

ORACLE
PRESS

Also Covers **Java SE 11** Developer Exam



OCP

ORACLE CERTIFIED PROFESSIONAL

JAVA SE 21 Developer

Exam
1Z0-830

JAVA SE 17 Developer

Exam
1Z0-829

Programmer's Guide

ORACLE

Khalid A. Mughal
Vasily A. Strelnikov

Important Information About The Programmer's Guide

This Programmer's Guide is comprised of two parts:

- *Part I: OCP Java SE 21 Developer (Exam 1Z0-830)*
- *Part II: OCP Java SE 17 Developer (Exam 1Z0-829)*

Part I can be used in conjunction with Part II to prepare for the *OCP Java SE 21 Developer exam*.

Part II can be used to prepare for the *OCP Java SE 17 and Java 11 Developer exams*.

OCP Oracle Certified Professional Java SE 21 Developer (Exam 1Z0-830) Java SE 17 Developer (Exam 1Z0-829) Programmer's Guide

Table of Contents

Cover

Title Page

Copyright Page

Volume 1

Dedication

Part I: Contents

Part I: Figures

Part I: Tables

Part I: Examples

Preface to OCP Java SE 21 Developer (Exam 1Z0-830)

About Part I: OCP Java SE 21 Developer

Exam Objectives

Java SE Platform API Documentation

Website for Part I: OCP Java SE 21 Developer

Request for Feedback

About the Authors

Acknowledgments

1 Pattern Matching for Java Objects

1.1 Type Pattern Matching with the instanceof Pattern Match Operator

Table of Contents

The instance of Type Comparison Operator

The instance of Pattern Match Operator

Valid Operand Types in the instance of Pattern Match Operator

- Same Types as Operands

- Unrelated Types as Operands

- Unrelated Classes as Operands

- Peer Classes as Operands

- Final Classes as Operands

- Sealed Types as Operands

- Generic Types as Operands

Properties of a Pattern Variable

Flow Sensitive Scope of Pattern Variables

1.2 Type Pattern Matching with the Enhanced switch Construct

- Using Type Patterns in case Labels

- The null Value as Case Constant in switch Construct

- Guarded Case Patterns

- Case Label Dominance

- Flow Sensitive Scope of a Pattern Variable in switch Construct

- Exhaustiveness of Case Pattern Labels

 - Sealed Types and Exhaustiveness

- Illegal Fall-Through to a Pattern in switch Statement

1.3 Record Patterns

1.4 Record Pattern Matching with the instance of Pattern Match Operator

- Type Inference of Pattern Variables using var

- Flow Sensitive Scope of Pattern Variables in Record Patterns

- Nested Record Patterns

- Using Record Patterns with Generic Types

1.5 Record Pattern Matching with the Enhanced switch Construct

- Nested Record Patterns in case Labels

- Guarded Record Patterns in case Labels

- Flow Sensitive Scope of Pattern Variables in the switch Construct

- Type Inference of Record Pattern Variables in case Labels

- Case Label Dominance in Record Patterns

- Illegal Fall-Through in case Labels with Record Patterns

Table of Contents

Using Sealed Types with Record Patterns in case Labels

Using Generic Record Patterns in case Labels

Using Generic Record Patterns parameterized with Sealed Types

1.6 Summary of Java Patterns

1.7 Using Qualified Name for Enum Constants in the switch Construct

1.8 Unexpected Failure in Pattern Matching

Review Questions

2 Sequenced Collections and Maps

2.1 Collections with Well-Defined Encounter Order

2.2 Sequenced Collections

The `SequencedCollection<E>` Interface

2.3 Sequenced Lists

The `List<E>` Interface

The `ArrayList<E>` Class

2.4 Sequenced Sets

The `SequencedSet<E>` Interface

The `SortedSet<E>` Interface

The `NavigableSet<E>` Interface

The `TreeSet<E>` Class

The `LinkedHashSet<E>` Class

2.5 Deques

The `ArrayDeque<E>` Class

The `LinkedList<E>` Class

2.6 Sequenced Maps

The `SequencedMap<K, V>` Interface

Views on Keys, Values, and Entries of a `SequencedMap<K, V>`

Composing Views

Static Entry Snapshots

The `SortedMap<K, V>` Interface

The `NavigableMap<K, V>` Interface

The `TreeMap<K, V>` Class

The `LinkedHashMap<K, V>` Class

Table of Contents

2.7 Unmodifiable Sequenced Collections, Sets, and Maps

2.8 Sequenced Concurrent Collections

2.9 Sequenced Concurrent Maps

Review Questions

3 Virtual Threads

3.1 Motivation for Virtual Threads

3.2 Virtual Thread Execution Model

3.3 Using Thread Class to Create Virtual Threads

Logging Information during Program Execution

Creating and Starting a Virtual Thread

3.4 Using Thread Builders to Create Virtual Threads

Important Aspects of Virtual Threads

Is a Thread Virtual?

Virtual Threads are Daemon Threads

Virtual Threads have Normal Priority

Virtual Threads belong to VirtualThreads Group

3.5 Using Thread Factory to Create Threads

3.6 Using Thread Executor Services

Using the Virtual-Thread-Per-Task Executor Service

Customizing the Thread-Per-Task Executor Service

The Executors Utility Class

3.7 Scalability of Throughput with Virtual Threads

3.8 Best Practices for Using Virtual Threads

Avoid Pinning of Virtual Threads

Pinning in Synchronized Block

Avoiding Pinning with a Reentrant Lock

Avoid Using Virtual Threads for CPU-Bound Tasks

Avoid Pooling of Virtual Threads

Minimize Using Thread-Local Variables with Virtual Threads

Avoid Substituting Virtual Threads for Platform Threads

Review Questions

A: Exam Topics: Java SE 21 Developer Professional

Table of Contents

B: Annotated Answers to Review Questions

C: Miscellaneous API Updates

C.1 The String and StringBuilder Classes

Searching in a String

Extending a String Builder

C.2 Constructing Locales

Part I: Index

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

Table of Contents

Y

Volume 2

Title Page

Dedication

Part II: Contents

Part II: Contents Overview

Part II: Contents

Part II: Figures

Part II: Tables

Part II: Examples

Foreword

Preface to OCP Java SE 17 Developer (Exam 1Z0-829)

About Part II: OCP Java SE 17 Developer

Using Part II: OCP Java SE 17 Developer

Website for Part II: OCP Java SE 17 Developer

Acknowledgments

1 Basics of Java Programming

1.1 The Java Ecosystem

1.2 Classes

1.3 Objects

1.4 Instance Members

1.5 Static Members

1.6 Inheritance

1.7 Aggregation

Review Questions

1.8 Sample Java Program

1.9 Program Output

Review Questions

Table of Contents

2 Basic Elements, Primitive Data Types, and Operators

- 2.1 Basic Language Elements
- 2.2 Primitive Data Types
- 2.3 Conversions
- 2.4 Type Conversion Contexts
- 2.5 Precedence and Associativity Rules for Operators
- 2.6 Evaluation Order of Operands
- 2.7 The Simple Assignment Operator =
- 2.8 Arithmetic Operators: *, /, %, +, -
- 2.9 The Binary String Concatenation Operator +
- 2.10 Variable Increment and Decrement Operators: ++, --
- Review Questions
- 2.11 Boolean Expressions
- 2.12 Relational Operators: <, <=, >, >=
- 2.13 Equality
- 2.14 Boolean Logical Operators: !, ^, &, |
- 2.15 Conditional Operators: &&, ||
- 2.16 Integer Bitwise Operators: ~, &, |, ^
- 2.17 Shift Operators: <<, >>, >>>
- 2.18 The Conditional Operator ?:
- 2.19 Other Operators: new, [], instanceof, ->

Review Questions

3 Declarations

- 3.1 Class Declarations
- 3.2 Method Declarations
- 3.3 Statements
- 3.4 Variable Declarations
- 3.5 Instance Methods and the Object Reference this
- 3.6 Method Overloading
- 3.7 Constructors

Table of Contents

3.8 Static Member Declarations

Review Questions

3.9 Arrays

3.10 Parameter Passing

3.11 Variable Arity Methods

3.12 The main() Method

3.13 Local Variable Type Inference

Review Questions

4 Control Flow

4.1 Selection Statements

4.2 The switch Statement

4.3 The switch Expression

Review Questions

4.4 Iteration Statements

4.5 The while Statement

4.6 The do-while Statement

4.7 The for(;;) Statement

4.8 The for(:) Statement

4.9 Transfer Statements

4.10 Labeled Statements

4.11 The break Statement

4.12 The continue Statement

4.13 The return Statement

Review Questions

5 Object-Oriented Programming

5.1 Implementing Inheritance

5.2 The Object Reference super

5.3 Chaining Constructors Using this() and super()

Review Questions

5.4 Abstract Classes and Methods

Table of Contents

5.5 Final Declarations

Review Questions

5.6 Interfaces

Review Questions

5.7 Arrays and Subtyping

5.8 Reference Values and Conversions

5.9 Reference Value Assignment Conversions

5.10 Method Invocation Conversions Involving References

5.11 Reference Casting and the instanceof Operator

5.12 Polymorphism

Review Questions

5.13 Enum Types

5.14 Record Classes

5.15 Sealed Classes and Interfaces

Review Questions

6 Access Control

6.1 Design Principle: Encapsulation

6.2 Java Source File Structure

6.3 Packages

6.4 Searching for Classes on the Class Path

Review Questions

6.5 Access Modifiers

6.6 Scope Rules

6.7 Implementing Immutability

Review Questions

7 Exception Handling

7.1 Stack-Based Execution and Exception Propagation

7.2 Exception Types

7.3 Exception Handling: try, catch, and finally

7.4 The throw Statement

Table of Contents

7.5 The throws Clause

Review Questions

7.6 The Multi-catch Clause

7.7 The try-with-resources Statement

7.8 Advantages of Exception Handling

Review Questions

8 Selected API Classes

8.1 Overview of the java.lang Package

8.2 The Object Class

8.3 The Wrapper Classes

Review Questions

8.4 The String Class

8.5 The StringBuilder Class

Review Questions

8.6 The Math Class

8.7 The Random Class

8.8 Using Big Numbers

Review Questions

9 Nested Type Declarations

9.1 Overview of Nested Type Declarations

9.2 Static Member Types

9.3 Non-Static Member Classes

Review Questions

9.4 Local Classes

9.5 Static Local Types

9.6 Anonymous Classes

Review Questions

10 Object Lifetime

10.1 Garbage Collection

10.2 Reachable Objects

Table of Contents

10.3 Facilitating Garbage Collection

10.4 Invoking Garbage Collection Programmatically

Review Questions

10.5 Initializers

10.6 Field Initializer Expressions

10.7 Static Initializer Blocks

10.8 Instance Initializer Blocks

10.9 Constructing Initial Object State

Review Questions

11 Generics

11.1 Introducing Generics

11.2 Generic Types and Parameterized Types

11.3 Collections and Generics

11.4 Wildcards

11.5 Using References of Wildcard Parameterized Types

11.6 Bounded Type Parameters

11.7 Generic Methods and Constructors

11.8 Implementing a Simplified Generic Stack

Review Questions

11.9 Wildcard Capture

11.10 Flexibility with Wildcard Parameterized Types

11.11 Type Erasure

11.12 Implications for Overloading and Overriding

11.13 Limitations and Restrictions on Generic Types

Review Questions

12 Collections, Part I: ArrayList<E>

12.1 Lists

12.2 Declaring References and Constructing ArrayLists

12.3 Modifying an ArrayList<E>

12.4 Querying an ArrayList<E>

Table of Contents

12.5 Iterating Over an ArrayList<E>

12.6 Converting an ArrayList<E> to an Array

12.7 Creating List Views

12.8 Arrays versus ArrayLists

Review Questions

13 Functional-Style Programming

13.1 Functional Interfaces

13.2 Lambda Expressions

13.3 Lambda Expressions and Anonymous Classes

Review Questions

13.4 Overview of Built-In Functional Interfaces

13.5 Suppliers

13.6 Predicates

13.7 Consumers

13.8 Functions

13.9 Two-Arity Specialization of Function<T, R>: BiFunction<T, U, R>

13.10 Extending Function<T, T>: UnaryOperator<T>

13.11 Extending BiFunction<T, T, T>: BinaryOperator<T>

13.12 Currying Functions

13.13 Method and Constructor References

13.14 Contexts for Defining Lambda Expressions

Review Questions

14 Object Comparison

14.1 The Objects Class

14.2 Implementing the equals() Method

14.3 Implementing the hashCode() Method

14.4 Implementing the java.lang.Comparable<E> Interface

14.5 Implementing the java.util.Comparator<E> Interface

Review Questions

15 Collections: Part II

Table of Contents

15.1 The Java Collections Framework

15.2 Collections

15.3 Lists

15.4 Sets

15.5 Sorted Sets and Navigable Sets

15.6 Queues

15.7 Deques

Review Questions

15.8 Maps

15.9 Map Implementations

15.10 Sorted Maps and Navigable Maps

Review Questions

15.11 The Collections Class

15.12 The Arrays Class

Review Questions

16 Streams

16.1 Introduction to Streams

16.2 Running Example: The CD Record Class

16.3 Stream Basics

16.4 Building Streams

16.5 Intermediate Stream Operations

16.6 The Optional Class

16.7 Terminal Stream Operations

16.8 Collectors

16.9 Parallel Streams

Review Questions

17 Date and Time

17.1 Date and Time API Overview

17.2 Working with Dates and Times

17.3 Using Temporal Units and Temporal Fields

Table of Contents

17.4 Working with Instants

17.5 Working with Periods

17.6 Working with Durations

17.7 Working with Time Zones and Daylight Savings

17.8 Converting Date and Time Values to Legacy Date

Review Questions

18 Localization

18.1 Using Locales

18.2 Properties Files

18.3 Bundling Resources

Review Questions

18.4 Core API for Formatting and Parsing of Values

18.5 Formatting and Parsing Number, Currency, and Percentage Values

18.6 Formatting and Parsing Date and Time

18.7 Formatting and Parsing Messages

Review Questions

19 Java Module System

19.1 Making the Case for Modules

19.2 The Modular JDK

19.3 Module Basics

19.4 Overview of Module Directives

19.5 Creating a Modular Application

19.6 Compiling and Running a Modular Application

19.7 Creating JAR Files

19.8 Open Modules and the opens Directive

19.9 Services

19.10 Creating Runtime Images

19.11 Categories of Modules

19.12 Migrating to Modules

19.13 Exploring Modules

Table of Contents

19.14 Summary of Selected Operations with the JDK Tools

Review Questions

20 Java I/O: Part I

20.1 Input and Output

20.2 Byte Streams: Input Streams and Output Streams

20.3 Character Streams: Readers and Writers

20.4 The Console Class

Review Questions

20.5 Object Serialization

Review Questions

21 Java I/O: Part II

21.1 Characteristics of a Hierarchical File System

21.2 Creating Path Objects

21.3 Working with Path Objects

21.4 Operations on Directory Entries

21.5 Reading and Writing Files Using Paths

21.6 Managing File Attributes

21.7 Creating Directory Entries

21.8 Stream Operations on Directory Entries

Review Questions

22 Concurrency: Part I

22.1 Threads and Concurrency

22.2 Runtime Organization for Thread Execution

22.3 Creating Threads

Review Questions

22.4 Thread Lifecycle

22.5 Thread Issues

Review Questions

23 Concurrency: Part II

23.1 Utility Classes TimeUnit and ThreadLocalRandom

Table of Contents

- 23.2 The Executor Framework
- 23.3 The Fork/Join Framework
- 23.4 Writing Thread-Safe Code
- 23.5 Special-Purpose Synchronizers
- 23.6 Synchronized Collections and Maps
- 23.7 Concurrent Collections and Maps

Review Questions

24 Database Connectivity

- 24.1 Introduction to Relational Databases
- 24.2 Introduction to JDBC
- 24.3 Establishing a Database Connection
- 24.4 Creating and Executing SQL Statements
- 24.5 Processing Query Results
- 24.6 Customizing Result Sets
- 24.7 Discovering Database and ResultSet Metadata
- 24.8 Implementing Transaction Control

Review Questions

25 Annotations

- 25.1 Basics of Annotations
- 25.2 Declaring Annotation Types
- 25.3 Applying Annotations
- 25.4 Meta-Annotations
- 25.5 Selected Standard Annotations
- 25.6 Processing Annotations

Review Questions

26 Secure Coding

- 26.1 Application Security Overview
- 26.2 Security Threat Categories
- 26.3 Java Security Policies
- 26.4 Additional Security Guidelines

Table of Contents

Review Questions

A: Taking the Java SE 17 and Java SE 11 Developer Exams

A.1 Preparing for the Exam

A.2 Registering for the Online Proctored Exam

A.3 How the Online Proctored Exam Is Conducted

A.4 The Questions

B: Exam Topics: Java SE 17 Developer

C: Exam Topics: Java SE 11 Developer

D: Annotated Answers to Review Questions

E: Mock Exam: Java SE 17 Developer

F: Annotated Answers to Mock Exam

G: Java Logging API Overview

G.1 Purpose of the Logging API

G.2 Configuring Logging

G.3 Writing Log Messages

G.4 Applying Guarded Logging

G.5 Summary

Part II: Index

A

B

C

D

E

F

G

H

I

J

K

Table of Contents

L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z