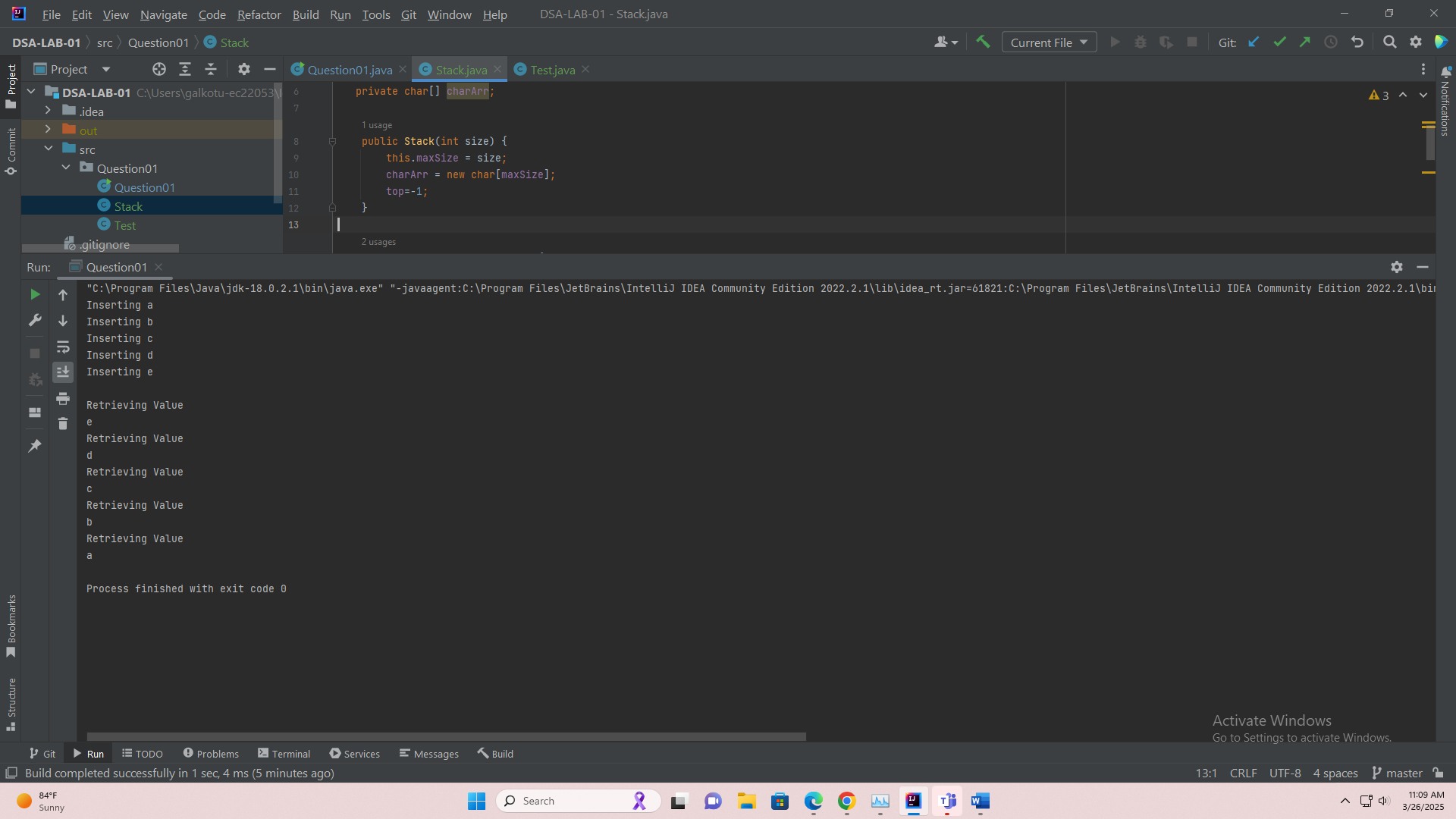
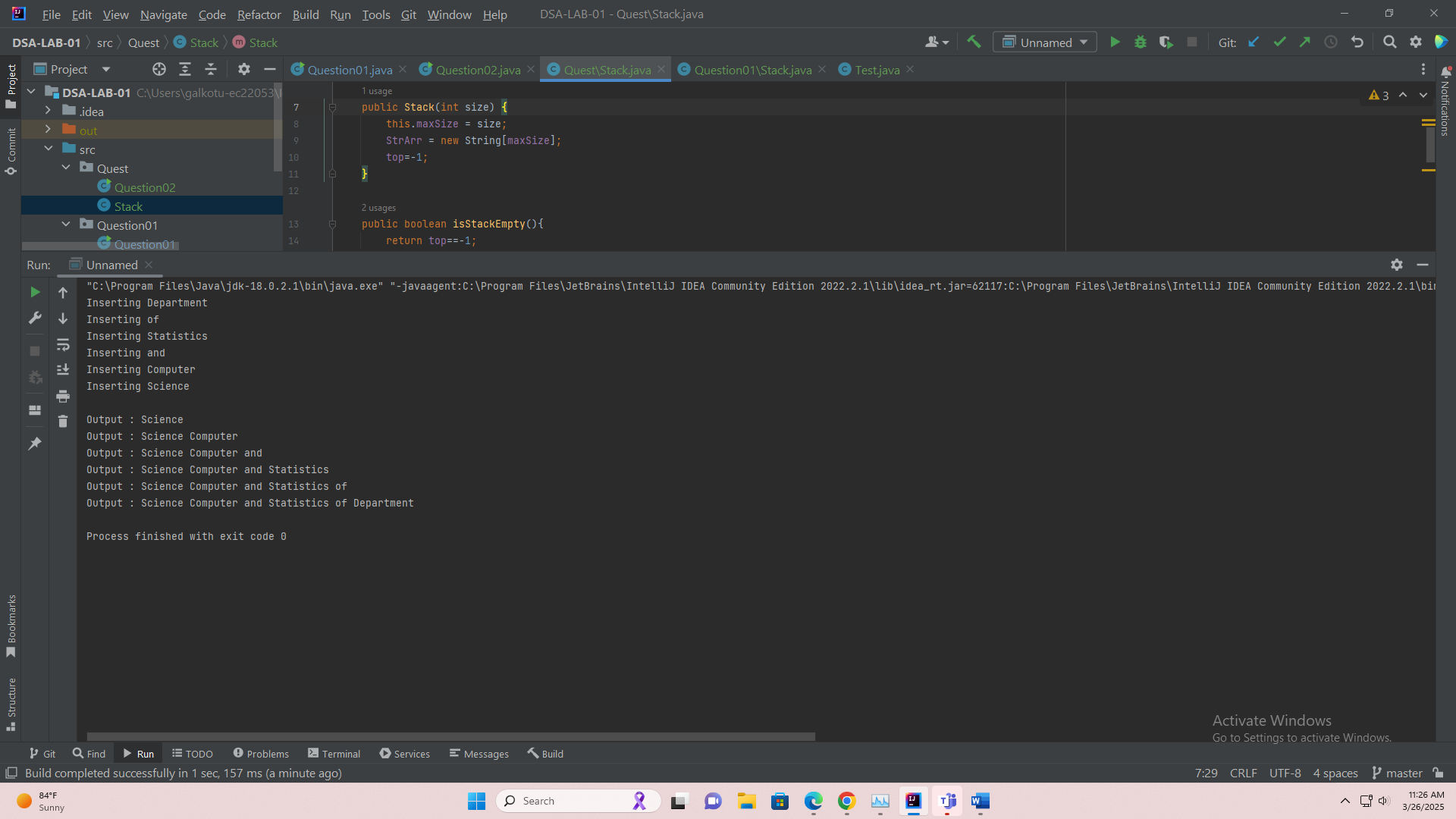
Question 01.

package Question01;  
public class Question01 {  
 public static void main(String[] args) {  
 Stack stackS = new Stack(5);  
 char wordArray[] = {'a','b','c','d','e'};  
 for (char a: wordArray) {  
 stackS.push(a);  
 }  
 System.*out*.println("");  
 while (!stackS.isStackEmpty()){  
 System.*out*.println(stackS.pop());  
 }  
  
 }  
}



package Question01;  
  
public class Stack {  
 private int top;  
 private int maxSize;  
 private char[] charArr;  
  
 public Stack(int size) {  
 this.maxSize = size;  
 charArr = new char[maxSize];  
 top=-1;  
 }  
  
 public boolean isStackEmpty(){  
 return top==-1;  
 }  
  
  
 public boolean isStackFull(){  
 return top==maxSize-1;  
 }  
  
 public int getTop() {  
 return top;  
 }  
  
 public void push(char newValue){  
 if (isStackFull()){  
 System.*out*.println("Stack Already Full");  
 }  
 else {  
 System.*out*.println("Inserting "+newValue);  
 charArr[++top] = newValue;  
 }  
 }  
  
 public char pop(){  
 if (isStackEmpty()){  
 System.*out*.println("Stack is Already Empty");  
 return charArr[top];  
 }  
 else {  
 System.*out*.println("Retrieving Value");  
 return charArr[top--];  
 }  
 }  
  
}

Question 02.

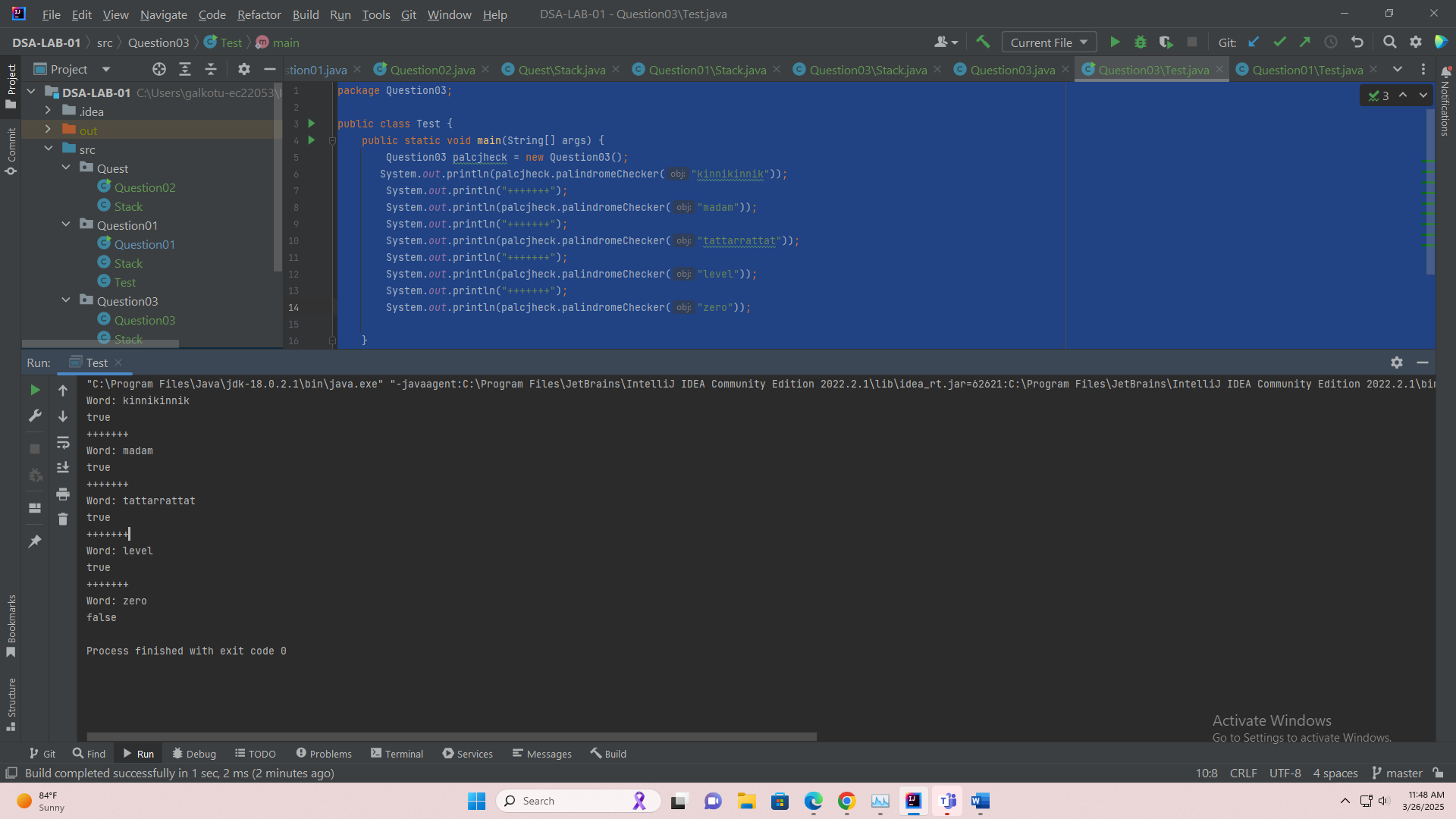
package Question02;  
  
public class Question02 {  
 public static void main(String[] args) {  
 Stack stackS = new Stack(6);  
 String wordArray[] = {"Department", "of", "Statistics", "and", "Computer", "Science"};  
 for (String a : wordArray) {  
 stackS.push(a);  
 }  
 System.*out*.println("");  
 String fOut = "";  
 while (!stackS.isStackEmpty()) {  
 fOut += stackS.pop();  
 fOut += " ";  
 System.*out*.println("Output : " + fOut);  
 }  
  
 }  
}

package Question02;  
public class Stack {  
 private int top;  
 private int maxSize;  
 private String[] StrArr;  
  
 public Stack(int size) {  
 this.maxSize = size;  
 StrArr = new String[maxSize];  
 top=-1;  
 }  
  
 public boolean isStackEmpty(){  
 return top==-1;  
 }  
  
  
 public boolean isStackFull(){  
 return top==maxSize-1;  
 }  
  
 public int getTop() {  
 return top;  
 }  
  
 public void push(String newValue){  
 if (isStackFull()){  
 System.*out*.println("Quest.Question02.Stack Already Full");  
 }  
 else {  
 System.*out*.println("Inserting "+newValue);  
 StrArr[++top] = newValue;  
 }  
 }  
  
 public String pop(){  
 if (isStackEmpty()){  
 System.*out*.println("Quest.Question02.Stack is Already Empty");  
 return StrArr[top];  
 }  
 else {  
 return StrArr[top--];  
 }  
 }  
}

Question 03.

package Question03;  
  
public class Question03 {  
  
 public boolean IsPalindromeOrNot;  
// Write a program to check whether a given string is palindrome using Stack Operations.  
// Example: madam, mom, rotator (Any word that reads the same forward or backward)  
 public boolean palindromeChecker(String obj) {  
 char[] c = obj.toCharArray();  
 Stack stackS = new Stack(c.length);  
 for (char a:obj.toCharArray()){  
 stackS.push(a);  
 }  
 System.*out*.println("Word: "+obj);  
 while (!stackS.isStackEmpty()) {  
 for (char a : obj.toCharArray()) {  
 if (a != stackS.pop()) {  
 IsPalindromeOrNot = false;  
 } else {  
 IsPalindromeOrNot = true;  
 }  
  
 }  
 }  
  
 return IsPalindromeOrNot;  
 }  
  
}

package Question03;  
public class Stack {  
 private int top;  
 private int maxSize;  
 private char[] charArr;  
  
 public Stack(int size) {  
 this.maxSize = size;  
 charArr = new char[maxSize];  
 top=-1;  
 }  
  
 public boolean isStackEmpty(){  
 return top==-1;  
 }  
  
  
 public boolean isStackFull(){  
 return top==maxSize-1;  
 }  
  
 public int getTop() {  
 return top;  
 }  
  
 public void push(char newValue){  
 if (isStackFull()){  
 System.*out*.println("Stack Already Full");  
 }  
 else {  
 // System.out.println("Inserting "+newValue);  
 charArr[++top] = newValue;  
 }  
 }  
  
 public char pop(){  
 if (isStackEmpty()){  
 System.*out*.println("Stack is Already Empty");  
 return charArr[top];  
 }  
 else {  
 // System.out.println("Retrieving Value");  
 return charArr[top--];  
 }  
 }  
  
}

package Question03;  
  
public class Test {  
 public static void main(String[] args) {  
 Question03 palcjheck = new Question03();  
 System.*out*.println(palcjheck.palindromeChecker("kinnikinnik"));  
 System.*out*.println("+++++++");  
 System.*out*.println(palcjheck.palindromeChecker("madam"));  
 System.*out*.println("+++++++");   
 System.*out*.println(palcjheck.palindromeChecker("tattarrattat"));  
 System.*out*.println("+++++++");  
 System.*out*.println(palcjheck.palindromeChecker("level"));  
 System.*out*.println("+++++++");   
 System.*out*.println(palcjheck.palindromeChecker("zero"));  
  
 }  
}