Table of Contents

[1.0 Object Oriented Programming Concepts 3](#_Toc32860950)

[ Basic Hello World Program in Java 3](#_Toc32860951)

[ Data Types & Variables, Primitive Data types , Wrapper classes 3](#_Toc32860952)

[ What is an Object 3](#_Toc32860953)

[ What is a class 3](#_Toc32860954)

[ Constructor in Java 3](#_Toc32860955)

[ Object Oriented Programming Features 3](#_Toc32860956)

[ Abstraction (class) 3](#_Toc32860957)

[ Encapsulation 3](#_Toc32860958)

[ Inheritance 3](#_Toc32860959)

[ Polymorphism (Method Overloading / Method Overriding) 3](#_Toc32860960)

[2.0. Abstract Class and Methods 3](#_Toc32860961)

[ Abstract Methods & Non Abstract Methods 3](#_Toc32860962)

[ Extends, Single Inheritance 3](#_Toc32860963)

[ Examples on Abstract class 3](#_Toc32860964)

[3.0. Interface in Java 3](#_Toc32860965)

[ Implements, Single Inheritance &, Multiple Inheritance 3](#_Toc32860966)

[ Difference between Abstract & Interface Class 3](#_Toc32860967)

[ When to use Abstract & Interface classes 3](#_Toc32860968)

[ Examples of Interface 3](#_Toc32860969)

[3.0. Conditional Statements 3](#_Toc32860970)

[ If Else condition 3](#_Toc32860971)

[ Nested if else condition 3](#_Toc32860972)

[ Switch case statement 3](#_Toc32860973)

[4.0. Loop Statements 3](#_Toc32860974)

[ While loop 3](#_Toc32860975)

[ Do- While 3](#_Toc32860976)

[ For Loop 3](#_Toc32860977)

[ Foreach loop 3](#_Toc32860978)

[5.0. Key words 3](#_Toc32860979)

[ Break 3](#_Toc32860980)

[ Continue 3](#_Toc32860981)

[6.0. Arrays 3](#_Toc32860982)

[ What is an Array 3](#_Toc32860983)

[ Advantages of Array 3](#_Toc32860984)

[ Examples on Array 3](#_Toc32860985)

[7.0. Packages, Classes & Objects 4](#_Toc32860986)

[ What are packages 4](#_Toc32860987)

[ Creating packages, Class & Objects 4](#_Toc32860988)

[ Types of Methods, User defined methods 4](#_Toc32860989)

[ Return multiple values from a method 4](#_Toc32860990)

[ Return single value from a method 4](#_Toc32860991)

[8.0. Polymorphism 4](#_Toc32860992)

[ What is Polymorphism 4](#_Toc32860993)

[ Method Overloading 4](#_Toc32860994)

[ Method Overriding 4](#_Toc32860995)

[9.0. Collections 4](#_Toc32860996)

[ Hash Map 4](#_Toc32860997)

[ Hash Table 4](#_Toc32860998)

[ Linked Hash Map 4](#_Toc32860999)

[ Tree Map 4](#_Toc32861000)

[ List 4](#_Toc32861001)

[ Array List 4](#_Toc32861002)

[ Hash Set 4](#_Toc32861003)

[ Linked Hash Set 4](#_Toc32861004)

[ Tree Set 4](#_Toc32861005)

[ Array List 4](#_Toc32861006)

[ Linked List 4](#_Toc32861007)

[ Vector 4](#_Toc32861008)

[10.0. Exception Handling 4](#_Toc32861009)

[ Checked Exception & Unchecked Exception 4](#_Toc32861010)

[ Tree Map 4](#_Toc32861011)

[ Try Catch Finally 4](#_Toc32861012)

[ Final, Finally, Finalize 4](#_Toc32861013)

[ Throw, Throws 4](#_Toc32861014)

[11.0. Strings 4](#_Toc32861015)

[ String 4](#_Toc32861016)

[ String Buffer 4](#_Toc32861017)

[ String Builder 4](#_Toc32861018)

[12.0. Important Core Java faqs 5](#_Toc32861019)

[ String Reverse 5](#_Toc32861020)

[ Concatenate two arrays and print in sorting order 5](#_Toc32861021)

[ Find Max & Min numbers in an array 5](#_Toc32861022)

[ Find Repeated Number or String in an array 5](#_Toc32861023)

[ Explain public static void main 5](#_Toc32861024)

[ Explain Object Oriented concepts in java 5](#_Toc32861025)

[ Static, Non static, default, Abstract & Non Abstract methods 5](#_Toc32861026)

[ Polymorphism (Method Overloading / Method Overriding) 5](#_Toc32861027)

[ Main, Static, Non static, default, Abstract & Non Abstract methods 5](#_Toc32861028)

[ Try, catch, final, finally and Finalize 5](#_Toc32861029)

[ System. Exit(0), System.gc() 5](#_Toc32861030)

## Object Oriented Programming Concepts

### Basic Hello World Program in Java

### Data Types & Variables, Primitive Data types , Wrapper classes

### What is an Object

### What is a class

### Constructor in Java

### Object Oriented Programming Features

### Abstraction (class)

### Encapsulation

### Inheritance

### Polymorphism (Method Overloading / Method Overriding)

## 2.0. Abstract Class and Methods

### Abstract Methods & Non Abstract Methods

### Extends, Single Inheritance

### Examples on Abstract class

## 3.0. Interface in Java

### Implements, Single Inheritance &, Multiple Inheritance

### Difference between Abstract & Interface Class

### When to use Abstract & Interface classes

### Examples of Interface

## 3.0. Conditional Statements

### If Else condition

### Nested if else condition

### Switch case statement

## 4.0. Loop Statements

### While loop

### Do- While

### For Loop

### Foreach loop

## 5.0. Key words

### Break

### Continue

## 6.0. Arrays

### What is an Array

### Advantages of Array

### Examples on Array

## 7.0. Packages, Classes & Objects

### What are packages

### Creating packages, Class & Objects

### Types of Methods, User defined methods

### Return multiple values from a method

### Return single value from a method

## 8.0. Polymorphism

### What is Polymorphism

### Method Overloading

### Method Overriding

## 9.0. Collections

### Hash Map

### Hash Table

### Linked Hash Map

### Tree Map

### List

### Array List

### Hash Set

### Linked Hash Set

### Tree Set

### Array List

### Linked List

### Vector

## 10.0. Exception Handling

### Checked Exception & Unchecked Exception

### Tree Map

### Try Catch Finally

### Final, Finally, Finalize

### Throw, Throws

## 11.0. Strings

### String

### String Buffer

### String Builder

## 12.0. Important Core Java faqs

### String Reverse

### Concatenate two arrays and print in sorting order

### Find Max & Min numbers in an array

### Find Repeated Number or String in an array

### Explain public static void main

### Explain Object Oriented concepts in java

### Static, Non static, default, Abstract & Non Abstract methods

### Polymorphism (Method Overloading / Method Overriding)

### Main, Static, Non static, default, Abstract & Non Abstract methods

### Try, catch, final, finally and Finalize

### System. Exit(0), System.gc()