    director: "Christopher Nolan",
    imdbRating: 8.3
  },
  {
    title: "The Dark Knight",
    year: 2008,
    director: "Christopher Nolan",
    imdbRating: 9.0
  },
  {
    title: "The Dark Knight Rises",
    year: 2012,
    director: "Christopher Nolan",
    imdbRating: 8.5
```

```
    imdbRating: 8.5
  }
];
```

And here is the rest of the program that uses this data to show some results about the movies. Check it out, it should be pretty self-explanatory.

```
// Get movie titles
const titles = [];
for (const movie of movieList) {
  titles.push(movie.title);
}
console.log(titles);

// Count movies by Christopher Nolan
const nolanMovieList = [];
for (const movie of movieList) {
  if (movie.director === "Christopher Nolan") {
    nolanMovieList.push(movie);
  }
}
console.log(nolanMovieList.length);

// Get titles of movies with an IMDB rating greater or equal to 7.5
const bestTitles = [];
for (const movie of movieList) {
  if (movie.imdbRating >= 7.5) {
    bestTitles.push(movie.title);
  }
}
console.log(bestTitles);

// Compute average movie rating of Christopher Nolan's movies
let ratingSum = 0;
let averageRating = 0;
for (const movie of nolanMovieList) {
  ratingSum += movie.imdbRating;
}
averageRating = ratingSum / nolanMovieList.length;
console.log(averageRating);
```