

<epam> | hws

HOW TO WORK WITH OTHERS CODE

Peter Veres
Senior Software Engineer
Peter_Veres2@epam.com

Why to read others code?

- Writing code is a fun
- Reading code is a hard work



- BUT
 - Time-to-time you have to join a team
 - Other people's code could even be your own
 - Your own code begins to look increasingly strange mere hours after you've finished writing it

Motivation



- Why should we read others code?
 - Skill of all the masters is right there, embedded in the code they have written
 - Being good at reading code is important to your survival
 - There will always be large chunks of code you had no hand in
 - Give it a year and other people's code could even be your own

Collective code ownership

- The codebase is owned by the entire team and anyone may make changes anywhere



A photograph of a busy city street at sunset. The sun is low in the sky, creating a warm, golden glow. In the foreground, many people are walking across the street. On the left, a large, ornate building with a balcony and a Union Jack flag is visible. A street sign on the building reads "GENT STREET W1". In the background, there are more buildings, some with scaffolding, and a red flag with the word "Superdry" on it. A Starbucks logo is visible on a building to the right. A large blue text overlay is positioned in the center of the image.

How to work with others code

WHAT CAN WE DO FOR US?

Interact

- Find lead developers, interact with them (talk, email, skype, etc.)



Read the code

- Build and run your program
- Find the high level logic
- Understand all the constructs
- Do a couple of random deep-dives



Read the code

- Read some tests
- Write your test code



Read the code

- Refactor some parts of the code
- Extract bits into a standalone program



Read the code

- Get yourself a code reading buddy
- Ask

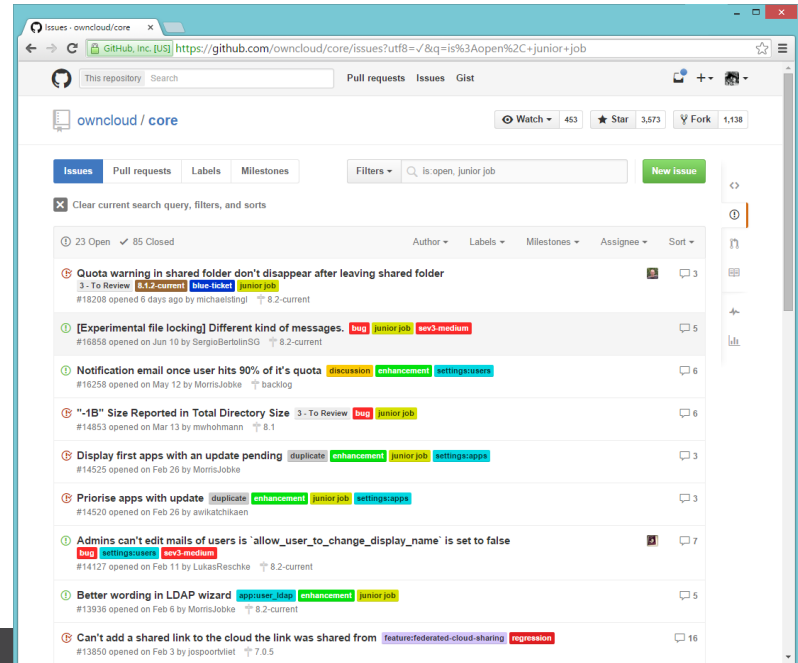


Read the code

- If you're new to a significant codebase, you will never get across it instantly, or even quickly
- Alternatively you can avoid reading code and always be the guy looking for someone to explain something to you

Fix bugs

- Easy way to start out is fixing the bugs targeted at newcomers
- These bugs won't be highly technical, but you'll be able to ease in to the project, building your confidence and knowledge



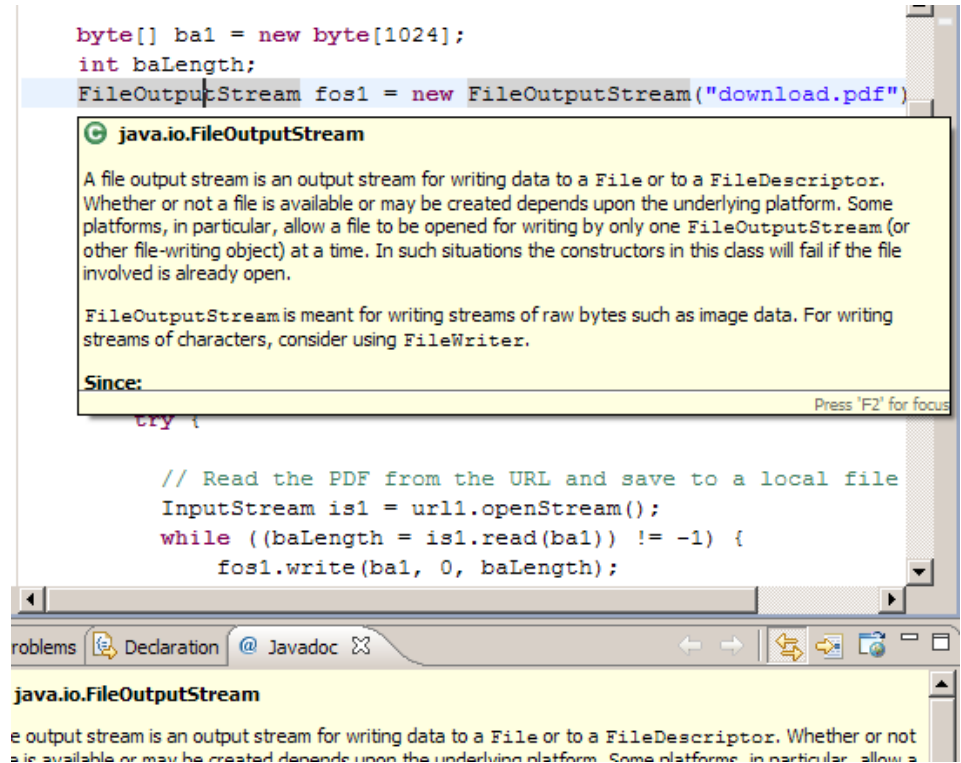
Find resources

- Ideas to look for
 - Mailing list archives
 - Project or company wiki
 - Project documentation
 - Version control history
 - Commit messages



Use a good IDE

- Use the features your IDE provides
 - Check references, navigation
 - Read code documentation
 - Run/debug the tests and the app
 - Visualize code
 - Refactor code
 - Etc.



Be considerate

- To reading others code is similar to code review
- Try to understand development choices even if you don't agree with them
- Staying humble and a good partner
- Learning curve can be from 3 weeks to 3
- Similar projects are written in a similar manner
- Think of those coming after you

Read and learn

- The more we learn, the more we can grow
- Others have come before us and the same mistakes
- So many of the best developers are also authors/bloggers
- Open source projects can be used as a coding reference

- Some books to mention
 - Clean Code (Robert C. Martin)
 - Refactoring (Martin Fowler)
 - Design Patterns (Gang of Four)
 - Effective Java (Joshua Bloch)
 - etc.



A photograph of a busy city street at sunset. The sun is low in the sky, creating a bright orange glow and long shadows. The street is lined with buildings, including a prominent white stone building on the left with a balcony and a flag. A red traffic light is visible. In the foreground, many people are walking on the sidewalk. A large, semi-transparent blue banner with white text is overlaid across the middle of the image.

How to work with others code

WHAT CAN WE DO FOR THE OTHERS?

What can we do?

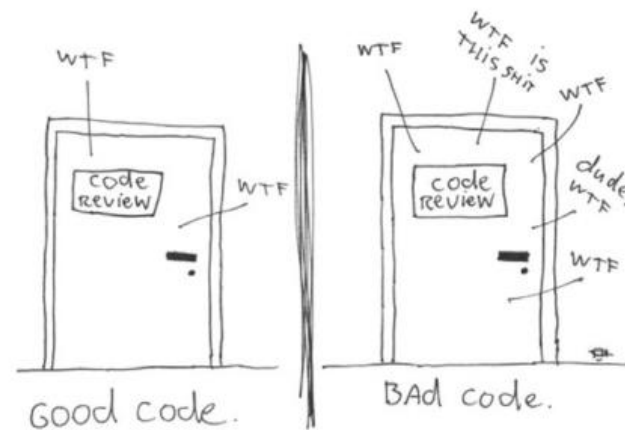
- Who are you writing code for, anyway?
- What's wrong with code that just works?



Clean Code

- Clean code is code that is easy to understand and easy to change
- Quality of the code could be measured

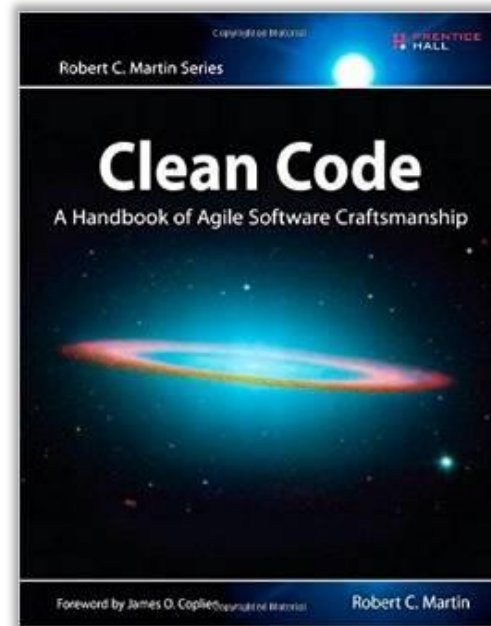
The ONLY VALID MEASUREMENT OF CODE QUALITY: WTFs/MINUTE



(c) 2008 Focus Shift

Clean Code

- Choose your names well
- Use common vocabulary
- Use meaningful names
- Methods should be small, and only doing one thing
- Don't comment bad code – rewrite it
- Format your code properly
- Apply code review
- etc.



Clean Code examples - names

```
public List<int[]> getThem() {  
    List<int[]> list1 = new ArrayList<int[]>();  
    for (int[] x : theList)  
        if (x[0] == 4)  
            list1.add(x);  
    return list1;  
}
```

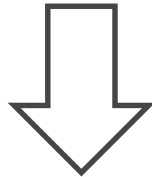
```
public List<Cell> getFlaggedCells() {  
    List<Cell> flaggedCells = new ArrayList<Cell>();  
    for (Cell cell : gameBoard)  
        if (cell.isFlagged())  
            flaggedCells.add(cell);  
    return flaggedCells;  
}
```

Clean Code examples - comments

```
/**  
 * Always returns true.  
 */  
public boolean isAvailable() {  
    return false;  
}
```

Clean code examples - comments

```
// Check to see if the employee is eligible for full benefits  
if ((employee.flags & HOURLY_FLAG) && (employee.age > 65))
```



```
if (employee.isEligibleForFullBenefits())
```


<epam> | hws

**THANKS FOR YOUR
KIND ATTENTION**