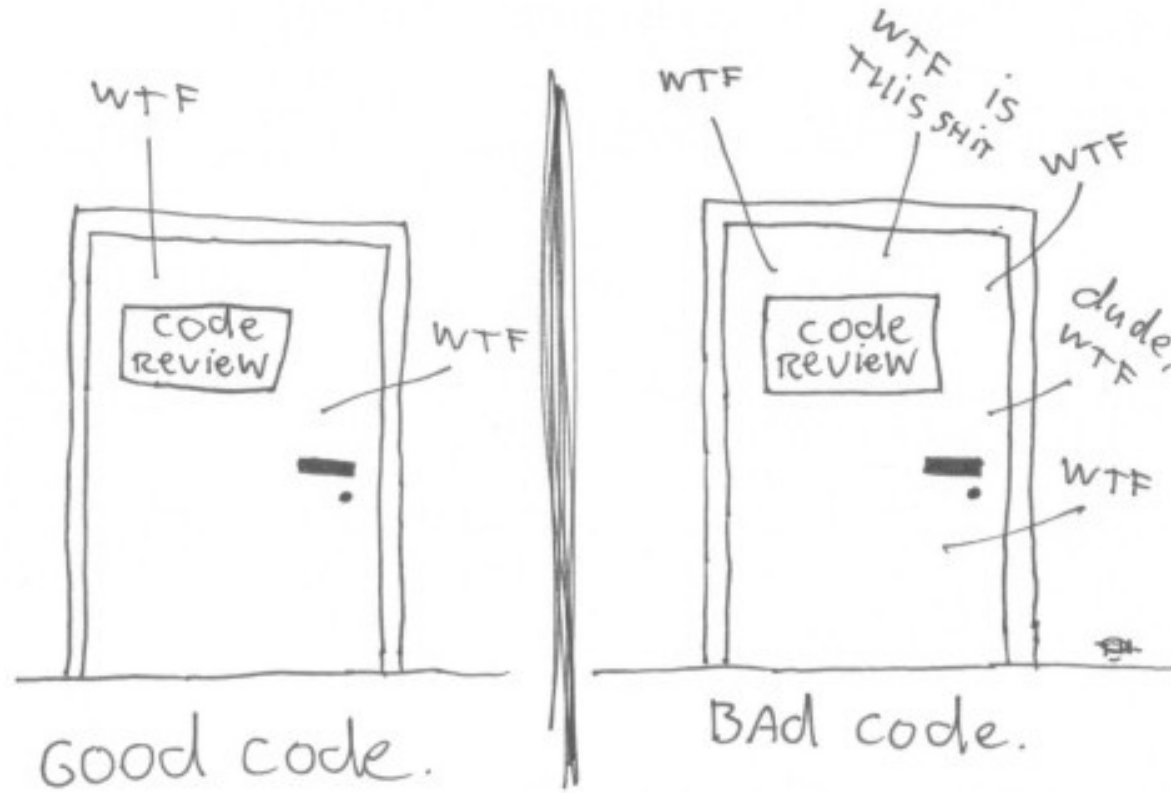


**CLEAN CODE**

The ONLY VALID MEASUREMENT  
OF CODE QUALITY: WTFs/MINUTE



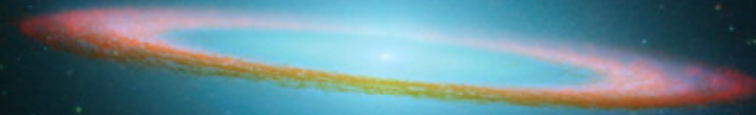
**OLVASHATÓ, KARBANTARTHATÓ  
ÉS BŐVÍTHETŐ KÓD**

PRENTICE  
HALL

Robert C. Martin Series

# Clean Code

A Handbook of Agile Software Craftsmanship



Foreword by James O. Coplien

Robert C. Martin

# MILYEN A ROSSZ KÓD?

- Nehezen átlátható - olvashatatlan
- A szerepkörök tisztázatlanok
- Problémás módosítani és bővíteni
- Minden részében óriási



# MILYEN A TISZTA KÓD?

- Kicsi, elegáns és hatékony
- Egyetlen dolgot csinál, de azt jól!
- Úgy olvasható, mint egy könyv

*“Clean code reads like well-written prose”*

*“Clean code always looks like it was written by someone who cares. There is nothing obvious that you can do to make it better.”*

*“[...]when each routine you read turns out to be pretty much what you expected[...].”*



**BESZÉDES NEVEK**

# BESZÉDES NEVEK

```
getThem: function() {  
    var tempList = [];  
    for (var x in this.list)  
        if (x[0] == 4)  
            tempList.push(x);  
    return tempList;  
}
```

# BESZÉDES NEVEK

```
getFlaggedCells: function() {  
    var flaggedCells = [];  
    for (var cell in this.gameBoard) {  
        if (cell[STATUS_VALUE] == FLAGGED) {  
            flaggedCells.push(cell);  
        }  
    }  
    return flaggedCells;  
}
```

# BESZÉDES NEVEK

```
getFlaggedCells: function() {  
    var flaggedCells = [];  
    for (var cell in this.gameBoard) {  
        if (cell.isFlagged()) {  
            flaggedCells.push(cell);  
        }  
    }  
    return flaggedCells;  
}
```

# BESZÉDES NEVEK

```
getFlaggedCells: function() {  
    return this.cellsOnBoard.filter(function(cell) {  
        return cell.isFlagged();  
    });  
}
```

# BESZÉDES NEVEK

```
var d; // number
```

```
var d; // elapsed time in days
```

```
var elapsedTimeInDays;
```

# FÉLREÉRTÉSEK ELKERÜLÉSE

```
var a = 1;                                MediaFileStorageValidator  
if (0 == 1) a = 01;                       MediaFilesStorageValidator  
else 1 = 01;
```

# KIMONDHATÓ NEVEK

```
var genymdhms;  
var modymdhms;
```

```
var generationTimestamp;  
var modificationTimestamp;
```



# KERESHETŐ NEVEK

```
for (var j=0, w=4; j<34; j++) {  
    w -= t[j];  
    if (w >= 0) s++;  
}
```

```
var pointsForTheWeek = 4;  
var sum = 0;  
for (var j = 0; j < NUMBER_OF_TASKS; j++) {  
    pointsForTheWeek -= taskEstimations[j];  
    if (pointsForTheWeek >= 0) sum++;  
}
```

# HUNGARIAN NOTATION

phoneNumber, cellArray, sAddress

phoneString, cellObjects, oAddress

phone, cells, address

# OSZTÁLYNEVEK

Manager, Processor, Data, Info

Customer, WikiPage, Account, AddressParser

SocketManager > SocketClient/SocketConnector

CustomerData > Customer

AddressInfo > Address/AddressParser

# METÓDUSNEVEK

// igék vagy igei kifejezések

postPayment, getName, save

```
var name = employee.getName();
```

```
customer.setName("Mike");
```

```
if (paycheck.isPosted()) { ... }
```

# EGYÉRTELMŰ KONSTRUKTOR FACTORY METÓDUSSAL

```
var complex = new Complex(19.2);
```

```
var complex = Complex.fromRealNumber(19.2);
```

```
Complex.fromRealNumber = function(realNumber) {  
    return new Complex(realNumber);  
}
```

# FACTORY METHOD

```
var api = new SuiteAPI('test-suite.com',  
    'secretKey', true);
```

```
var api = new SuiteAPI('proxy-suite.com',  
    'secretProxyKey', true);
```

```
SuiteAPI.getFor('test-suite.com');
```

```
SuiteAPI.get();
```

# VÁLASSZ EGY KIFEJEZÉST UGYANARRA A CÉLRA

`fetch - retrieve - get`

`fetchCells - retrieveCells - getCells`

`controller - manager - handler`

`DeviceController - DeviceManager - DeviceHandler`

# PROGRAMOZÓKNAK SZÓL

SectionDrawStrategy, CustomerDeleteCommand,  
JobQueue, AccountVisitor

~~iSectionDrawStrategyAbstractFactory~~



# MILYEN A JÓ NÉV?

- Egyszerű, de egyenes, felfedi a szándékainkat, pontosan leírja, hogy mit takar
- Mindig egyértelmű
- Könnyen kereshető
- Nem tartalmaz típusokat
- **Angol!**

**FÜGGVÉNYEK**

**KICSIK!**

**1. szabály:** legyen kicsi

**2. szabály:** legyen még kisebb

Kevesebb, mint **150 karakter egy sorban**

Kevesebb, mint **20 sor**

**EGYETLEN DOLGOT  
CSINÁLJANAK!**

# ABSZTRAKCIÓS SZINTEK TARTÁSA ÉS A LESZÁLLÓ SZABÁLY

```
function responseToRequest(request) { // Legmagasabb
    request.body = getHtml();          // absztrakció
}

...

function getHtml() {                  // Középső absztrakció
    View.render({ data: 'TEST' });
}

...

View.render = function() {           // Alacsony
    htmlBody.append('\n');           // absztrakció
}
```



**FÜGGVÉNY ARGUMENTUMOK SZÁMA**

**MAXIMUM 1**

**NEHEZEN 2**

**NAGYON INDOKOLT ESETBEN 3**



```
var dialog = new Dialog(300, 500, 'Test window',  
    true, this.container);  
  
var dialog = new Dialog(null, null, 'Test window',  
    null, this.container);  
  
var dialog = new Dialog({  
    title: 'Test window',  
    parentElement: this.container  
});
```

# BOOLEAN PARAMÉTEREK

```
html.render(true);
```

```
html.print();
```

```
html.getContent();
```

# Ø MEGLEPETÉS, Ø MELLÉKHATÁS!

```
// tegyél valamit vagy válaszolj valamire, de
```

```
// sose egyszerre!
```

```
var name = user.setFullName('Test', 'Elek');
```

```
// Helyett:
```

```
user.setFullName('Test', 'Elek');
```

```
user.getName();
```

```
setAndCheckIfExists()
```

```
// Helyett:
```

```
if (attributeExists("username")) {
```

```
    setAttribute("username", "unclebob");
```

```
}
```

# DRY

DO NOT REPEAT YOURSELF

# MILYEN A JÓ FÜGGVÉNY?

- Kicsi
- Annál is kisebb
- Egy dolgot csinál
- Egy absztrakciós szintet tartalmaz
- Kevés paramétere van
- Nincs mellékhatása

**KOMMENTEK**

**A LEGJOBB KOMMENT A NEM  
LÉTEZŐ KOMMENT**

# FEJEZD KI MAGAD A KÓDBAN

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

if (employee.isEligibleForFullBenefits()) { ... }
```



# OBJEKTUMOK ÉS ADATSTRUKTÚRÁK

# PRIMITIVE OBSESSION

```
var Car = function() {  
    this.red = 0; this.blue = 0; this.green = 0;  
};
```

```
Car.prototype.paint = function(red, green, blue) {  
    this.red = red;  
    this.blue = green;  
    this.green = blue;  
}
```

```
var car = new Car();  
car.paint(122, 123, 210);
```

Mi van, ha CMYK-ban szeretném megadni a színt?

# PRIMITIVE OBSESSION

```
var Car = function() {  
    this.color = null;  
};
```

```
Car.prototype.paint = function(color) {  
    this.color = color;  
}
```

```
var car = new Car();  
car.paint(Color.fromRGB(122, 123, 210));  
car.paint(Color.fromCMYK(1, 2, 3, 4));
```

# THE LAW OF DEMETER

```
function getOutputDirectory(context) {  
    return context  
        .getOptions()  
        .getDirectory()  
        .getAbsolutePath();  
}
```

# EGY OSZTÁLY METÓDUSA CSAK A KÖVETKEZŐ METÓDUSOKAT HÍVHATJA MEG:

```
User.prototype.someMethod = function(otherObject) {  
    // Saját osztályának metódusait  
    this.anotherMethod();  
  
    // Az osztály példányváltozóiba mentett objektumok metódusait  
    this.customer.customerMethod();  
  
    // Egy argumentumban átadott osztály metódusait  
    otherObject.methodFromArgument();  
  
    // A metódus által létrehozott objektum metódusait  
    (new Permission).permissionMethod();  
}
```

# OSZTÁLYOK

# FORMÁTUM

- Konstansok
- Publikus és privát példányváltozók
- Publikus függvények
- Privát függvények abban a sorrendben, ahogy a publikusban meg vannak hívva

1. szabály: legyen kicsi

2. szabály: legyen még kisebb

Kevesebb, mint 10 metódus, privátokkal együtt



# SRP

**SINGLE RESPONSIBILITY  
PRINCIPLE**

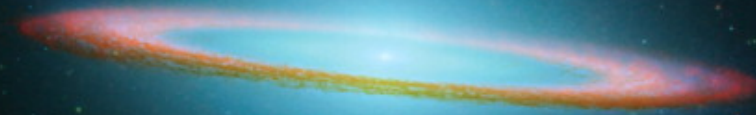
# OSZTÁLY KOHÉZIÓ

PRENTICE  
HALL

Robert C. Martin Series

# Clean Code

A Handbook of Agile Software Craftsmanship



Foreword by James O. Coplien

Robert C. Martin

