Question 1:

Given a string, reverse each word in it and reverse the entire string. For example, if the given string is “hope you are doing well”, then the output should be “llew gniod era uoy epoh”. Assume that all characters in the string are lower case.

Create a method called reverseEverything(String s) where ‘s’ is the input from the user in order to solve this question.

**Package general\_quesion**

*/\*\*  
 \* Created by Sharif on 3/3/2018.  
 \*/*

**public** **class** Quesion\_01 {

**public** **static** **class** Main {

**final** **static** String ***msg*** = "hope you are doing well";

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

System.***out***.println(*reverseEverything*(***msg***));

}

**private** **static** StringBuilder reverseEverything(String s) {

**if**(s.isEmpty())

**return** **null**;

StringBuilder output = **new** StringBuilder();

output.append(s);

output = output.reverse();

**return** output;

}

}

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Question 2:

Below is the source code of a sample webpage which displays a food item along with its servings.

<html>

<head>

<body>

<div class=”itemContent”>

<span class="title ng-binding" ng-bind="food.name">Apple butter</span>

<span class="description ng-binding" ng-bind="food.\_servingDesc">1 Tbsp</span>

<span class="title ng-binding" ng-bind="food.name">Apple(s) fresh</span>

<span class="description ng-binding" ng-bind="food.\_servingDesc">1 medium</span>

<span class="title ng-binding" ng-bind="food.name">Applesauce, unsweetened </span>

<span class="description ng-binding" ng-bind="food.\_servingDesc">1/2 cup(s)</span>

<span class="title ng-binding" ng-bind="food.name">Apple(s), dried</span>

<span class="description ng-binding" ng-bind="food.\_servingDesc">1/4 cup(s)</span>

<span class="title ng-binding" ng-bind="food.name">Juice, Cranberry-apple drink</span>

<span class="description ng-binding" ng-bind="food.\_servingDesc">8 fl oz</span>

</div>

</body>

</head>

</html>

Use your knowledge of WebDriver to fetch the following:

1. Display the third and fifth item from the above list. Your output should be:

Applesauce, unsweetened

Juice, Cranberry-apple drink

1. Fetch each food name and its servings and store them in a Map. Iterate through all the entries in the Map and print them.

**import** org.openqa.selenium.By;  
**import** org.openqa.selenium.WebDriver;  
**import** org.openqa.selenium.WebElement;  
**import** org.openqa.selenium.chrome.ChromeDriver;  
**import** org.testng.annotations.Test;  
**import** java.util.ArrayList;  
**import** java.util.Iterator;  
**import** java.util.List;  
*/\*\*  
 \* Created by Sharif on 3/3/2018.  
 \*/***public class** Quesion\_02 {  
 **public** WebDriver **driver**;  
 @Test  
 **public void** test01(){  
 System.setProperty("webdriver.chrome.driver", "baseUrlForDriver");  
 driver = new ChromeDriver();  
 driver.get("webPageUrl");  
 List<WebElement> listOfFoodItem= driver.findElements(By.className("title\_ng-binding"));  
 Iterator<WebElement> itr = listOfFoodItem.iterator();  
 ArrayList<String> ActualArray = new ArrayList<String>();  
 while(itr.hasNext()){  
 if (ActualArray.size()==3) {  
 ActualArray.add(itr.next().getText());  
 System.out.println(ActualArray);  
 }else if (ActualArray.size()==5){  
 ActualArray.add(itr.next().getText());  
 System.out.println(ActualArray);  
 }  
 }   
 }  
}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Question 3:

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:

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.

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.

public class Question\_03 {

**public void** doesFileExist(String path{  
 **f** = **new** File(System.*getProperty*(**"user.dir\\path\\test.txt"**);  
 **FI** = **new** FileInputStream(**f**);  
 *return path;*  
}

public static void main(String [] args) {

Question\_03 obj = new Question\_03(doesFileExist);

try {

FileReader fileReader = new FileReader(fileName);

BufferedReader bufferedReader = new BufferedReader(fileReader);

while((line = bufferedReader.readLine()) != null) {

System.out.println(line);

}

bufferedReader.close();

}

catch(FileNotFoundException ex) {

System.out.println(

"Unable to open file '" +

fileName + "'");

}

catch(IOException ex) {

System.out.println(

"Error reading file '"

+ fileName + "'");

}

}

}