AARHUS UNIVERSITET

Software Engineering and Architecture

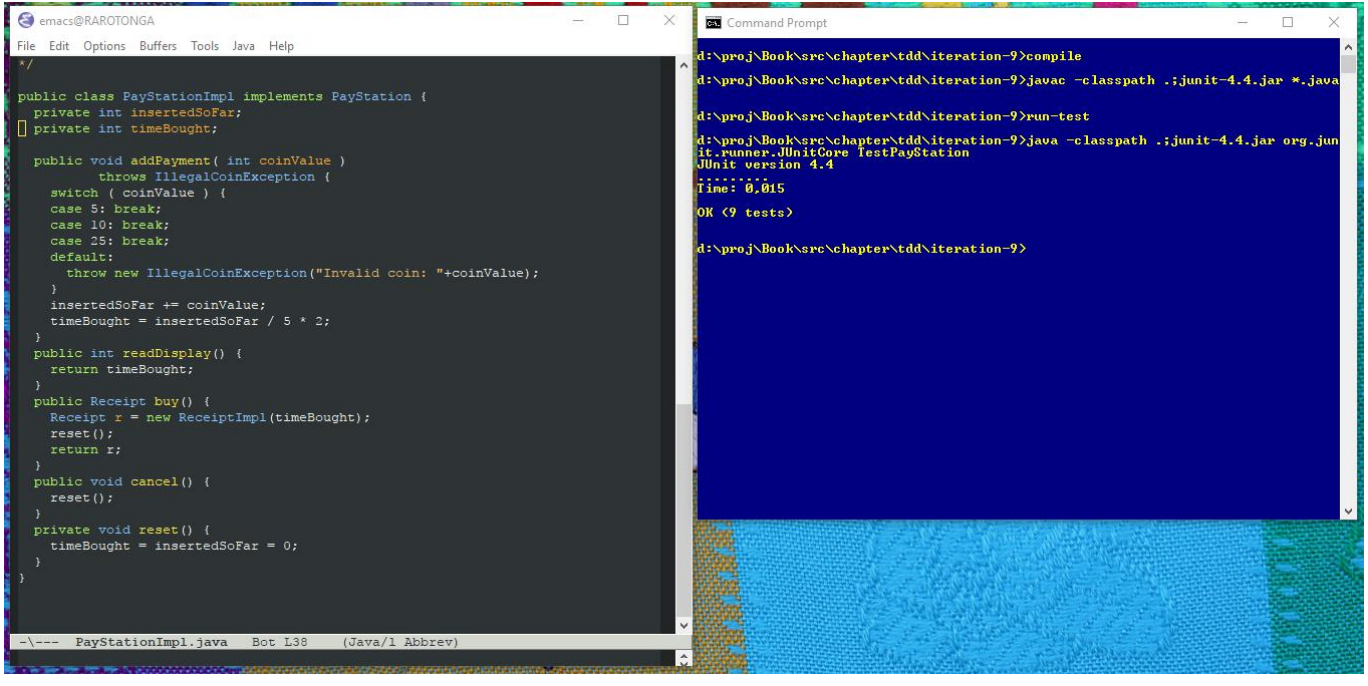
Integrated Development Environment

IntelliJ

AARHUS UNIVERSITET

Looking Back in Time

- All you actually need is
  - An editor
  - A compiler

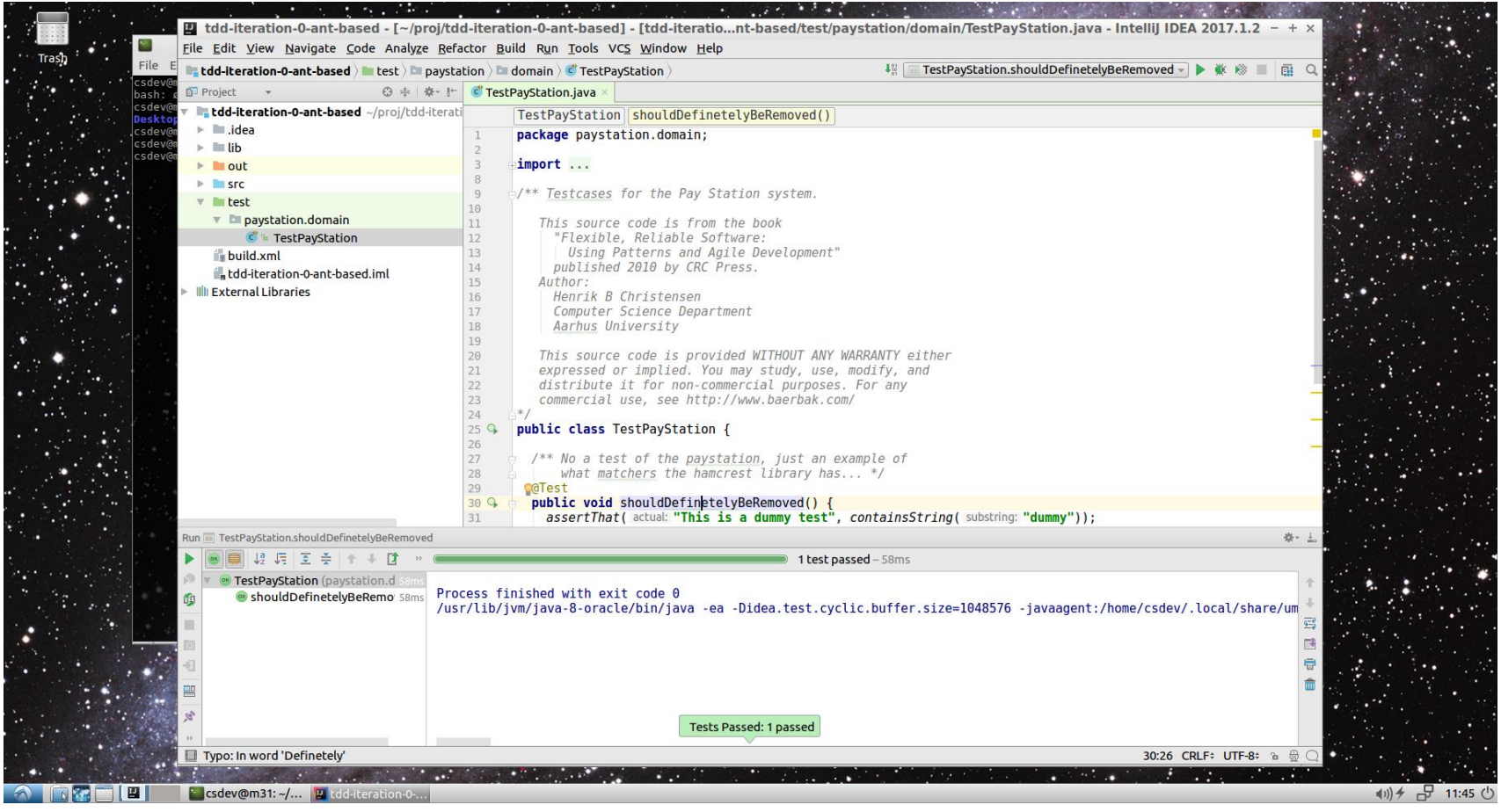


AARHUS UNIVERSITET

However…

- Modern IDE's have a deep understanding of the programming language that helps a lot in development
  - Context sensitive help and **completion**
  - **Refactoring** support, especially *renaming identifiers*
  - Integration with **SCM tools**
  - Strong **browsing** of large package/source file hierarchies
- And of course things that *others* ☺ use
  - Debugger
- (Why do I almost never use a debugger?
  - Because I take very small steps!
    - If I cannot figure a defect out, a few *temporary* System.out does it ☺

IntelliJ


tdd-iteration-0-ant-based - [~/proj/tdd-iteration-0-ant-based] - [tdd-iteratio...nt-based/test/paystation/domain/TestPayStation.java - IntelliJ IDEA 2017.1.2
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
TestPayStation.java
package paystation.domain;
import ...
public class TestPayStation {
@Test
public void shouldDefinetelyBeRemoved() {
assertThat(actual: "This is a dummy test", containsString(substring: "dummy"));
1 test passed – 58ms
Process finished with exit code 0
Tests Passed: 1 passed
Typo: In word 'Definetely'

AARHUS UNIVERSITET

Highlights in *Power Coding*

Past the ‘Run All Tests’ right-click

Solution First Programming

- ‘Quick Fix’ programming
  - *Write immediately what you want*
  - *Let IDE fill in the details under your guidance **‘Alt+Enter’***

```
@Test
public void shouldShow2MinFor5CentEntered() throws IllegalCoinException {
  assertThat(ps.readDisplay(), is( value: 2));
}
```

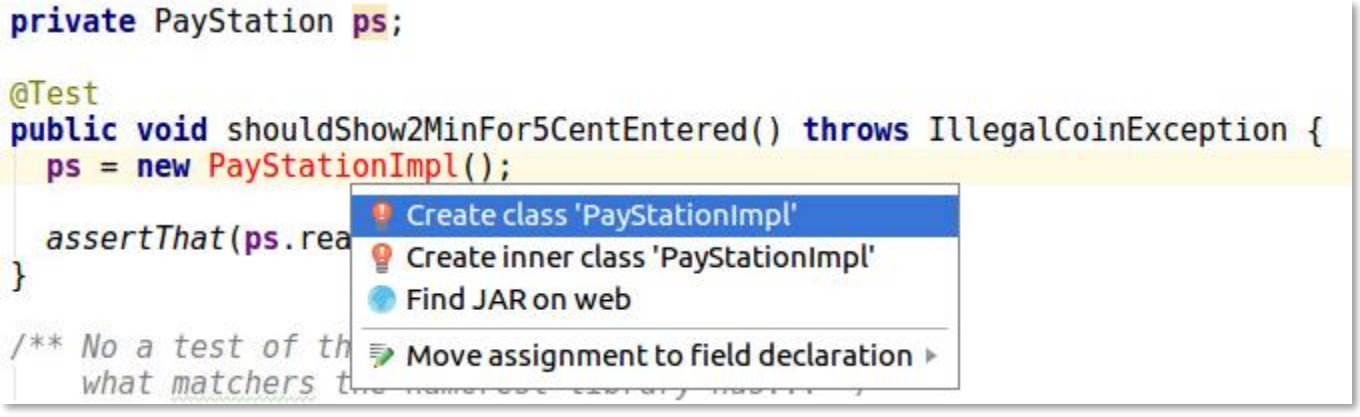
```
@Test
public void shouldShow2MinFor5CentEntered() throws IllegalCoinExcep
  assertThat(ps.readDisplay(), is( value: 2));
}

/** No a test
    what match
@Test
public void sh
  assertThat(
```

Create local variable 'ps'
Create field 'ps'
Create parameter 'ps'
Import static constant 'sun.plugin.services.PlatformService.ps'
Rename reference

AARHUS UNIVERSITET

Never Write Classes

- IDE does it quicker **‘Alt-Enter’**

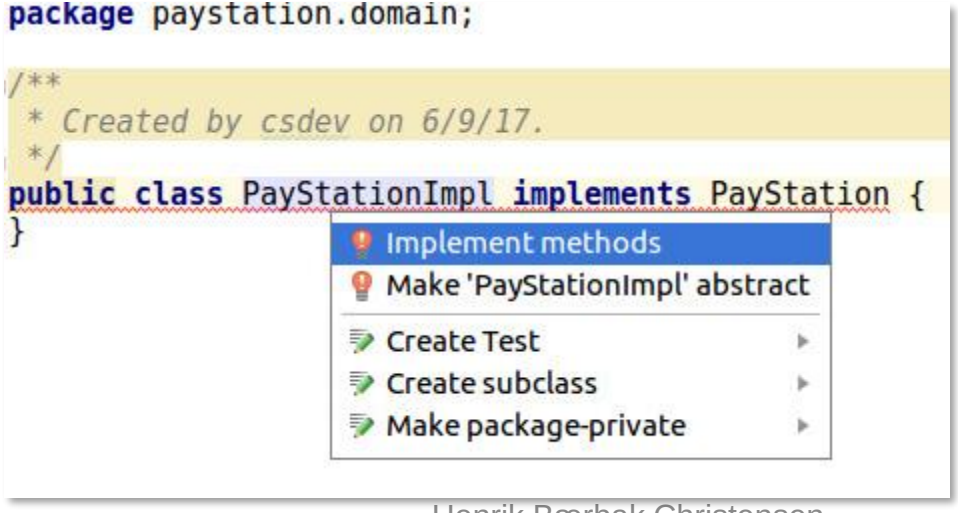


AARHUS UNIVERSITET

‘Temporary Stub’

- All classes in TDD starts as ‘temporary stubs’
  - All methods are ‘null’ implementations:
    - Empty or ‘return null;’ or ‘return 0;’ etc.

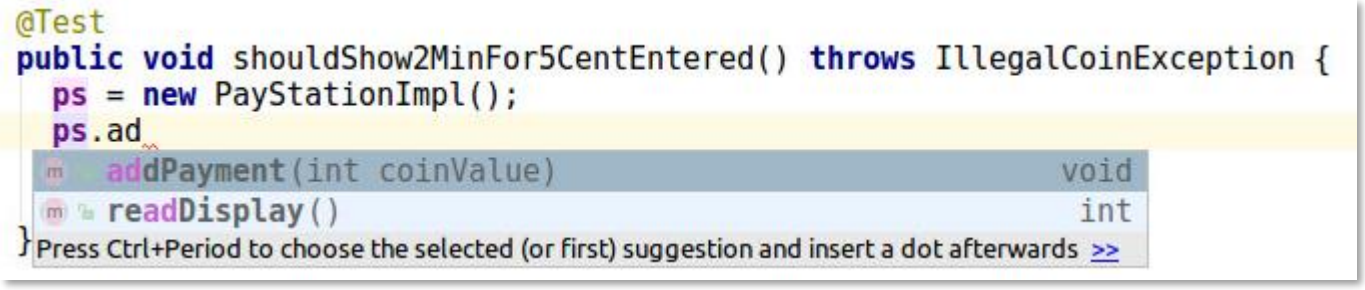
- ‘**Alt-Enter**’ does it faster than you



AARHUS UNIVERSITET

Code Completion

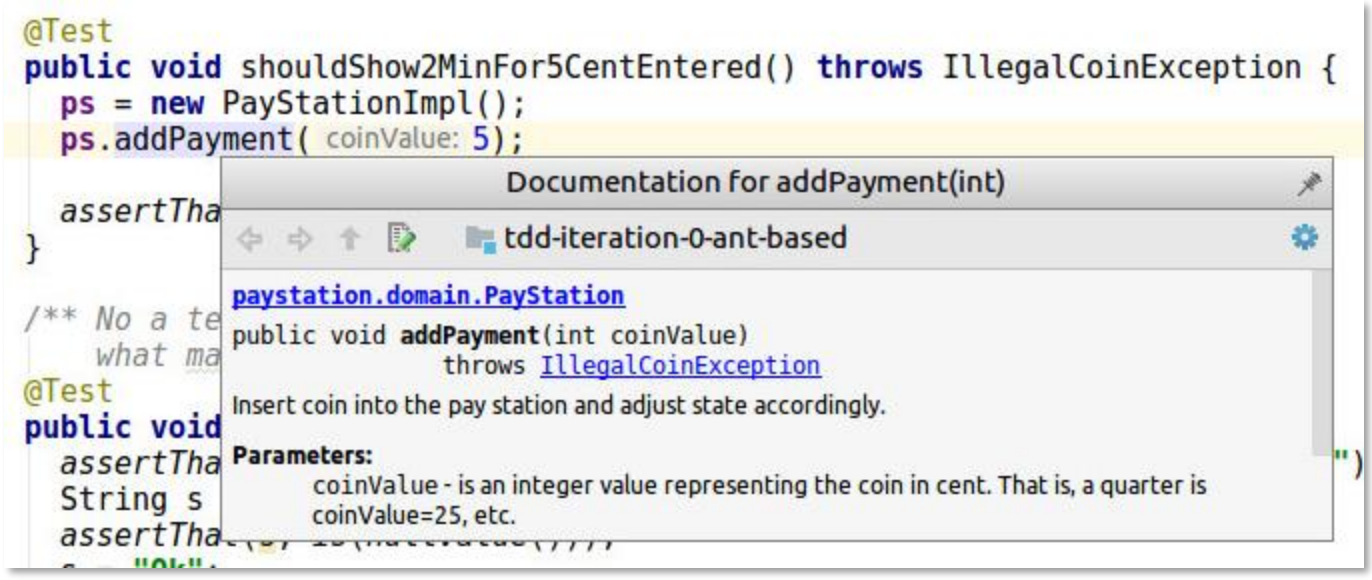
- Never write method names in full **‘Ctrl-space’**



AARHUS UNIVERSITET

Review the Docs

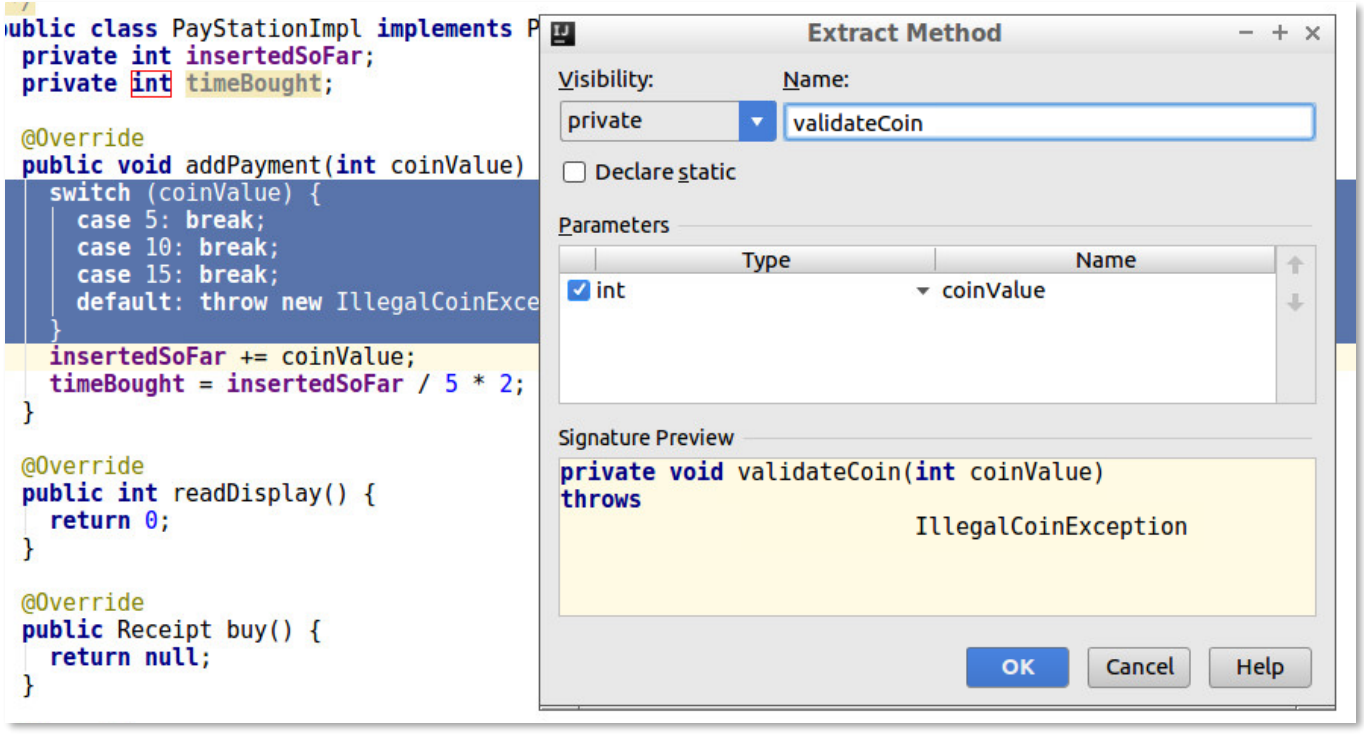
- Hit the ‘Ctrl-Q’ to see the JavaDocs



AARHUS UNIVERSITET

Refactor Your Code

- We will look at that later, but – example
  - Extract Method **‘Ctrl-Alt-M’**



After…

- Hit the button and you get…

```
@Override
public void addPayment(int coinValue) throws IllegalCoinException {
  validateCoin(coinValue);
  insertedSoFar += coinValue;
  timeBought = insertedSoFar / 5 * 2;
}

private void validateCoin(int coinValue) throws IllegalCoinException {
  switch (coinValue) {
    case 5: break;
    case 10: break;
    case 15: break;
    default: throw new IllegalCoinException("Invalid coin, value is "+coinValue);
  }
}
```

And…

- IDE's automatically notice that you add source files and can keep the SCM system up-to-date!
  - You cannot compile your friends code after pulling it from a git repo if one file was not added to the repo!

AARHUS UNIVERSITET

Summary

- You can code in notepad/vi/gedit and the Java compiler
- However modern IDEs help you code *much faster*
- Drawback
  - Steep learning curve ☹
    - Find a ‘cheat sheet’ /default keymap on the web!
  - At the exam, there is no IntelliJ!