

# SE 333/433 Software Testing & Quality Assurance

Assignment Project Exam Help

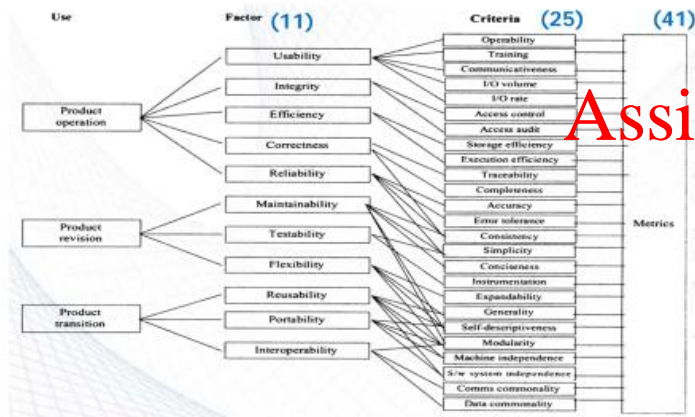
<https://powcoder.com>

Code-smells and bad-design practices

Add WeChat powcoder

# Last Week

## McCall's Model



## Chidamber and Kemerer OO Metrics (CK - 1994)

- Weighted methods per class (MWC)
- Depth of inheritance tree (DIT)
- Number of children (NOC)
- Coupling between object classes (CBO)
- Response for class (RFC)
- Lack of cohesion metric (LCOM)

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

28

## McCall's quality factors



# Functional vs Non-Functional Requirements

## Functional Requirements



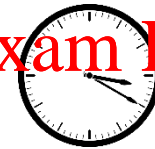
Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Functionality  
of the program

## Non-Functional Requirements



Execution Time



Size



Battery



Memory

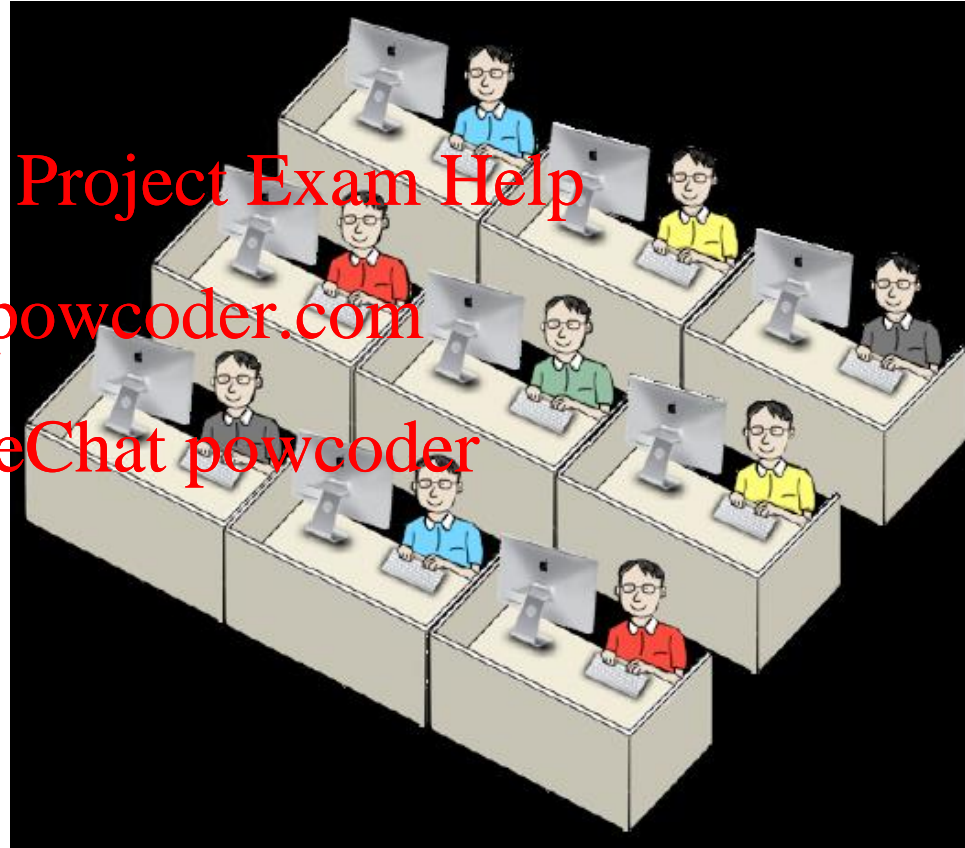


- Text chat
  - Friends list
- 

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder





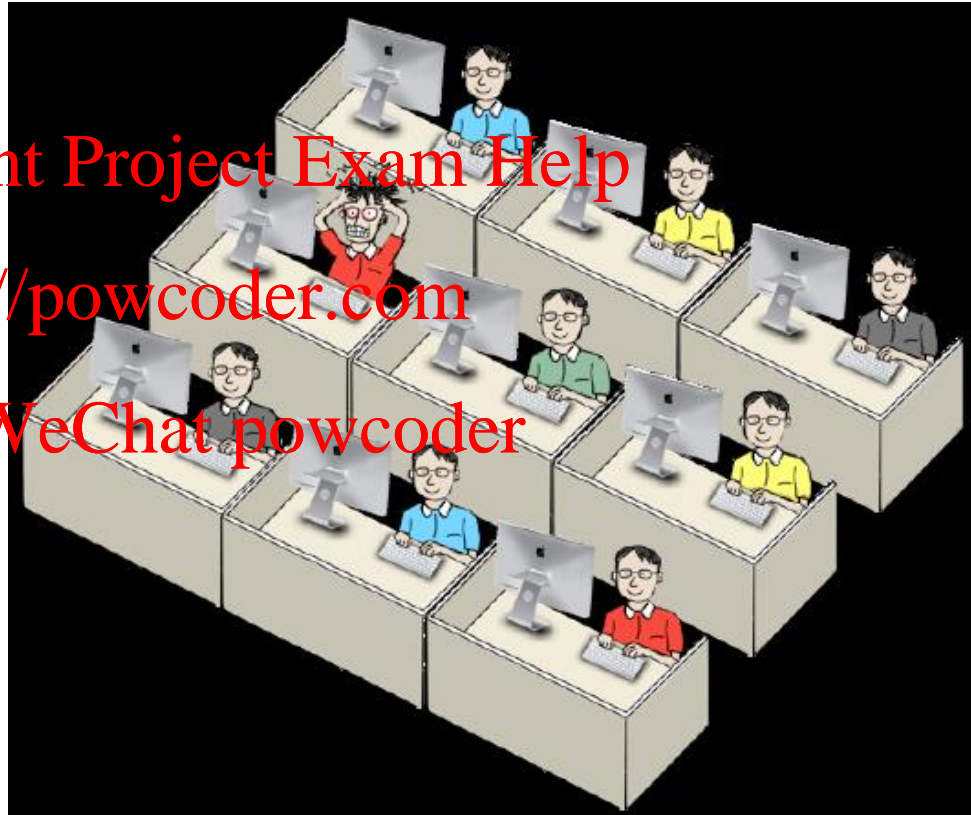
- Text chat
- Friends list

- Voice Communication

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder





- Text chat
- Friends list

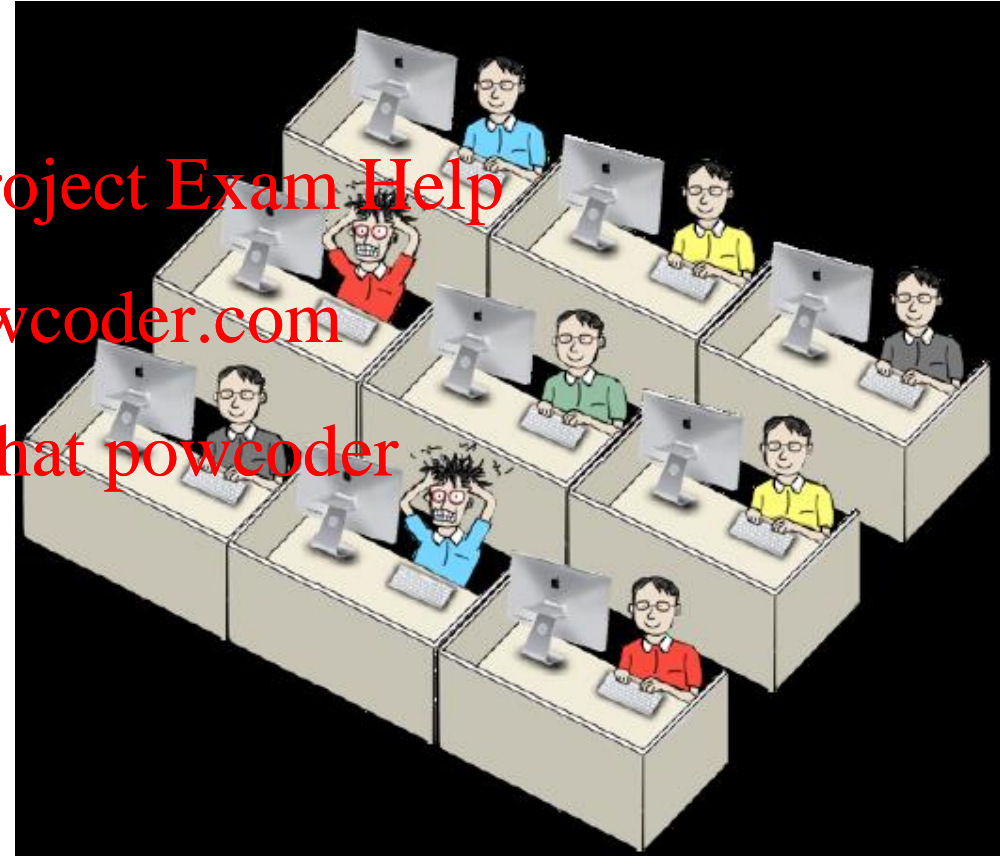
- Voice Communication



Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder







- Text chat
- Friends list

- Voice Communication



- Battery consumption

- Execution time

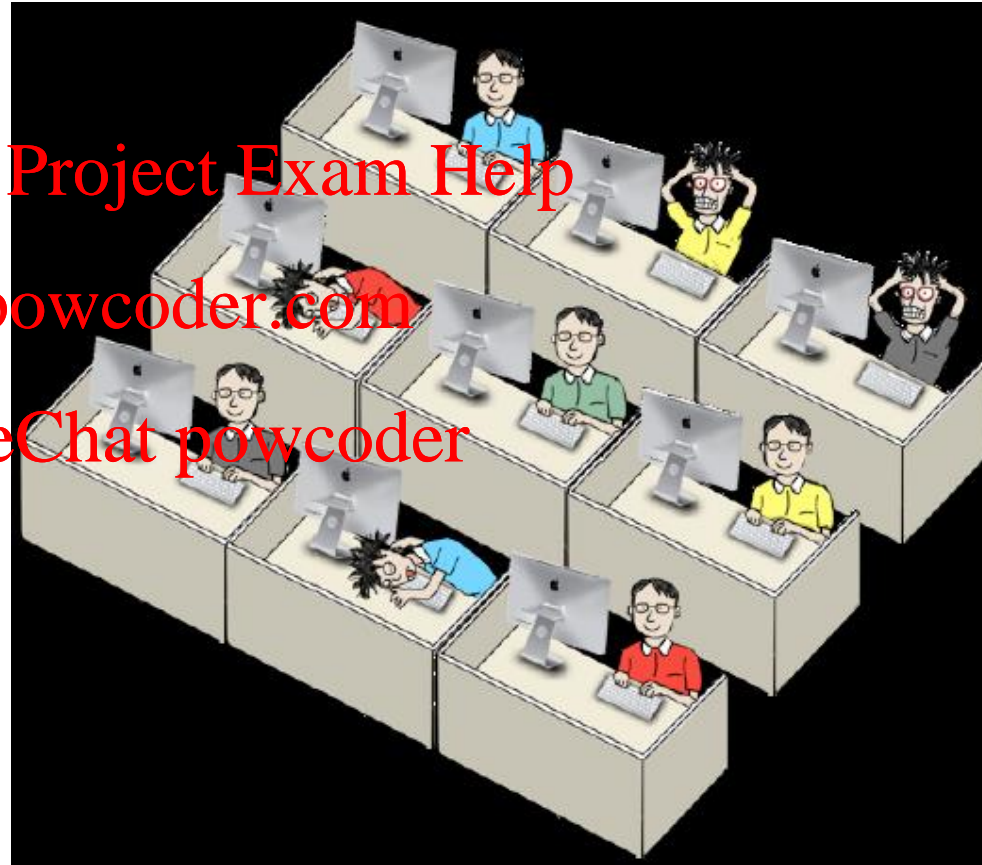


- Size

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder





- Text chat
- Friends list



- Voice Communication



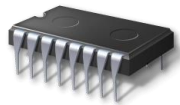
-Execution time



-Battery consumption



-Size



-Memory leak

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder





## Functional Requirements



humans have to define these

## Non-Functional Requirements



a machine can optimize these

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

## Functional Requirements



humans have to define these

## Non-Functional Requirements



a machine can optimize these

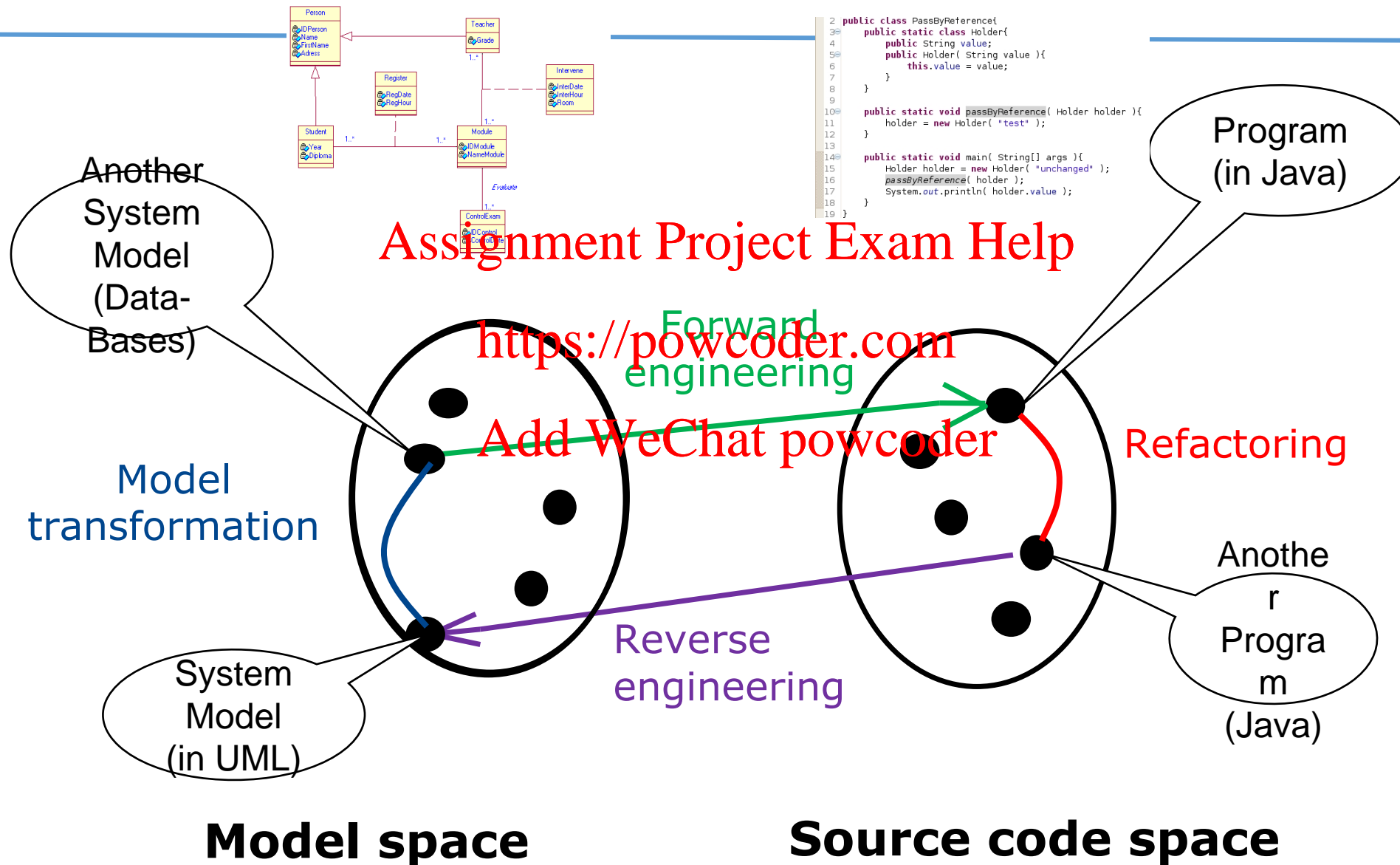
Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

## Assignment Project Exam Help

- Software Refactoring:
    - Antipatterns Detection <https://powcoder.com>
    - Antipatterns Correction (Refactoring Recommendation)
- Add WeChat powcoder



- Refactoring **Assignment Project Exam Help**
  - The process of improving a code after it has been written **by** **changing its internal structure without changing the external behavior** (<https://powcoder.com> **Add WeChat powcoder** Fowler et al., '99)
  - Examples: *Move method, extract class, move attribute, ...*
- Two main refactoring steps
  1. detection of code fragments to improve (e.g., Anti pattern)
  2. identification of refactoring solutions



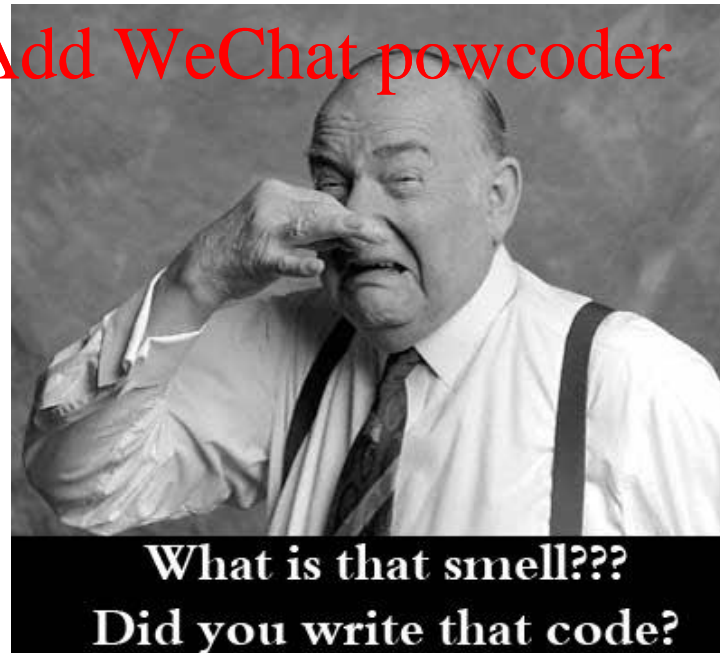
# Learning objectives

- Anti-patterns: Overview
- Examples of anti-patterns
- How to detect anti-patterns using quality metrics?

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



# Design defects/antipatterns

- Design defects/antipatterns are poor coding and design choices introduced during different phases of software development
  - Anomalies, code smells, bad smells...
  - Make the design harder to understand, to change
  - Design situations that adversely affect the development of a software (**not bugs**)

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

# Management Antipatterns



# Examples of antipatterns

- Duplicated code
  - Long method
  - God class
  - Long parameter list
  - Message chain
  - Data class
  - Functional decomposition
  - ...
- Assignment Project Exam Help
- <https://powcoder.com>
- Add WeChat powcoder

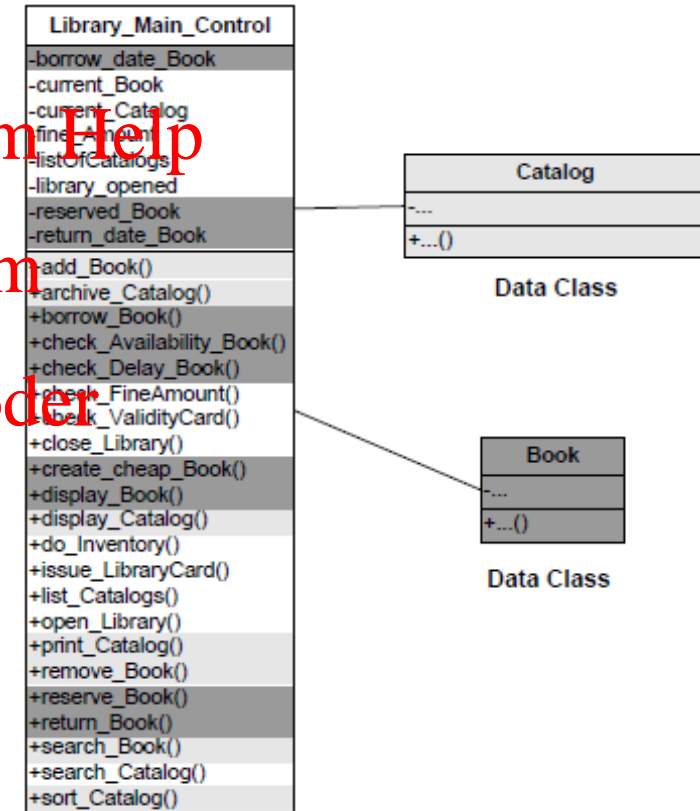
# The Blob (God Class)

- Definition

- Procedural-style design leads to one object with numerous responsibilities and most other objects only holding data or executing simple processes.

- Symptoms

- A Blob is a *controller* class, *abnormally large*, with *almost* no parents and no children. It *mainly uses* data classes, i.e. *very small* classes with *almost* no parents and no children ([Brown et al. '98](#)).





- Causes

- *Lack of an object-oriented architecture.*
  - *Inadequate understanding of OO principles*
- *Lack of (any) architecture*
  - *Design, interaction between object etc.*
- *Too limited intervention*

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

# Duplicate Code or Cut and Paste Programming

- Code reused by copying source statements leads to significant maintenance problems.
  - Duplicate methods in subclasses
  - Duplicate expressions in same class
  - Duplicate expressions in different classes

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

```
def to_time(value)
  if value.respond_to?(:to_time)
    value.to_time
  else
    Coercion::String.to_time(to_string(value))
  end
end

def to_datetime(value)
  if value.respond_to?(:to_datetime)
    value.to_datetime
  else
    Coercion::String.to_datetime(to_string(value))
  end
end
```

```
public function add()
{
    if($this->input->post("departmentName"))
    {
        $departmentName = $this->input->post("departmentName");
        $departmentBudget = $this->input->post("departmentBudget");

        $this->departments_m->addDepartment($departmentName, $departmentBudget);
        redirect("/departments");
    }

    $data = array('title' => 'Add Department - DB Hotel Management System', 'page' => 'departments');
    $this->load->view('header', $data);
    $departments = $this->departments_m->get_departments();
    $viewdata = array('departments' => $departments);
    $this->load->view('departments/add', $viewdata);
    $this->load->view('footer');
}
```

Assignment Project Exam Help

<https://powcoder.com>

```
public function edit($department_id)
{
    if($this->input->post("departmentName"))
    {
        $departmentName = $this->input->post("departmentName");
        $departmentBudget = $this->input->post("departmentBudget");

        $this->departments_m->editEmployee($department_id, $departmentName, $departmentBudget);
        redirect("/departments");
    }

    $data = array('title' => 'Edit Department - DB Hotel Management System', 'page' => 'departments');
    $this->load->view('header', $data);
    $department = $this->departments_m->getDepartment($department_id);
    $viewdata = array('department' => $department[0]);
    $this->load->view('departments/edit', $viewdata);
    $this->load->view('footer');
}
```

Add WeChat powcoder

# Large method

- A method that does more than one thing
  - Many things, sometimes unconnected things
- Problems
  - Could indicate low levels of abstraction, low level of class design, reduced re-usability
  - Harder to test, poor readability

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

# Long Parameter List

- Introduce parameter object
- Only worthwhile if there are several methods with same parameter list, and they call each other

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

```
user= userManager.create(USER_NAME,group, USER_NAME,  
“test”, Language, false, false, new Date(), “blah”, “new Date() )
```



# Message Chain

- Long list of method calls:
  - `customer.getAddress().getState()`
  - `window.getBoundingBox().getOrigin().getX()`

<https://powcoder.com>

Add WeChat powcoder

# Message Chain

```
Class Employee{
    public function getConfiguration() {
        $this->employeeConfig->getConfiguration();
    }
}

Class EmployeeConfig{
    public function getConfiguration() {
        $this->config->getConfiguration();
    }
}

Class Config{
    public function getConfiguration() {
        $this->loadConfiguration();
    }
}
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

# Data Class/Lazy Class

- Class has no methods except for getter and setters

Assignment Project Exam Help

<https://powcoder.com>



```
public class Example implements InterfaceExample {  
    @Override  
    public void doSomething() {  
        // Input text is put here.  
    }  
}
```

Add WeChat powcoder

# Switch statements

- **Switch** statements are very rare in properly designed object-oriented code
  - Therefore, a **switch** statement is a simple and easily detected “bad smell”
  - Of course, not all uses of **switch** are bad
  - A switch statement should *not* be used to distinguish between various kinds of object

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

# Example 1, continued

- class Animal {  
    final int MAMMAL = 0, BIRD = 1, REPTILE = 2;  
    int myKind; Assignment Project Exam Help  
    ...  
    String getSkin() { <https://powcoder.com>  
        switch (myKind) {  
            case MAMMAL: return "hair"; Add WeChat powcoder  
            case BIRD: return "feathers";  
            case REPTILE: return "scales";  
            default: return "skin";  
        }  
    }  
}



# Example 1, improved

```
class Animal {  
    String getSkin() { return "skin"; }  
}  
class Mammal extends Animal {  
    String getSkin() { return "hair"; }  
}  
class Bird extends Animal {  
    String getSkin() { return "feathers"; }  
}  
class Reptile extends Animal {  
    String getSkin() { return "scales"; }  
}
```

Assignment Project Exam Help  
<https://powcoder.com>  
Add WeChat powcoder

# Dead Code/ Old Baggage

- Description

- System contains many classes whose purpose is not known
  - Lava Flow, ~~Dead Code~~
- Much of the code is left over from previous ideas and no longer has a purpose
  - was once fluid and useful, now is solid lava that you are afraid to remove

- Consequences

- difficult to maintain, just gets worse

# Comments?

- The purpose of comments should be only “why you are doing something (to help future modifiers, rather than “what code is doing”
- Whenever possible make your code express the intent of the comment and remove the comment.
- Comments are to provide intent that is not expressible in code
- Any comment that duplicates what the code says should be deleted

```
public void add(Object element) {  
    if (!readOnly) {  
        int newSize = size + 1;  
        if (newSize > elements.length) {  
            // grow the array  
            Object[] newElements =  
                new Object[elements.length + 10];  
            for (int i = 0; i < size; i++)  
                newElements[i] = elements[i];  
            elements = newElements;  
        }  
        elements[size++] = element;  
    }  
}
```

```
public void add(Object element) {  
    if (!readOnly) {  
        int newSize = size + 1;  
        if (newSize > elements.length) {  
            grow();  
        }  
        elements[size++] = element;  
    }  
}
```

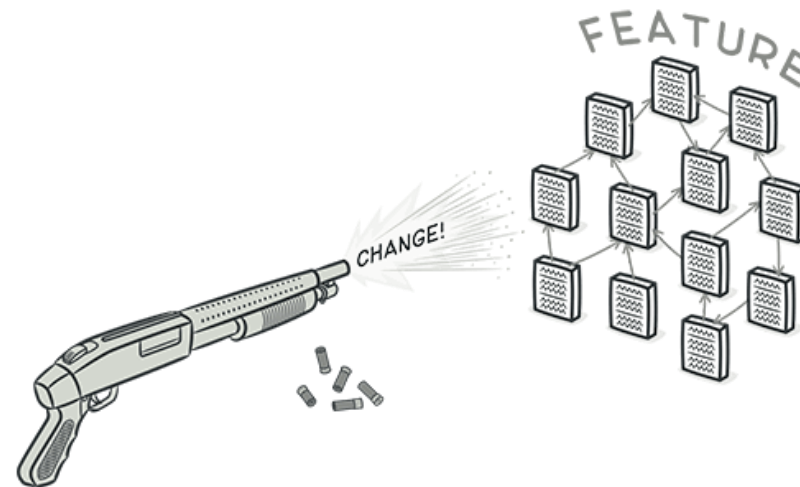
Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

# Shotgun Surgery

- This happens when, you want to make some kind of change, you are forced to make a lot of changes to a lot of different classes
- And when changes are all over the place, they are hard to find, and it's easy to miss an important change



# Functional Decomposition

- Description

- Classes with names "functions"
- All class attributes are private and are used only inside the class
- Classes with a single action similar to a procedural function.

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

- Consequences

- No O/O benefits such as inheritance and polymorphism
  - Expensive to maintain
  - Complexity of testing software
  - Complexity of reuse of the code

# Spaghetti code

- Description

- System hard to debug, modify
- Bunch of code similar in structure to a bowl of spaghetti.
  - Bad coding practices

- Consequences

- Low readability
- Impossible to understand how it exactly works

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

# Feature envy

- When a method seems more interesting in a class, other than the one in actually it is
- Example : a method that invokes half a dozen getting methods on another object to calculate some value.

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



```
class User
{
    private $contactInfo;

    public function __construct()
    {
        $this->contactInfo = new ContactInfo();
    }

    public function getFullAddress()
    {
        $address = $this->contactInfo->getStreetName();
        $address .= " " . $this->contactInfo->getStreetNumber() . ', ';
        $address .= $this->contactInfo->getZipCode() . ', ';
        $address .= $this->contactInfo->getCity() . ', ';
        $address .= $this->contactInfo->getCountry();

        return $address;
    }
}
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

# Class Activity

Assignment Project Exam Help

See additional file added on d2l

<https://powcoder.com>

Add WeChat powcoder