

Java Programming Lab Manual

Asadullah Shah

Zeeshan Bhatti



IIUM PRESS

INTERNATIONAL ISLAMIC UNIVERSITY MALAYSIA

JAVA PROGRAMMING LAB MANUAL

Editors

**Asadullah Shah
Zeeshan Bhatti**

**Department of
Kulliyah of Information & Communication Technology
International Islamic University Malaysia**

January 2011



IIUM Press

Published by:
IIUM Press
International Islamic University Malaysia

First Edition, 2011.
©IIUM Press, IIUM

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without any prior written permission of the publisher.

Perpustakaan Negara Malaysia

Cataloguing-in-Publication Data

Bibliography p.
Includes Index
ISBN

ISBN: 978-967-418-086-7

Member of Majlis Penerbitan Ilmiah Malaysia – MAPIM
(Malaysian Scholarly Publishing Council)

Printed by :
IIUM PRINTING SDN. BHD.
No. 1, Jalan Industri Batu Caves 1/3
Taman Perindustrian Batu Caves
Batu Caves Centre Point
68100 Batu Caves
Selangor Darul Ehsan

TABLE OF CONTENTS

Dedication	v
Preface	xi
Acknowledgement	xiii
Chapter 1: Introduction to Java	15
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 2: Compiling and Executing Java Code	21
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 3: Introduction to Data Types in Java	27
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 4: Arithmetic and Relational Operators	33
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 5: Relational and Logical Operators	39
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 6: Bitwise Operators and Type Casting	45
- Asadullah Shaikh	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 7: Conditional Statements	51
- Asadullah Shaikh	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 8: If-Else Statements	55
- Asadullah Shaikh	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 9: The If-Else-If Statements	61
- Asadullah Shaikh	
- Zeeshan Bhatti	
- Asadullah Shah	

Chapter 10: Switch Statement	67
- Asadullah Shaikh	
- Zeeshan Bhatti	
- Asadullah Shah	
	73
Chapter 11: Loops	
- Agha Javed Phatan	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 12: Do-While Loops	79
- Agha Javed Phatan	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 13: For Loops	85
- Agha Javed Phatan	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 14: Loop Pitfalls and Exercise	93
- Agha Javed Phatan	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 15: Nested Loops	99
- Agha Javed Phatan	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 16: Arrays in Java	105
- Waseem Javaid Soomro	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 17: Two Dimensional Arrays	111
- Waseem Javaid Soomro	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 18: Strings	119
- Waseem Javaid Soomro	
- Zeeshan Bhatti	
- Asadullah Shah	

Chapter 19: Methods of String Class	127
- Waseem Javaid Soomro	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 20: Object and Classes	135
- Waseem Javaid Soomro	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 21: Constructors	141
- Dil Nawaz Hakro	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 22: User defined Methods	147
- Dil Nawaz Hakro	
- Zeeshan Bhatti	
- Asadullah Shah	viii
Chapter 23: Arguments Passing	155
- Dil Nawaz Hakro	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 24: Return Statement	163
- Dil Nawaz Hakro	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 25: Working with Classes	169
- Dil Nawaz Hakro	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 26: Multiple Classes	177
- Muniba Shaikh	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 27: Method Overloading and Overriding	185
- Muniba Shaikh	
- Zeeshan Bhatti	
- Asadullah Shah	

Chapter 28: Modifiers for class data	191
- Muniba Shaikh	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 29: Inner Classes	199
- Muniba Shaikh	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 30: Comparison between different class types	207
- Muniba Shaikh	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 31: Exception Handling	215
- Kamran Khowaja	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 32: Throw & Throws Keyword	223
- Kamran Khowaja	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 33: Local Block and Static Block	231
- Kamran Khowaja	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 34: Abstract Classes	237
- Dini Oktarina Dwi	
- Zeeshan Bhatti	
- Asadullah Shah	
Chapter 35: Interfaces	245
- Dini Oktarina Dwi	
- Zeeshan Bhatti	
- Asadullah Shah	

Chapter 28

Modifiers for Class Data

Muniba Shaikh, Zeeshan Bhatti, Asadullah Shah

Abstract

In Java various modifiers are used to extend the functionality of the code and give more power to the programmer. Some of these modifiers are discussed in this chapter where students will learn how to use them and what are their benefits in programming.

28.1 Static keyword

A *static field* (a.k.a. *class field* or *class variable*) is shared by all objects of the class. A non-static field (a.k.a. *instance field* or *instance variable*) belongs to an individual object.

- *static* is basically a memory category. When a program is loaded, a chunk of memory is allocated to hold the program's code and permanent data. This memory is called "static," as opposed to the "dynamic" memory allocated for each newly created object. Dynamic memory is allocated and freed as objects are created and disposed of. If you have a constant that is the same for all objects of the class, there is no need to duplicate it in each object; instead you can declare this constant as static to store it once in static memory, together with the class's code. Hence the term "static." Instance variables are held in "dynamic" memory.
- A static field can hold a constant shared by all objects of the class:

```
public class RollingDie
{
//Reserved words:static final
```