***J a va Pr ogra mmi ng La ngua ge***

|  |  |
| --- | --- |
| **Document Code** | **OOP2** |
| **Version** | **1.0** |
| **Effective Date** | **07/05/2024** |

**Hanoi, 05/20**

**Contents**

[Java I/O Fudamentals 4](#_bookmark0)

[Objective: 4](#_bookmark1)

[Business needs: 4](#_bookmark2)

[Technologies 4](#_bookmark4)

[Stored Data 4](#_bookmark5)

[Assignment Specification 4](#_bookmark6)

[Business Rules: 4](#_bookmark7)

[Problem Description: 5](#_bookmark8)

[User Interface Requirement 5](#_bookmark9)

**Concurrency**

# Objective:

» Understand basic concepts of the thread.

» Know how to create a new thread, handling thread priorities.

» Practice with the Life Cycle of a Thread.

» Can enforcing thread synchronization.

» Working with the multi-thread.

# Business needs:

Create a Java console program to manage the candidates.

# Technologies

The product implements one or more technology:

* Object Oriented Programming paradigm: Encapsulation, Overload, Inheritance, Override, Feature Hiding, Polymorphism, Abstract class, Interface.
* Threading, IO.
* Utility class: ArrayList, List, HashMap.

# Stored Data

» All data information will be saved to the file.

# Assignment Specification:

The trainee create package named fa.training.recruiment.candidate contain 2 classes **ExperienceCandidate**

and **FresherCandidate**:

» Each candidate has flowing attributes: first name, last name, birth date, address, phone, email.

» **ExperienceCandidate** has more attributes: years’ experience, professional skill.

» **FresherCandidate** has more attributes: graduation date, graduation rank, education.

» Getter, setter, default constructor and constructor with all attributes.

**Business Rule:**

» Birth date: yyyy (example: 2024)

» Phone: at least 7 positive numbers

» Email: valid email format

» Years’ experience: integer (example: 10)

» Graduation rank: in (Excellence, Good, Fair, Poor)

# Problem Description:

Create 3 concurrent threads:

1. The first thread allows create new object **ExperienceCandidate**, another thread allows create new object **FresherCandidate**, all input information from keyboard.
2. If input wrong any attribute, user must re input until correct. When a new object created, all information must store consecutively in ExperienceCandidates.dat và FresherCandidates.dat. The trainee choose file format (txt or dat).
3. The third thread allow read content of ExperienceCandidates.dat, FresherCandidates.dat and display by table format.

Note: every thread will lock the file when read or write.

# User Interface Requirement

N/A