Note: All 3 Questions are compulsory.

Hints:

Keep code simple / concise

Include all binaries needed to run the project

Avoid hardcoding values, especially if hardcoding will prevent us from running your solution locally

Use Page Object Model, where applicable

Check the code into a public GitHub repo and send us the link. The exercises should be in a state such that we can pull from GitHub & execute. Please work independently. Happy coding!

Question 1:

----------

There is a file containing a word and its possible meanings (like a Dictionary). The contents of the file look like this:

Apple – a fruit, a tech firm

Table – an object, contains rows and columns when used in context of computers

Orange – a fruit

Given a path to the file, do the following:

a) Create a method called doesFileExist(String path) which takes the path of the file and tells the user if the file exists at that path or not. Assume all paths are relative to your project structure. If the file does not exist, catch the requisite exception.

 firstly create a file ./src/test/WordMeaning.properties at this location and paste below things in it:

Apple=a fruit, a tech firm

Table=contains rows and columns when used in context of computers

Orange=a fruit

Solution to part a)

public void doesFileExist(){

File f = new File(./src/test/WordMeaning.properties);

try {

if(f.exists()) {

System.out.println("File exists at this location");

}}

 else{

 catch (Exception e) {

      System.out.println("File not found");

    }

}

 }

b) Read each word and its possible meanings and print them out. Your output should look like this:

Word1

Meaning 1

Meaning 2

Word2

Meaning1

Meaning2

Use appropriate data structures wherever necessary.

solution b,

public class Test {

    static void doesfileexit(String filename) {

        File file = new File(filename);

        try {

            BufferedReader br = new BufferedReader(new FileReader(file));

            String line = br.readLine();

            while (line != null) {

                System.out.println("\n");

                for (int i = 0; i < line.length(); i++) {

                    char c = line.charAt(i);

                    if (c == '-' || c == ',') {

                        System.out.println();

                    } else {

                        System.out.print(c);

                    }

                }

                line = br.readLine();

            }

        } catch (Exception e) {

            System.out.println("file not found");

        }

    }

    public static void main(String[] args) {

        System.out.println("Enter the filepath");

        Scanner obj = new Scanner(System.in);

        String filnm = obj.nextLine();

        doesfileexit(filnm);

    }

}

Question 2:

----------

The following exercise does not require user login. Please use ID or class as selectors.

Steps:

1. Navigate to <https://www.weightwatchers.com/us/>

2. Verify loaded page title matches “WW (Weight Watchers): Weight Loss & Wellness Help”

3. On the right corner of the page, click on “Find a Studio”

4. Verify loaded page title contains “Find WW Studios & Meetings Near You | WW USA”

5. In the search field, search for meetings for zip code: 10011

6. Print the title of the first result and the distance (located on the right of location title/name)

7. Click on the first search result and then, verify displayed location name/title matches with the name of the first searched result that was clicked.

8. From this location page, print TODAY’s hours of operation (located towards the bottom of the page)

9. Create a method to print the number of meeting the each person(under the scheduled time) has a particular day of the week

e.g. printMeetings("Sun")

Output should be:

Person A 3

Person B 1

Write an automated test for this scenario using WebDriver.

public static void main(String[] args) throws InterruptedException {

// TODO Auto-generated method stub

System.setProperty("webdriver.chrome.driver",

"C:\\Users\\sharm\\Downloads\\chromedriver\_win32 (1)\\chromedriver.exe");

WebDriver driver = new ChromeDriver();

driver.navigate().to("<https://www.weightwatchers.com/us/>");

String actualTitle = driver.getTitle();

driver.manage().window().maximize();

String expectedTitle = " WW (Weight Watchers): Weight Loss & Wellness Help";

if (actualTitle.equalsIgnoreCase(expectedTitle))

System.out.println("Title Matched");

else

System.out.println("Title didn't match");

driver.navigate().to("<https://www.weightwatchers.com/us/find-a-workshop/>");

String actualTitle1 = driver.getTitle();

System.out.println("actualTitle1 is : "+actualTitle1);

String expectedTitle1 = "Find WW Studios & Meetings Near You | WW USA";

Assert.assertEquals(actualTitle1, expectedTitle1);

if (actualTitle1.equalsIgnoreCase(expectedTitle1))

System.out.println("Title1 Matched");

else

System.out.println("Title1 didn't match");

driver.findElement(By.id("location-search")).sendKeys("10011");

driver.findElement(By.className("rightArrow-daPRP")).click();

// Print the title of first result and distance

String firstResult = driver.findElement(By.className("linkUnderline-1\_h4g")).getText();

System.out.println("firstResult is: " + firstResult);

String firstDistance = driver.findElement(By.className("distance-OhP63")).getText();

System.out.println("firstDistance is: " + firstDistance);

driver.findElement(By.className("linkUnderline-1\_h4g")).click();

Thread.sleep(2000);

// 7. Click on the first search result and then, verify displayed

// location name/title matches with the name of the first searched

// result that was clicked.

String firstResultAfterClickNextPage = driver.findElement(By.className("locationName-1jro\_")).getText();

Assert.assertEquals(firstResultAfterClickNextPage, firstResult);

//8. From this location page, print TODAY’s hours of operation (located towards the bottom of the page)

Calendar calendar = Calendar.getInstance(TimeZone.getDefault());

     int dayOfWeek = calendar.get(Calendar.DAY\_OF\_WEEK);

     System.out.println("dayOfWeek is: "+dayOfWeek);

     String Friday  = driver.findElement(By.xpath("//div[@class='day-NhBOb'][6]")).getText();

if (dayOfWeek==5){

System.out.println("Hours of operation are: "+Friday);

}

driver.close();

driver.quit();

}

Question 3:

----------

Generate 500 random numbers and create a method to print the nth smallest number in a programming language of your choice.

**package** LoginPgm;

**import** java.util.Scanner;

**public** **class** RandomNumbers{

**public** **static** **void** main(String[] args) {

Scanner sc=**new** Scanner(System.***in***);

**int** r=0;

System.***out***.print("Enter the nth number you want from smallest

number");

**int** n=sc.nextInt();

**int** min=(**int**)Math.*random*();;

**int** count=1;

**for**(**int** i=1;i<500;i++){

r=(**int**)Math.*random*();

**if**(r<min)

min=r;

count++;

**if** (count==n)

**break**;

}

System.***out***.print("the smallest number at "+n+"position is" + r);

}

}