### Learning Materials

Learning Playlist Java: <https://www.youtube.com/playlist?list=PLFE2CE09D83EE3E28>

Groovy: <https://groovy-lang.org/documentation.html> , [tutorials](https://www.youtube.com/playlist?list=PLhW3qG5bs-L8T6v6DgsZo93DgYDmOF9u4)

Coding Conventions: [Learn Java Conventions](https://www.oracle.com/java/technologies/javase/codeconventions-introduction.html)

### Topic: OOP Concepts

Part 1: Day 1

Question 1: Shape Hierarchy

Create a hierarchy of shape classes - Shape, Circle, and Rectangle. Implement methods to calculate area and perimeter for each shape. Demonstrate polymorphism by creating an array of Shape objects containing circles and rectangles. Calculate and display the total area and perimeter of all shapes in the array.

Question 2: Employee Inheritance

Create a base class Employee with attributes name and salary. Derive two classes, Manager and Worker, from the Employee class. The Manager class should have an additional attribute bonus. Implement a method to calculate the total salary (salary + bonus) for managers. Create an array of Employee objects containing managers and workers, and display their total salaries.

Question 3: Banking System

Create a class BankAccount with attributes accountNumber and balance. Derive two classes, SavingsAccount and CheckingAccount, from the BankAccount class. Implement methods to deposit and withdraw funds from accounts. Demonstrate polymorphism by creating an array of BankAccount objects containing both savings and checking accounts. Process a series of deposits and withdrawals on the accounts.

Question 4: Vehicle Inheritance

Create a base class Vehicle with attributes make, model, and year. Derive two classes, Car and Motorcycle, from the Vehicle class. Implement a method to display the vehicle information. Create an array of Vehicle objects containing cars and motorcycles, and display their information.

Question 5: Online Shopping

Create a class Product with attributes productName and price. Derive two classes, PhysicalProduct and DigitalProduct, from the Product class. The PhysicalProduct class should have an additional attribute weight. Implement methods to calculate the final price for physical products (including shipping) and digital products. Create an array of Product objects containing both physical and digital products, and display their final prices.

Part: 2 : Day 2

Question 1: Abstract Class and Interface Usage

You are tasked with designing a media player application that can handle both audio and video files. Implement the necessary class structure using abstract classes and interfaces. Your application should be able to play, pause, and stop media files. Design an abstract class MediaPlayer with basic methods and properties shared by both audio and video players. Create interfaces AudioPlayer and VideoPlayer with methods specific to their functionalities. Provide an example of how you would use these abstract classes and interfaces to create instances of audio and video players.

Question 2: Multiple Interfaces and Implementation

You are developing a shape manipulation library. Design a system using abstract classes and interfaces to handle various shapes. Create an abstract class Shape with properties like area and perimeter, along with methods like calculateArea() and calculatePerimeter(). Implement interfaces Drawable (with a draw() method) and Resizable (with a resize(double factor) method) to indicate additional behaviors. Design two specific shape classes – Circle and Rectangle – that inherit from Shape and implement the appropriate interfaces. Provide an example demonstrating how you can create instances of these shapes and utilize their methods.

Question 3: Extending Abstract Classes and Interfaces

You are working on a simulation game where different types of vehicles interact in a virtual city. Create a simulation framework using abstract classes and interfaces. Design an abstract class Vehicle with properties like positionX and positionY, and methods like moveTo(double x, double y) and displayInfo(). Implement an interface Navigable with methods like turnLeft(), turnRight(), and stop(). Design a class Car that extends the Vehicle abstract class and implements the Navigable interface. Additionally, create an interface Flyable with methods like takeOff(), flyTo(double x, double y), and land(). Design a class Airplane that extends the Vehicle abstract class and implements the Flyable interface. Provide an example of using these classes and interfaces within a simulated environment.

### Topic: System Design - SOLID and JAVA Applications

As a java programmer you should be able to understand the following concepts regarding system design and java application.

SOLID Principles

Learn each of the five SOLID principles and provide an example of how they can be applied in the design of a Java application

Caching Strategies

Understand different caching strategies (e.g., in-memory caching, distributed caching) and discuss when each strategy would be suitable.

Provide examples of Java libraries or frameworks that can be used for implementing caching in applications

Singleton Design

How would you implement Singleton design pattern in the following scenario

You are tasked with implementing a Logger class using the Singleton design pattern. The Logger class should have a method logMessage(String message) that prints the provided message along with a timestamp.

Implement the Logger class in such a way that there is only one instance of the Logger class throughout the application's lifecycle, ensuring that any attempts to create new instances result in using the existing instance.

You have to use java code for this task.

### Topic: Exception Handling

Use case: Custom Exception

You are to design a Sign up page for a matrimonial service

The page will ask user to input following information with limitations:

Name: username must be their real name. It should only contain [A-Z,a-z,\s]. Say, Farhan Basher is accepted But Farhan$ Basher2 will throw an error

Age: age must be an integer between 22 to 45.

Nationality: Nationality must be Bangladesh

Contact: Contact must be an 11 digit mobile number.

If a user creates any error in any of the fields, the system will throw an error message. If all data are correctly put, system will show “Congratulations You Are Successful”

You have to use Java Exception handling to create the messages.

### Problem Solving Day 1:

1. You are given an array of 5 integers. Find the highest number and the lowest one
2. You are given a String A & String B. Print true if A contains all the letters B has.

i.e:

A= letter, B= Latter ; Output: false

A=Scared B= Races; Output : True

1. Find all pairs of an integer array whose sum is equal to a given number.
2. Find the missing number in a given integer array of 1 to 100
3. Remove the duplicate elements from a list without any built in method
4. Find the occurrence of characters in a word using 2D array

### Problem Solving Day 2:

Use proper variable and method naming. Naming an integer 'X' is improper and is NOT considered best practice.Optimize code, as much as you can. Avoid using brute force approaches, which increases code complexity. If there is a method for a specific Collection, try using that.

1. Given an Integer Array, let's say int[ ] arr = {10, 32, 1, 8, 32, 92, 41,71,34,64,99}

i) Find the maximum and minimum value of the given array, making sure the line below is in the code System.out.println("The Max value is: " + findMaxValue(arr) + "and Min Value is: " + findMinValue(arr). Basically, you have to complete those 2 unimplemented methods. Cannot use built-in functions

ii) Convert the array to a List, and then print the List using enhanced for loop. Do it in a function called 'convertingToListAndPrint(int[] arr)

1. Let's say this is an ArrayList:

groceryList = ["Eggs","Cheese","Chicken","Milk", "Beef", "Potato","Potato", "Carrot", "Eggs", "Eggs"]

Somehow, the user, noted down some elements twice

i) Implement the user groceryList in Java using ArrayList

ii) Remove the duplicate elements of the list, and print them using enhanced for loop.

iii) The user wants to make sure whether he wrote down Potato in his groceryList. Without using the brute force approach, create a method "isPotatoThere(...)" that returns true, if it exists, else false.

iv) The user realized he mistakenly wrote "Beef" instead of "Mutton". Update the shopping list. Make sure, you mention the line below in your code. System.out.println("The updated List are" + updatedArrayList(...))

1. Construct a dictionary (HashTable/HashMap any one of your wishes), and print them. Each individual can have none, one or more than one address. Code for the given scenario, making the individual name as the unique key

Akib has 3 addresses = Mirpur, Dhanmondi, Shiddheshwari -> Sajeeb has 1 address = Lalmatia

Niloy has 2 addresses = Puran Dhaka, Rajarbag -> Ratul lives abroad has no address

i) Construct the Dictionary, using the scenario above. ii) Print them such that the output is like this

Individual 1: Akib Address 1: Mirpur Address 2: Dhanmondi Address 3: Shiddheshwari

\*The one with no address should show - "NO ADDRESS", but you cannot put 'NO ADDRESS' in the dictionary.

### Case Studies with Java:

#### Create a report card system:

You are to design a report card database. The DB will look like the following:

| Name | Marks | Position |
| --- | --- | --- |
| Sameen | 150 | 2nd |
| Farhan | 150.5 | 1st |
| Mahdi | 150.5 | 1st |
| Tanjila | 100 | 3rd |

The user story will be;

1. As a user if I enter the Name of a student it will show me his Marks and Position.
2. As a user if I enter the position, it will display Students’ names and Marks
3. As a user if I enter marks it displays the students name who got it and nth highest rank of the numbers. If marks is not available then send message “No one got this marks”
4. As a user I can input the student name and Marks only. DB will display the details with position.
5. If the user is sending any wrong input type, the system will throw an error message.

Design a system using Java, where you are to use only Arrays for data storage. Sorting and searching must not take more than nlogn time. Space complexity is n\*n\*n.

#### Sentiment Analysis of Food review:

Read: <https://www.turing.com/kb/natural-language-processing-function-in-ai>

You have to analyze sentiments from given reviews.

System Use case:

From given review entered by user, system will output positive or negative as a result

Output: A review will be taken as positive if the review scores greater than 0 else negative

Algorithm:

Count the sentiment for all sentences. Say, if a review has multiple sentences, the score will be the summation of all sentences. Steps to count sentiments

* Remove pronouns,auxiliary verbs,prepositions,conjunctions.
* Each positive word will be weighted as 1 and negatives will be weighted as -1.
* Use txt files or xls to make the word dictionary.

Example:

The food was good however, the spice mix was not event at all place - Score : 0 ; Negative

N.B: NOT + POSITIVE = Negative and vise versa

Design a system using Java, where you are to use Strings and Regex. Sorting and searching must not take more than linear time. Space complexity is n\*n\*n.

Expectation:

After solving the given algorithm problems, trainees will become familiar with Java primitive types, Strings, Array, Loops, Pattern analysis, file parsing and designing thoughts into codes.

### Topic: HTTP Requests

As a web scraper developer, our job is to get public and private data from a given source. To be able to do so, you must have a clear concept of HTTP requests. You must know how a request is made, how to solve captcha, setting credentials and so on.

For today’s task, you will be having a deep understanding of HTTP in java.

I am providing a simple resource for the learning purpose. The more you know about network data analysis the better you become in ethical hacking.

Resource:

* <https://developer.mozilla.org/en-US/docs/Web/HTTP/Basics_of_HTTP>
* <https://www.baeldung.com/java-9-http-client>
* <https://www.baeldung.com/guide-to-okhttp>
* <https://www.baeldung.com/java-curl>

By the end of the day, you must be able to use Http requests to get HTML code of any website.

#### Problems:

For the given urls get the underlying HTML code of the website.

* 1. <https://www.finra.org/arbitration-mediation/arbitration-awards>
  2. <https://www.nseindia.com/invest/investors-regulatory-actions>
  3. <https://www.accessdata.fda.gov/scripts/ImportRefusals/index.cfm>
  4. <https://www.epa.vic.gov.au/about-epa/public-registers/court-proceedings>
  5. <https://www.dfat.gov.au/international-relations/security/sanctions/consolidated-list>
  6. <https://www.interpol.int/How-we-work/Notices/View-Red-Notices>

### Web scraping with JDK17:

## For the given requirements, you have to capture data using \*\*Groovy and jdk17 libraries. Your output will be stored as XML. Use the XMLSlurper library to create the output file.

Source Key/groovy file name: htmlParsing1

Time : 9 hours

Main URL:

https://www.gov.uk/government/publications/businesses-not-complying-with-money-laundering-regulations-in-2018-to-2019

Scraping Instructions for Relevant Data Fields

From the main source URL capture information from each document labeled by year (2021-2022).

Capture the following fields into the similar corresponding xsd record fields, following natural intuition and conventional logic

· Main Entity Name

· Address

· Entity URL

· Remark

· Event

· Event Date

Main Entity Names

· Capture main entity name from the “Business name” field

Address

o Where no address is listed, default all addresses to Country = United Kingdom

o When address is available, capture from the “Registered address” field and default country to United Kingdom

Entity URL

· Default url to html document for the tax year (2018-2019, 2019-2020, 2020-2021, 2021-2022).

Remark

· Begin with “Penalty Amount:” and capture the fine from the “Penalty amount” field

Event Description

For the event description, begin with “This entity appears on the UK Government Corporate Report published list of businesses that have not complied with the regulations.

Follow with “Description:” and capture from the Description field

Event Date

· There will be no event date

### Task instructions:

1. Go through the requirements thoroughly
2. Use your website analysis knowledge so far
3. Once you have understood every details, start coding
4. Maintain java coding conventions strictly ( [Learn Java Conventions](https://www.oracle.com/java/technologies/javase/codeconventions-introduction.html))
5. Scrape data with clean readable regex
6. There has to be one groovy script only.
7. Output file must be XML

### **Parsing with Framework:**

#### HTML:

List Name: US Commerce Department BIS Unverified Entity List

Source Key: BOIS\_HTML

**TimeLine: 18 hours**

Item Category: Regulator Domestic National Level

Main URL: **https://www.ecfr.gov/current/title-15/part-744**

Scraping Instructions for Relevant Data Fields

From main source URL, go to “**Supplement No. 4 to Part 744—Entity List**”

Capture the following fields into the similar corresponding xsd record fields, following natural intuition and conventional logic

• Main Entity Name

• Alias

• Event

• Address

Main Entity Name

• Capture from “Entity” column

Alias

• Capture from “Entity” column following the main entity name when “a.k.a.” / “alias” appears

o Example: Services GP Tek, a.k.a. ***Nouvelle Option***

Relationship

* Maintain relationship amongst the entities

Address

• Capture from “Entity” column following the main entity name and alias

o Example: Services GP Tek, a.k.a. Nouvelle Option, 1305 Rue Pise, Brossard, QC J4W 2P7, Canada; and 203-760 Rue Galt, Montreal, QC H4G 2P7, Canada; and 6271 Rue Beaulieu, Montreal, QC, H4E 3E9, Canada

Event Description

• All entities will have this boilerplate description, “This entity is subject to the restrictions and requirements outlined in 744.15 of the EAR which involves exports, reexports, and transfers (in-country) involving parties to the transaction who are listed in this supplement.”

Event Date

• Capture from the “Federal Register citation” column

o Example: 80 FR 60532, October 7, 2015

***You will be parsing for Country China and the United Arab Emirates*.**

**Related Readings:**

1. [**https://github.com/sebpo1/scraping/blob/framework\_training/RDCScrapper/development/src/main/groovy/current\_scripts/scrapian\_entity\_creation.groovy**](https://github.com/sebpo1/scraping/blob/framework_training/RDCScrapper/development/src/main/groovy/current_scripts/scrapian_entity_creation.groovy)
2. [**https://github.com/sebpo1/scraping/blob/framework\_training/RDCScrapper/development/src/main/groovy/current\_scripts/address\_parsing.groovy**](https://github.com/sebpo1/scraping/blob/framework_training/RDCScrapper/development/src/main/groovy/current_scripts/address_parsing.groovy)

#### **XLSX Parsing:**

**TimeLine: 15 hours**

**Source key: XLSXParsing**

Source: **https://www.dfat.gov.au/international-relations/security/sanctions/consolidated-list**

***Consolidated List XLS***

Must consolidate references into a single row. Eg. 3, 3a, and 3b should all be in a single row.

Capture:

• Entity Name (if type entity)

• Individual Name (if type individual)

• Alias (all name types except primary name)

• Reference Number (just #)

• Date of Birth

• Place of Birth

• Citizenship

• Address

• Additional Info

• Remarks = [Listing Information + Committees + Control Date] (Combined. Each field labeled with “Field Name:”)

#### HTML Parsing II:

Timeline: 18 hours

Source url: <https://arec.alabama.gov/pages/professionals/legal/disciplinaryactions.aspx>

Scraping Instructions for Relevant Data Fields

From main source URL, capture entities from main data table of disciplinary actions

Capture the following fields into the similar corresponding xsd record fields, following natural intuition and conventional logic

• Main Entity Name

• Alias

• Event

• Address

Main Entity Name

• Capture entity name from “Name” column

Alias

• Capture where aka, fka or dba is used

• Names in () should be captured as alias

• Use of “/” also indicates an alias

Address

• Default all entities to Country = United States and State = Alabama

Event Description

• Default all entities to this boilerplate description “This entity appears on the Alabama Real Estate Commission list of Disciplinary Actions.”

Entity URL

o Capture the link from “Name” column hyperlink as the entity URL

Event Date

• Capture from “Hearing Date” column

Event Coding

• Event coding will be RES / ACT for all entities

You have to capture all the data. Using **load more option.**