

# Design Recipe

June 20, 2005

## 1. Data Analysis & Design

- Determine what information is relevant to the problem
- Design a data representation of this data; represent this graphically
- Translate the graphical design into Java

## 2. Purpose & Header

- Construct a sentence describing what the program does; place it in a comment (Hint: use 'this' in the sentence")
- Name the program and write it as a method inside the class

## 3. Method Examples

- Create examples for the method and demonstrate what the program should do in these cases.
- Hint: examples should reflect different outcomes where applicable.

## 4. Body - Template step

- Record what pieces of data are available within the class

## 5. Body

- Express what the program does in code.

## 6. Test

- Ascertain that the program does what we expected, using the method examples.
- If an example fails, examine the example and the program body to reassess.

Problem: Is the student old enough to vote?

```
-----  
| Student |  
|-----|  
| int age |  
-----
```

```
class Student {  
  
    int age;  
  
    Student(int age) {  
        this.age = age;  
    }  
    //new Student(3)  
    //new Student(21)  
  
    //To determine if this student is over 18  
    boolean oldEnoughToVote() {  
        // ... this.age ...  
        return this.age >= 18;  
    }  
    //new Student(3).oldEnoughToVote() == false  
    //new Student(21).oldEnoughToVote() == true  
    //new Student(18).oldEnoughToVote() == true  
  
}
```