chem-bla-ics

Cleaner CDK Code #1: List and the for-each loop

Egon Willighagen

Published March 7, 2010

Citation
Willighagen, E. (n.d.). In *chem-bla-ics*. chem-bla-ics. https://doi.org/10.59350/jx0j3-s9893

Keywords
Cdk, Java

Copyright
Copyright © Egon Willighagen 2010. Distributed under the terms of the Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

In a desperate attempt to force me to write on my CDK code snippet book, I'm going to write some code tips to create clear code. Hopefully, this is useful for people writing patches and reviewers alike, too.

Use List instead of the untyped List

Quite some time ago, the Java language introduced typed lists. These lists can contain only objects of a particular type, which is a very common use case. Indeed, the CDK has quite a few lists that are strongly typed. Typing the list prevents you from accidentally adding something of the wrong type, but also reduced the amount of casting, so that your code becomes cleaner.

Instead of:

```
List atoms = atomContainer.atoms();
for (int i=0; i<atoms.size(); i++) {
  IAtom atom = (IAtom)atoms.get(i);
  // .. do something
}
```

you can do:

```
List<IAtom> atoms = atomContainer.atoms();
for (int i=0; i<atoms.size(); i++) {
  IAtom atom = atoms.get(i);
  // .. do something
}
```

If you do not need the index, use a for-each loop

When iterating over atoms in a list, you sometimes need to know the index, for example, to compare the IAtom with that at the same position in another list. However, when this is not needed, you can use the Java for-each loop instead. This will further simplify the above code to:

```
for (IAtom atom : atomContainer.atoms()) {
  // .. do something
}
```