



SAMPLE TEST PDF

THE BEGINNING OF A MARVELLOUS JOURNEY IN THE  
CODING WORLD

---

## Java Programming Language

---

*Purpose :*  
Learning

*Author :*  
Louveau Simon

24 mars 2021

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>4</b>
<b>2</b>	<b>The Object-Oriented Approach</b>	<b>5</b>
2.1	Goals . . . . .	5
2.2	Principles . . . . .	5
<b>3</b>	<b>Classes and Objects</b>	<b>6</b>
3.1	Definition . . . . .	6
3.2	Java Programming Language . . . . .	6
3.3	Advantages . . . . .	6
3.4	The Object-Oriented View of Programming . . . . .	6
3.5	Delegating Responsibilities . . . . .	6
<b>4</b>	<b>Classes and Methods</b>	<b>7</b>
4.1	Encapsulation . . . . .	7
4.2	Information Hiding . . . . .	7
4.3	Visibility . . . . .	7
4.4	Class and Instance Variables . . . . .	7
4.5	Class and Instance Methods . . . . .	7
<b>5</b>	<b>The Java Programming Environment</b>	<b>8</b>
5.1	Principles . . . . .	8
5.2	Advantages . . . . .	8
5.3	The Structure . . . . .	8
5.4	Defining A Class . . . . .	8
5.5	Declaring A Variable . . . . .	8
5.6	Primitive Types . . . . .	8
5.7	Reference Types . . . . .	8
5.8	Array Types . . . . .	8
5.9	Declaring A Method . . . . .	8
5.10	Polymorphic Methods . . . . .	8
5.11	The Body Of A Method . . . . .	8
5.12	Local Variables . . . . .	8
5.13	Control Instruction . . . . .	8
5.14	Expressions . . . . .	8
<b>6</b>	<b>Java : Messages, Instantiation, And Initialization Of Objects</b>	<b>9</b>
6.1	Messages . . . . .	9
6.2	Creating Objects . . . . .	9

6.3	Instantiating Arrays . . . . .	9
6.4	Constructors . . . . .	9
6.5	Instantiation Parameters . . . . .	9
6.6	Static Initialization Blocks . . . . .	9
6.7	Destroying Objects . . . . .	9
6.8	Executing A Java Program . . . . .	9
<b>7</b>	<b>Object-Oriented Development Methodology</b>	<b>10</b>
7.1	Development Tasks . . . . .	10
7.2	CRC Cards . . . . .	10
7.3	Example Of Development : Interactive Recipe Book . . . . .	10
7.4	Example 2 : The Eight Queens Puzzle . . . . .	10
7.4.1	Java Implementation . . . . .	10
<b>8</b>	<b>Inheritance</b>	<b>11</b>
8.1	The Class Hierarchy . . . . .	11
8.2	Object Polymorphism . . . . .	11
8.3	Abstract Classes . . . . .	11
8.4	The Substitution Principle . . . . .	11
8.5	Applications Of Inheritance . . . . .	11
8.5.1	Specialization . . . . .	11
8.5.2	Specification . . . . .	11
8.5.3	Construction . . . . .	11
8.5.4	Generalization . . . . .	11
8.5.5	Extension . . . . .	11
8.5.6	Limitation . . . . .	11
8.5.7	Variation . . . . .	11
8.5.8	Combination . . . . .	11
8.6	Inheritance in Java . . . . .	11
8.7	Object Polymorphism in Java . . . . .	11
8.8	Accessing Superclass Elements . . . . .	11
8.9	Inheritance And Visibility . . . . .	11
8.10	Constructors And Inheritance . . . . .	11
8.11	Multiple Inheritance In Java . . . . .	11
<b>9</b>	<b>Exceptions And Packages</b>	<b>12</b>
9.1	Exceptions . . . . .	12
9.2	Exceptions In Object-Oriented Programming . . . . .	12
9.3	Exceptions In Java . . . . .	12
9.4	Handling Exceptions . . . . .	12
9.5	Delegating Exceptions . . . . .	12

9.6	Triggering Exceptions . . . . .	12
9.7	Example : Statistics Application . . . . .	12
<b>10</b>	<b>Cloning, Equivalence Checking, And Serialization</b>	<b>13</b>
10.1	Copying Objects . . . . .	13
10.2	Cloning . . . . .	13
10.3	Deep Cloning . . . . .	13
10.4	Cloning In Java . . . . .	13
10.5	Example : Stack Of References . . . . .	13

# 1 Introduction

## **2 The Object-Oriented Approach**

### **2.1 Goals**

### **2.2 Principles**

## 3 Classes and Objects

### 3.1 Definition

### 3.2 Java Programming Language

### 3.3 Advantages

### 3.4 The Object-Oriented View of Programming

### 3.5 Delegating Responsibilities

## 4 Classes and Methods

### 4.1 Encapsulation

### 4.2 Information Hiding

### 4.3 Visibility

### 4.4 Class and Instance Variables

### 4.5 Class and Instance Methods



## 5 The Java Programming Environment

### 5.1 Principles

### 5.2 Advantages

### 5.3 The Structure

### 5.4 Defining A Class

### 5.5 Declaring A Variable

### 5.6 Primitive Types

### 5.7 Reference Types

### 5.8 Array Types

### 5.9 Declaring A Method

### 5.10 Polymorphic Methods

### 5.11 The Body Of A Method

### 5.12 Local Variables

### 5.13 Control Instruction

### 5.14 Expressions

## **6 Java : Messages, Instantiation, And Initialization Of Objects**

### **6.1 Messages**

### **6.2 Creating Objects**

### **6.3 Instantiating Arrays**

### **6.4 Constructors**

### **6.5 Instantiation Parameters**

### **6.6 Static Initialization Blocks**

### **6.7 Destroying Objects**

### **6.8 Executing A Java Program**

## 7 Object-Oriented Development Methodology

### 7.1 Development Tasks

### 7.2 CRC Cards

### 7.3 Example Of Development : Interactive Recipe Book

### 7.4 Example 2 : The Eight Queens Puzzle

#### 7.4.1 Java Implementation

## 8 Inheritance

### 8.1 The Class Hierarchy

### 8.2 Object Polymorphism

### 8.3 Abstract Classes

### 8.4 The Substitution Principle

### 8.5 Applications Of Inheritance

#### 8.5.1 Specialization

#### 8.5.2 Specification

#### 8.5.3 Construction

#### 8.5.4 Generalization

#### 8.5.5 Extension

#### 8.5.6 Limitation

#### 8.5.7 Variation

#### 8.5.8 Combination

### 8.6 Inheritance in Java

### 8.7 Object Polymorphism in Java

### 8.8 Accessing Superclass Elements

### 8.9 Inheritance And Visibility

### 8.10 Constructors And Inheritance

### 8.11 Multiple Inheritance In Java

## 9 Exceptions And Packages

### 9.1 Exceptions

### 9.2 Exceptions In Object-Oriented Programming

### 9.3 Exceptions In Java

### 9.4 Handling Exceptions

### 9.5 Delegating Exceptions

### 9.6 Triggering Exceptions

### 9.7 Example : Statistics Application

## 10 Cloning, Equivalence Checking, And Serialization

### 10.1 Copying Objects

### 10.2 Cloning

### 10.3 Deep Cloning

### 10.4 Cloning In Java

### 10.5 Example : Stack Of References