



Why getter and setter methods are evil

: Allen Holub
: (lemigeo@hotmail.com)
:
: Bizcom Java Article Center

[Intermediate level java developer]

가 ,

Bizcom Java Article Center

1 - (ahnyounghoe@empal.com)

Allen Holub "Why getter and setter methods are evil" JavaWorld

<http://www.javaworld.com/javaworld/jw-09-2003/jw-0905-toolbox.html>

"Why getter and setter methods are evil" by Allen Holub was originally published by JavaWorld (www.javaworld.com), copyright JavaWorld.com, translated and reprinted with permission.

<http://www.javaworld.com/javaworld/jw-09-2003/jw-0905-toolbox.html>

(Summery)

getter/setter (idiom) (accessor) ,
 (encapsulation) , getter/setter / (design methodology) . (2003-09-05)

“ (is evil) ” ,
 “[Why extends Is Evil](#)” get/set
 getter/setter 가 , (OO)
 (Object Oriented : OO) ,
 , OO , getter setter
 가
 (red flag) .
 getter/setter
 , getter/setter

(On the nature of design)

가 .
 “[Why extends is evil](#)” (Talkback)
 extends가 가 가
 가 가 .
 (trade-
 off) 가 가 ,
 (context) ,

. Gang of Four Design Patterns

“Consequences”

(chapter)

(idiom) 가

가

(incentive)

(design idiom)

가

. so they de-emphasize that the way in which you do it is less than

ideal.

”

(practice)

(EJB : Enterprise JavaBean)

가

. EJB

가

가

(Data abstraction)

getter/setter

getX() 1,000
가 , getX()

X 가 가

X가 int , long , 1,000 가
int ,
가
1,000 가

(data abstraction)

가
(constant) 가 private

public , ,
1,000

(implementation hiding principle) ,

“ 가 가?” 가

(accessor) getter/setter , public 가 가
가 가

가?

가

(encapsulation principle)

가 (design process)

getter/setter 가 , 가

가 getter/setter 가

(relationship)

(association)

(relationship)

(association)

가 , 가

가 , (design-by-guessing strategy)

가 (force of habit)

가 , C () struct 가 , struct 가 , public 가 , private

private , public

public

Draw thyself

(UI) , UI (UI builder class) getAttribute() drawYourself(...)

가 Identity , drawYourself() identity JComponent (give-me-a-JComponent-that-represents-your-identity) getIdentity가 “get” 가

가 Identity 가 , identity가

drawYourself() UI . UI “identity”가 ID

가 ID 가 identity

JComponent (give-me-a-JComponent-that-represents-your-identity)

가 , identity

UI

AWT (Abstract Window Toolkit)

Swing

UI

UI

AWT/Swing

(JavaBeans)

“ (JavaBeans) 가?” ,

?

getter/setter

JavaBeans

. BeanCustomizer, BeanInfo

BeanDescriptor

getter/setter

. getter/setter

, UI

Class

API (introspection API)

(metadata)

2004

Java 1.5

private int property;

public int getProperty () { return property; }

```
public void setProperty (int value){ property = value; }
```

```
private @property int property;
```

UI

API

가? (When is an accessor okay?)

가

“getter”가

(procedural boundary layer)

가

Java Database Connectivity (JDBC)

API

가

가

가

(not-knowing-how-it-will-be-used)

JDBC

(A design strategy)

, getter/setter 가?
 use case (task) 가
 use case가 use case .)
 use case use case
 (activities) use case use case
 use case . ,

1989 Kent Beck Ward Cunningham 가 ,
 get/set 가

가 가
 가 ,

Cunningham CRC(classes, responsibilities, collaboration)
 4 x 6


- (Class) :
- (Responsibilities) : 가
- (Collaborators) :
 가

CRC ,
 , Beck Cunningham use case , use case
 가 가 .

CRC
가 ,
, use case
(use case) , use case
가 가 , CRC
가
가
가
가 ,
CRC , CRC 가
CRC 가 가 가 , (CRC)
. 4 x 6
CRC
가
CRC
Unified Modeling Language (UML) /
, get/set
, get/set 가
(Summing up)
가 , (getter/setter)
가
, get/set
가 ,

99% 가

(coder) 가 , getter/setter 가
Cunningham CRC

가 Holub on Patterns :
Learning Design Patterns by Looking at Code , 가 Apress (www.apress.com)


(Resources)

Java Toolbox (1998 7) 가
UI

<http://www.javaworld.com/javaworld/jw-07-1999/jw-07-toolbox.html>

CRC Kent Back Ward Cunningham

<http://c2.com/doc/oopsla89/paper.html>

David Bellin Susan Suchman Simon (Addison-Wesley 1997, ISBN 0201895358)
The CRC Card Book CRC

<http://www.amazon.com/exec/obidos/ASIN/0201895358/alleiholuasso>

Java Toolbox Allen Holub

<http://www.javaworld.com/columns/jw-toolbox-index.shtml>

David Geary Java Design Patterns

<http://www.javaworld.com/columns/jw-java-design-patterns-index.shtml>

Design Patterns, Eric Gamma, Richard Helm, Ralph Johnson, and John Vlissides
(Addison-Wesley Publishing Co., 1995; ISBN: 0201633612)

<http://www.amazon.com/exec/obidos/ASIN/0201633612/javaworld>

JavaWorld Topical Index **Design Patterns**

http://www.javaworld.com/channel_content/jw-patterns-index.shtml

JavaWorld Topical Index **Object-Oriented Design and Programming**

http://www.javaworld.com/channel_content/jw-oop-index.shtml

Programming Theory & Practice

<http://www.javaworld.com/javaforums/postlist.php?Cat=&Board=TheoryPractice>

JavaWorld 가

<http://www.javaworld.com/subscribe>