1. Introduction to Java Language

3. Large Project Specific

Large Project Features

- Sophisticated functionality
- Large code libraries
- Large team & employee turnover
- Different roles of team members
- Long life cycle

Sophisticated functionality

- Several hundreds or even thousands application’s functions
- Large user manual
- Sometimes sophisticated algorithms

Large code libraries

- An application contains 20 – 100 class libraries
- A class library consists of 20 – 70 classes
- A class contains 200 – 2000 code lines
- So, application’s code =

  100 000 – 20 000 000 code lines

Large team & employee turnover

- Several sub teams in different cities and/or countries
- 5-15 (up to 50-70) developers in each sub team
- 10-20% of natural turnover in a team
- Step by step shifting of responsibilities to Ukraine

Different roles of team members

- Developers + team leader
- Architects, algorithm designers
- Testers
- Business analysts
- Technical writers

Long life cycle

- 5-20 years of development
- 5 – 15 application versions (usually containing sub versions and modifications)
- Real examples:
  - Project 1 - 1996 – 2007
  - Project 2 - 2001 – 2010
  - Project 3 - 1989 – till now

Project Docs

- User manual
- Program source code

That’s All!

Work with Code

- Creation
- Maintenance (80% of lifecycle)
- Review
- Bug Fixing

Code Quality

- Low code quality increases losses in interactions between team members

Code Convensions

- Code should be **well documented** and **standardized**

- **Code should be as simple as possible**

Declarations

- One declaration per line is recommended since it encourages commenting
- Put declarations only at the beginning of blocks
- Try to initialize local variables where they’re declared.

Class Declarations

- No space between a method name and the parenthesis
- Open brace “{” appears at the end of the same line as the declaration statement
- Closing brace “}” starts a line by itself indented to match its corresponding opening statement, except when it is a null statement the “}” should appear immediately after the “{“
- Methods are separated by a blank line

Statements

- Each line should contain at most one statement.
- A while statement should have the following form:

```
while (condition) {
      statements;
 }
```

Compound Statements

- The enclosed statements should be indented one more level than the compound statement
- The opening brace should be at the end of the line that begins the compound statement; the closing brace should begin a line and be indented to the beginning of the compound statement

Naming Conventions

- Class names should be nouns, in mixed case with the first letter of each internal word capitalized.
- Methods should be verbs, in mixed case with the first letter lowercase, with the first letter of each internal word capitalized.

How to Study Code Conventions

- Step by step studying
- Wide use code review
- Use possibilities of IDE text editor
  - – You can use Eclipse to format code accordingly the conventions: right click in the text editor and select Source > Format menu item.