

# Java Fresher's Training

Apr 16

Name:	
Emp. Id	

## Important: Clause Concerning Security

This handbook is the intellectual property of Cybage Software Pvt. Ltd and is meant for the usage of the intended Cybage employee for training purpose only.

This should not be used for any other purpose or reproduced in any other form without written permission and consent of the concerned authorities.

Prepared by	Sangeeta Joshi <a href="mailto:sangeetaj@cybage.com">sangeetaj@cybage.com</a>
Reviewed by	
For more information, contact Geetanjali Johar (Ext: 7341 )	

*"Education is not the filling of a pail, but the lighting of a fire."*

*— William Butler Yeats (1865-1939)*

# Contents

1.INTRODUCTION TO JAVA INTRODUCTION .....	3
3.FUNDAMENTALS OF JAVA : 2 .....	6
4. OOPS CONCEPTS .....	8
5. ABSTRACT CLASSES & INTERFACES.....	10
6. JAVA MISCELLANEOUS .....	12
7. EXCEPTION HANDLING .....	14
8. COLLECTIONS & GENERICS .....	16
9. IO, SERIALIZATION & NETWORKING.....	18
10. MULTITHREADING .....	20
11. JDBC.....	22
12. JEE OVERVIEW .....	24
13. SERVLETS INTRO .....	25
14. SERVLET FEATURES .....	27
15. SESSION TRACKING .....	29
16. SERVLET FILTERS .....	31
17. JSP INTRO.....	33
18.JSP	35

## Notes for assignments:

- Make assumptions wherever necessary, and mention the same against the question.
- Use Exception Handling wherever required.
- Paste the solution application on the share folder (information will be provided during training).

# 1.Introduction to Java Introduction

## Agenda

- Evolution of java
- History of Java
- Java Buzzwords
- Language Introduction
- Primitive data types

## 2. Fundamentals of Java : 1

### Agenda

- Basic class structure  
Java Coding Conventions
- Accessors & Mutators
- Default Constructor
- Constructor Overloading
- “this”
- Types of variables

## Assignments

### Assignment # 1

Create a class Greeting to print a message: "Hello World"

### Assignment # 2

Create a class Employee & write Accessors & Mutators  
Create objects with default constructor & set values using mutators.  
Display/print values of attributes of each object created.

### Assignment # 3

Write a class Date. Overload the constructor to initialize the attributes.  
Create Date objects using non-parameterized & parameterized constructors both.  
Print values of day, month & year for every object.

## 3.Fundamentals of Java : 2

### Agenda

- Types of variables in Java
- Static
- Memory Layout
- Packages
- Static Imports
- Arrays

## Assignments

### Assignment # 1

Create an array of Employees & display nos. of Employee objects created.

### Assignment # 2

Create a class Calc inside a package “utility”. This class will have a method findRoot() which will return the square root of a given number.

Create another class User1 inside a package allusers. User1 calls the method findRoot() to get square root of a number.

These two packages should be in different directories on two different drives. (may be C: & D:)

## 4. OOPs Concepts

### Agenda

- History of OOPs
- Need of OOP
- Object Characteristics
- Major Pillars of OOP
- Minor Pillars of OOP
- Inheritance – subclassing
- Polymorphism: Static & Dynamic
- Super keyword



## Assignments

### Assignment # 1

Create a class hierarchy: Employee

Manager

SalesManager

Override calculateSalary() method. Manager will have allowance field & SalesManager will have commission field.

### Assignment #2

Create a class: Bank Account

Create subclasses: 1. Saving & 2.Current

Rate of interest is different for different types of account.

## 5. Abstract classes & Interfaces

### Agenda

- Abstract classes
- Interfaces
- Difference between Abstract Classes & Interfaces

Assignments
-------------

## Assignment # 1

Create a class hierarchy :

```
graph TD
    Animal --> Cat
    Animal --> Dog
    Animal --> Lion
```

	Cat	Dog	Lion
Animal class will have methods like <code>respirate()</code> & <code>talk()</code>			

Override method talk() from Animal class in its each subclass  
Create an array of Animal references & observe dynamic polymorphism

## Assignment # 2

Assignment # 2	
Complete class bookwork	Share

Override method talk() from Animal class in its each subclass  
Create an array of Animal references & observe dynamic polymorphism

## Assignment # 2

Assignment # 2	
Complete class bookwork	Share

## Assignment # 2

Create a class hierarchy :

	Shape	
Rectangle	Circle	Triangle

Override method calculateArea() from Shape class in its each subclass

## Create an array of Shape references & observe dynamic polymorphism

### Assignment # 3

Create an interface: Printable – method print()  
Write a class Utilty to have a method printAll() (Printable {})

Write a class Utility to have a method printAll( Printable [])  
Pass different objects (Shapes & Animals) to printAll() to print them.

	1	2	3	4	5	6	7

### Assignment # 3

Create an interface: Printable – method print()

Write a class Utility to have a method printAll( Printable [])  
Pass different objects (Shapes & Animals) to printAll() to print them

Pass different objects (Shapes & Animals) to print in() to print them.

## 6. Java Miscellaneous

### Agenda

- Object class
- Garbage Collector
- Wrapper classes
- Auto boxing & unboxing
- Annotations

Assignments	
<b>Assignment # 1</b>	Override equals(),toString() in Employee & Account class
<b>Assignment # 2</b>	Call System.gc(), check whether GC runs or not
<b>Assignment # 3</b>	Accept numbers from command line arguments & sort them

# 7. Exception Handling

## Agenda

### ➤ Exception

- What are exceptions?
- Difference between Exception and Errors.
- Exception Class Hierarchy.
- Checked and Unchecked Exceptions.

### ➤ Exception Handling

- Try
- Catch
- Finally
- Throw
- Throws

### ➤ User Defined Exception

## Assignments

### Assignment # 1

Create a `NegativeNumberException` to be thrown by the method `findSqrRoot()` if user enters a negative number.

### Assignment # 2

Create an `InsufficientBalanceException` to be thrown by `withdraw()` method in `Account` class if amount to be withdrawn is greater than balance.

**6TFVTV RES SANGEETA**

## 8. Collections & Generics

### Agenda

- Collection
- Set
- List
- Map
- Iterator
- Generics



## Assignments

### Assignment # 1

Create a sorted set of Employee objects.(Sort on **Emp ID**)

### Assignment # 2

Create a list of Employee objects & sort them *by names* in alphabetical order (may be same as above.)

### Assignment # 3

Create a HashMap for books & no.of pages.

Display contents of Map using iterator. (Where Book is a class)?

(Override equals() & hashCode() in book class. .note:equals/hashcode contract should not be violated)

## 9. IO, Serialization & Networking

### Agenda

- I/O Fundamentals
- Stream Classes
- Input and Output Stream
- File I/O
- File Writer
- File Stream
- Scanner
- Intro to Networking
- Client - Server
- Socket
- Port number

## Assignments

### Assignment # 1

Read input data from keyboard & write it to a file "File1".  
Copy contents of File1 to File2.

### Assignment # 2

Serialize & deserialize Account object which will have  
a Date type of field : doc (date of creation)

### Assignment # 3

Create a simple chat application (Client-Server)

# 10. Multithreading

## Agenda

- Thread
- Thread Scheduling
- Thread Lifecycle
- Synchronization
- Deadlock

## Assignments

### Assignment # 1

Two members of a joint Account are trying to withdraw Rs.3000 from the account simultaneously & balance is Rs.5000.

Create a multithreaded program to handle this situation

# 11. JDBC

## Agenda

- Fundamentals of JDBC
- Why we need JDBC?
- What is JDBC?
- JDBC Driver Types
- Driver Manager
- Statement
- Transactions

## Assignments

### Assignment # 1

Insert & update Employee data in database & display it using Prepared Statement.  
Insert some records in Student Table

### Assignment # 2

Create a stored procedure for updating the basic salary of employees as 10%HRA should be added to basic salary .

### Assignment # 3

Write generic code to display column names of a table along with data in the table (same code should work to display records from Emp table and then for Student table

## 12. JEE OVERVIEW

### Agenda

- N tier Architecture
- JEE Overview
- Web Applications



## 13. Servlets Intro

### Agenda

- What is a servlet
- Servlet Lifecycle
- Simple Servlet

## Assignments

### Assignment # 1

Create a simple servlet to print Hello message with User's name

### Assignment # 2

Create a simple servlet demo to demonstrate Servlet's Life cycle

### Assignment 3: CybageNet with BooksReview assignment (Detailed Testcases enclosed as Annex:1)

1. CybageNet Login: a) Administrator Login

b) User Login

2. Library System:

a) Administrator can do following:

- Insert books into the database
- Remove books from the database

b) User can do following:

- Book search (based on Title)
- Show Details of the selected book (Including reviews)
- add a review

## 14. SERVLET Features

### Agenda

- Servlet Config
- Servlet Context
- Request Dispatcher

<b>Assignments</b>
<b>Assignment 2: continue with BookReview assignment</b>

# 15. Session Tracking

## Agenda

- Hidden Form Field
- URL Rewriting
- Cookies
- Http Session

Assignments
<p><b>Assignment 1: continue with BookReview assignment</b></p> <p><b>Assignment # a</b></p> <p>Adminstrator should be able to check how many users are logged in simultaniously.</p>
<p><b>Assignment # b</b> Use cookies to show no. of visitors to the site</p>

## 16. Servlet Filters

### Agenda

- Filters Introduction
- Filter Demo

## Assignments

### Assignment 1: continue with BookReview assignment

Create a simple logging filter to log the information about LogIn timings of the users



## 17. JSP Intro

### Agenda

- JSP Elements
- JSP Scripting
- Implicit Objects

Assignments
<b>Assignment 1: Replace servelets by JSPs in BookReview assignment</b> <b>(Example:Login &amp; Books details page)</b>

# 18.JSP

## Agenda

- JSP standard actions
- Intro to JSTL

<b>Assignments</b>
<b>Assignment 1: continue with BookReview assignment</b> <b>Create JavaBean for Book (BookBean) &amp; Use useBean standard action</b>

# Java Mini Project

## Project Description

Develop a **Dynamic Web Application for Customer Service Dept.:**

A registered customer (Authenticated) is able to carry out following operations:  
(There should be a login page where registered users can login & new users can register)

- a: Launch a new complaint
- b: View status of his complaint
- c: View details of **all customers** who have launched complaints
- d: View details of all **complaints** launched by **a particular customer**

Application should be based on **MVC model II pattern** as under:

**Model** : Java Beans (Use of **VO** -Value Objects)

**View** : JSPs

**Controller** : a Servlet

Use **DAO (Data Access Objects)** pattern for Database Access

Create DBConnector class as **Singleton** for Database Connectivity

Note:

Make assumptions wherever necessary.

Use Exception Handling wherever required.

## Declaration by the Participant My Understanding

Name:					
Employee ID:					
Sr. No	Topic	Theoretical		Practical	
		Yes	No	Yes	No
1	OOPs and Inheritance				
2	Abstract Classes and Interfaces				
3	Java5 Features and Annotations				
4	Exception Handling and Collections Framework				
5	Collections Framework and Generics				
6	IO, Serialization, Networking				
7	Java 7				
8	Java 8				
9	Multithreading & Synchronization				
10	JDBC				
11	JavaEE				
12	Basics of Servlets & its lifecycle				
13	Servlet features, session tracking				
14	Servlet filters				
15	JSP basics & scripting				
16	JENKIN				
17	RESTful Webservices				
18	Spring Framework intro+ IOC/DI Container +Bean definition, xml config				
19	RESTful Webservices				
20	JSP standard actions				
21	MVC architecture, Intro to Spring MVC				
22	JSTL				
23	Spring MVC				
24	Spring Annotations\REST Support				
25	JPA				

26	Glassfish as a server				
27	HTML5				
28	CSS 3				
29	JavaScript				
30	Angular JS				

**Signature of Participant**

<b>For Office Use:</b>		
<b>Signature of Trainer</b>	<b>Signature of Evaluator</b>	<b>Signature of Training Head</b>