Java Record

CS 1410 2024
Cody Henrichsen

1

Why

▷ Much of the boilerplate code of a Java model class is unnecessary to write
▷ Your data source is already defined and you DO NOT want to change its internal state
▷ You have JSON or CSV data available
▷ No inheritance requirements

2

Record Data

▷ Header specifies constructor and data members at once
▷ Maintains private data as it should
▷ Accessor methods are just the name of the data member followed by parens!
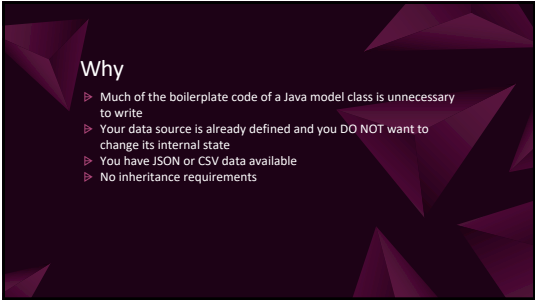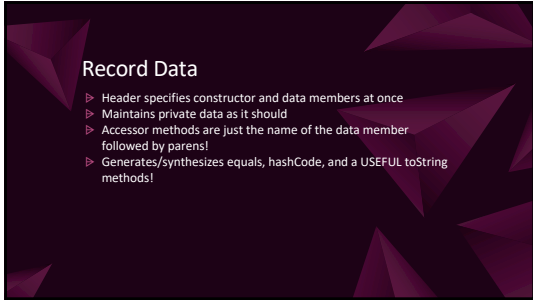▷ Generates/synthesizes equals, hashCode, and a USEFUL toString methods!

3

Definition

▷ A record starts with a package statement like a class
▷ Then any imports needed as well
▷ The header uses the keywords public record followed by the Type and each data member separated by commas
▷ The body is simply an open/close set of squiggles!!

4

Record Example

```
package data.model;

public record InternetCat(String [] tags, String mimetype, int size, String createdAt, String editedAt, String _id)
{
}
```

5

Use

▷ Records are instantiated just like an object via the new keyword followed by a constructor call!
▷ Internal state is accessed by calling the appropriate data member accessor method AKA the name followed by parens.
▷ The toString method specifies the Type followed by a set of [] with each data member then an = and the toString of the associated value each separated with commas

6

7

8