**JAVA Coding Standards**

# Use Proper Naming Conventions

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Conventions** | **Type** | **Example** |
| Classes | Should start with uppercase letter | Noun | String - Predefined  StringBuffer - Predefined  accounts - User defined classes  Accounts - User defined classes |
| Interface | First letter in uppercase | Adjective | Runnable  Serializable  Comparable |
| Methods | First letter in lowercase | verbs  or  verbs + Noun | print()  eat()  run()  getName  setName |
| Variables | First letter in lowercase  First letter in second word Start with uppercase | Noun | name  age  salary  mobileNumber |
| Constants | All letters in uppercase |  | MAX\_VALUE  MAX\_PRIORITY  MIN\_PRIORITY  PI |

# Using Comments

* Using comments highly useful coding element that can elevate the readability of a code to new heights. And veteran programmers know it the best.
* Code comments become a norm when programmers have to work with a team. They enable programmers to define, explain, notify and suggest the purpose behind specific lines of code.
* Comments should not be used to explain code. If the logic is not intuitive, it should be rewritten. Use comments to answer a question that the code can’t.
* Another way to state it is that the comment should explain the “why” versus “what” it does.
* For example, the following code converts temperature values from Celsius to Fahrenheit.

class Main {

public static void main(String args[])

{

int c = 35; int f = 0;

f = ((9\*c))/5+32; //Converts Celsius to Fahrenheit

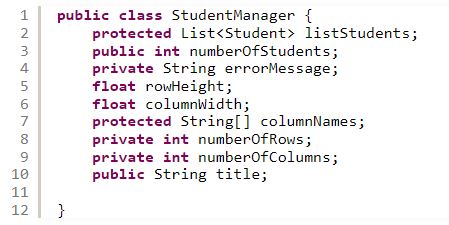
System.out.print(f);

    }

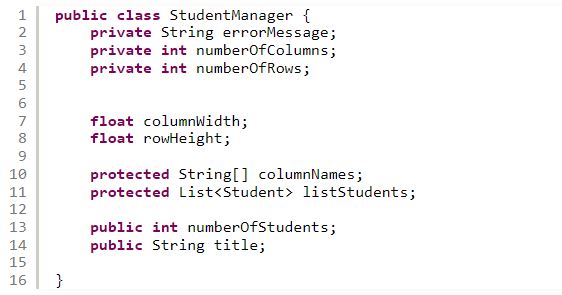
}

# Ordering Class members by scope

* + Private > default > protected > public

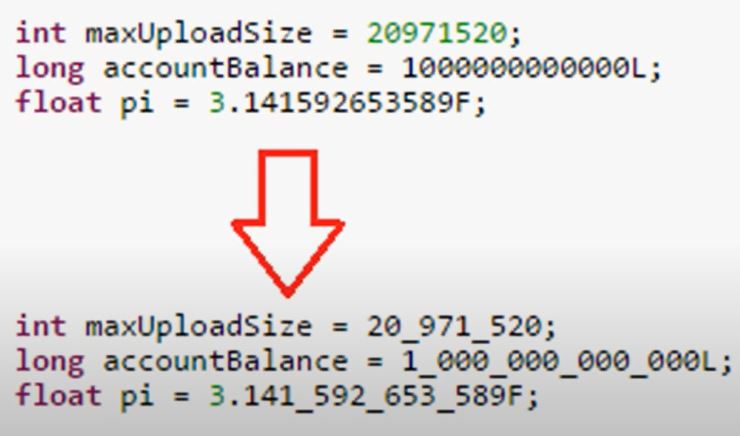


BAD



GOOD

# Using underscores in Numeric Literals

* + Use underscores (\_) in lengthy numeric literals => more for readability

# Exception Handling

## Do not Leave Catch Blocks Vacant

* + When run a segment if we suspect it had error we use to try, catch.

try{

}catch{

}

Catch{

}

Finally{

}

* + The try…. catch couple is an essential Java utility that programmers use for handling exceptions.
  + The try block encapsulates the dubious code that might give an exception, while the catch block houses the definition of the exception.
  + The consensus among experienced programmers is that the catch block should not be left empty unless needed. This is because excluding any explanation for an exception in the catch block can prevent a bad piece of code from getting addressed. This, in turn, makes rectifying the code even more difficult.

