Hi Viewers,

Thanks for evaluating my exercise and below are the explanations which I did,

Result from Terminal:

```
$ ./gradlew clean test


←[2A←[1m<←[0;1m-------------> 0% CONFIGURING [0s]←[m←[35D←[1B←[1m> root project > Resolve files of :classpath←[m←[44D←[1B←[2A←[1m<←[0;32;1m==←[0;39;1
> :compileJava←[m←[0K←[14D←[1B←[2A←[1m<←[0;32;1m=======←[0;39;1m-------> 50% EXECUTING [0s]←[m←[34D←[1B←[1m> :compileTestKotlin←[m←[20D←[1B←[2A←[1m<←[
←[34D←[1B←[1m> :test > 0 tests completed←[m←[27D←[1B←[2A←[1m<←[0;32;1m============←[0;39;1m--> 90% EXECUTING [1s]←[m←[34D←[2B
←[1A←[1m> :test > Executing test com.wtc.interview.graph.PeopleTest←[m←[59D←[1B←[3A←[0K
←[1m> Task :test←[m←[0K
←[0K
com.wtc.interview.graph.PeopleTest > testExtendedFamilySize ←[32mPASSED←[39m

com.wtc.interview.graph.PeopleTest > testPeopleMapIsCorrectlyPopulated ←[32mPASSED←[39m

com.wtc.interview.graph.PeopleTest > testPeopleRelationships ←[32mPASSED←[39m

←[32;1mBUILD SUCCESSFUL←[0;39m in 2s
7 actionable tasks: 7 executed
←[0K
←[0K
←[0K
←[3A←[1m<←[0;1m-------------> 0% WAITING←[m←[26D←[1B> IDLE←[6D←[1B> IDLE←[6D←[1B←[3A←[2K←[1B←[2K←[1B←[2K←[2A
Dhakshina@LAPTOP-PJ5VFOB3 MINGW64 ~/Documents/family-graph-kandasamy
$
```

I have added 3 Java and 1 Junit classes as below:

1. **Person -> Entity class**
    a. To add name, email, age and to add family relationship

2. **People -> Mail Logic goes here i.e., Family Graph**
    a. To add Relationship between 2 Person
    b. To retrieve Extended Family Size

3. **CSV File Reader - > Util class for reading your 2 CSV files**

    a. readPeople() => Reading input from people.csv
    b. readRelationships() => Reading input from relationships.csv
4. **PeopleTest -> Junit Java test class used to test our application**

Exercise 1:

Implemented code and data structures that read the files and use them to build an in-memory data structure that represents the people in the file and their relationships with each other.
Achieved using Set & Map java util collection

Exercise 2 - Validate correct people loaded

com.wtc.interview.graph.PeopleTest > testPeopleMapIsCorrectlyPopulated

By checking the size of People object

Exercise 3 - Validate correct relationships loaded

com.wtc.interview.graph.PeopleTest > testPeopleRelationships

By testing getRelationshipNumber of each person

Exercise 4 - Write a method that calculates the size of the extended family

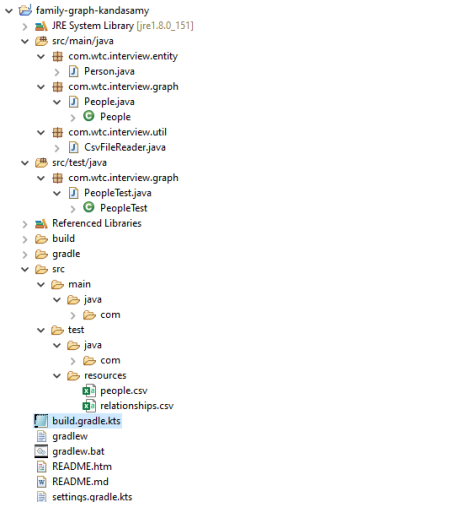
com.wtc.interview.graph.PeopleTest > testExtendedFamilySize

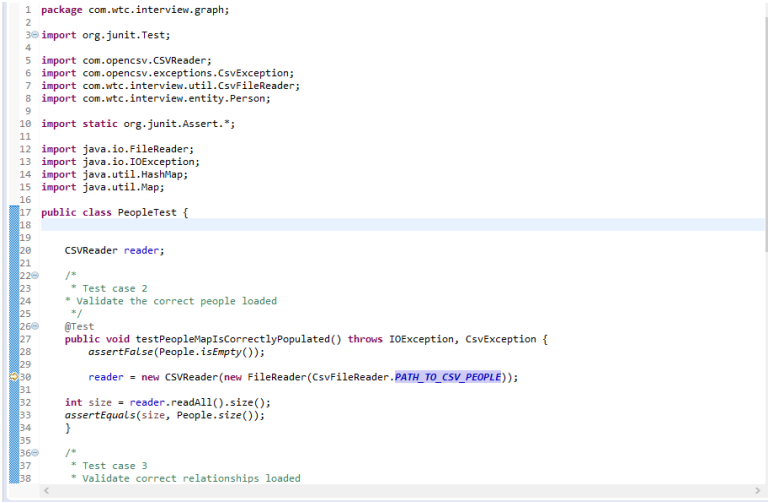
By checking the getExtendedFamilySize of each person

Reference:

```
Gradle Version: 5.2.1
Java Home: C:\Program Files\Java\jdk1.8.0_151
```

Project structure:





```java
package com.wtc.interview.graph;

import org.junit.Test;

import com.opencsv.CSVReader;
import com.opencsv.exceptions.CsvException;
import com.wtc.interview.util.CsvFileReader;
import com.wtc.interview.entity.Person;

import static org.junit.Assert.*;

import java.io.FileReader;
import java.io.IOException;
import java.util.HashMap;
import java.util.Map;

public class PeopleTest {


    CSVReader reader;

    /*
     * Test case 2
    * Validate the correct people loaded
     */
    @Test
    public void testPeopleMapIsCorrectlyPopulated() throws IOException, CsvException {
        assertFalse(People.isEmpty());

        reader = new CSVReader(new FileReader(CsvFileReader.PATH_TO_CSV_PEOPLE));

    int size = reader.readAll().size();
    assertEquals(size, People.size());
    }

    /*
     * Test case 3
     * Validate correct relationships loaded
```