# SOFTWARE DEVELOPMENT

## Software Development

Continuous Assessment 02nd– 06th of December of 2020

Patryck Brenner da Silva Rocha
x20131763@student.ncirl.ie

**Patryck Brenner Da Silva Rocha**
**Student number: 20131763**

Table 1 Section 1 Questions

| Last Digit of Student ID | Section 1 Questions |
|---|---|
| 0 | Q 1A. |
| 1 | Q 1B. |
| 2 | Q 1C. |
| 3 | Q 1D. |
| 4 | Q 1E. |
| 5 | Q 1F. |
| 6 | Q 1G. |
| 7 | Q 1H. |
| 8 | Q 1I. |
| 9 | Q 1J. |

Table 2 Section 2 Questions

| Last Digit of Student ID | Section 2 Questions |
|---|---|
| 8 | Q 2A. |
| 6 | Q 2B. |
| 4 | Q 2C. |
| 2 | Q 2D. |
| 0 | Q 2E. |
| 9 | Q 2F. |
| 7 | Q 2G. |
| 5 | Q 2H. |
| 3 | Q 2I. |
| 1 | Q 2J. |

**Q 1D.** A company has commissioned the development of an application to determine the salary increase to be given to their employees. The application prompts the user to provide an employee's position, the number of years he/she worked in the company and his/her current yearly salary. Next, the application computes the new salary according to the rules from the following table:

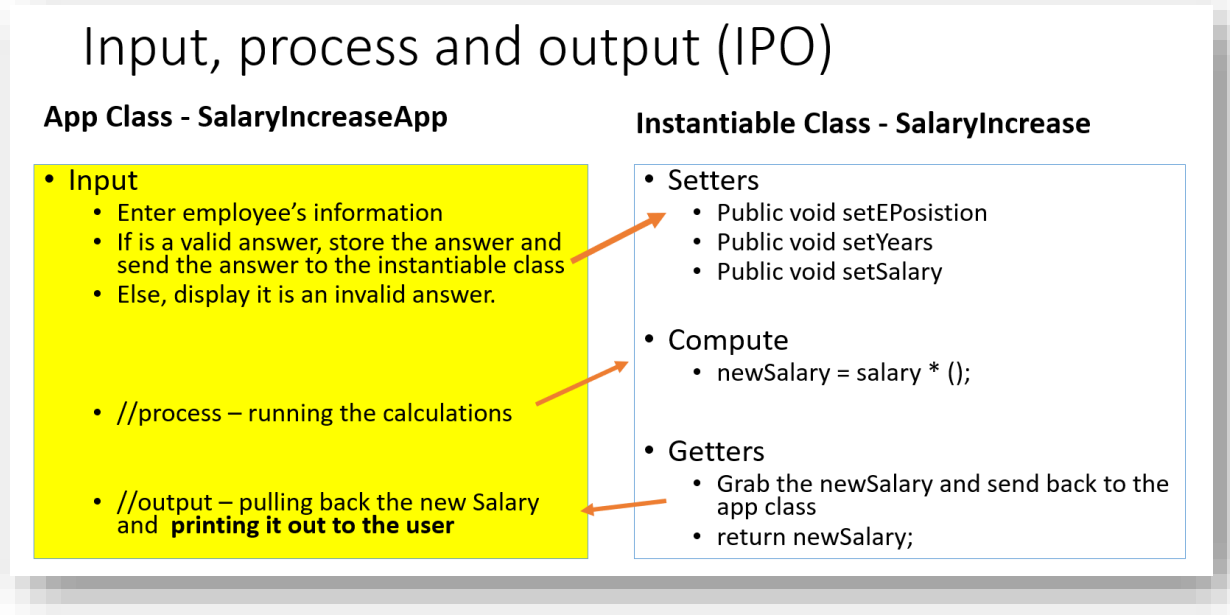
| Position | Number of years worked in the company | Increased salary |
|---|---|---|
| manager | Up to 6 inclusive | 1.02 * salary |
| | Above 6 | 1.03 * salary |
| team leader | Up to 6 inclusive | 1.025 * salary |
| | Above 6 | 1.04 * salary |
| software developer | Up to 6 inclusive | 1.03 * salary |
| | Above 6 | 1.04 * salary |

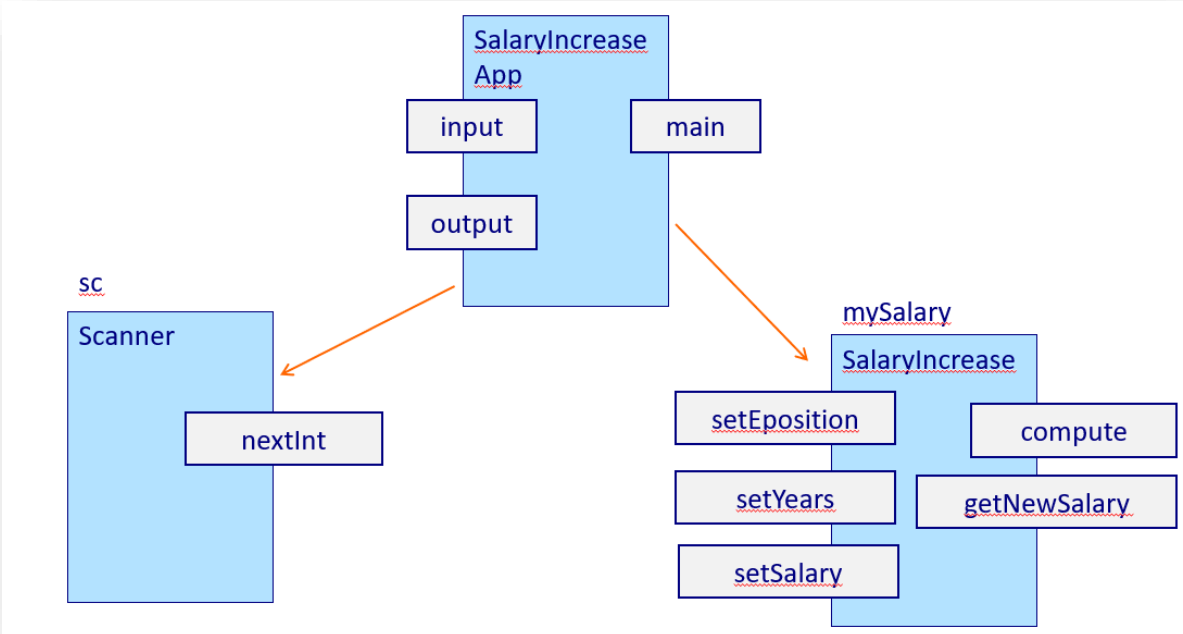
1. Develop an **instantiable class** for this application which contains:
   - A class definition
   - Suitable data members (instance variables)
   - A constructor
   - All necessary setter methods to set the details provided by the user
   - A compute method to determine the new salary
   - A getter method to return the new salary

Name the instantiable class **SalaryIncrease.**

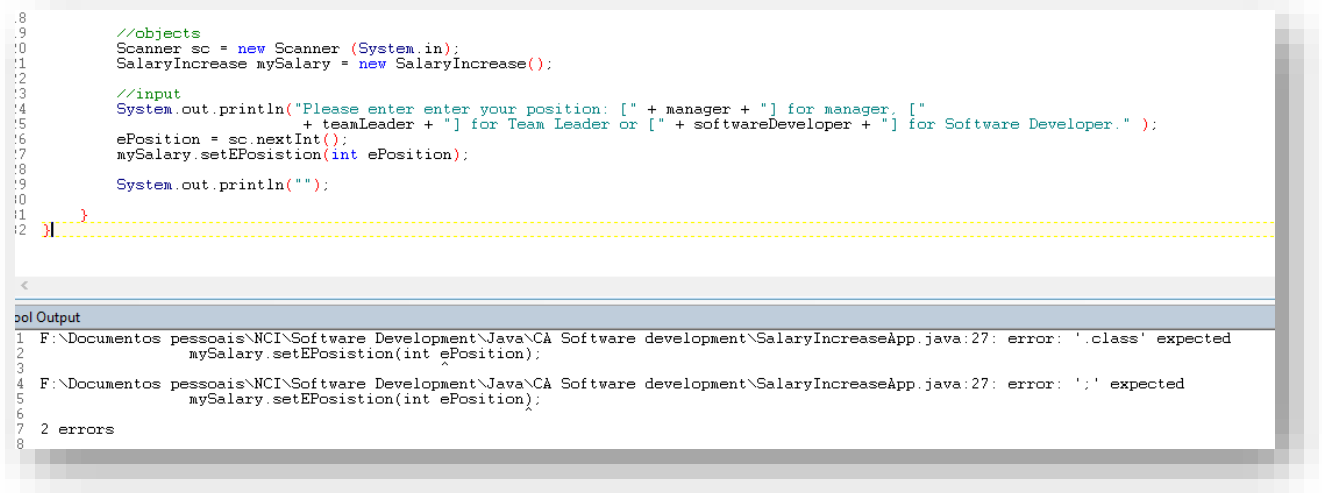
2. **Develop an application** that uses the instantiable class *SalaryIncrease* (the instantiable class previously developed) to calculate an employee's new salary. The application will display the updated salary on the screen. In the application class, please add a short comment for each method of the *SalaryIncrease* class that you use/call in the application to explain why that method is needed. Name the application class **SalaryIncreaseApp**

- **Question 1D: A description of the input, main processing, and output (IPO)**



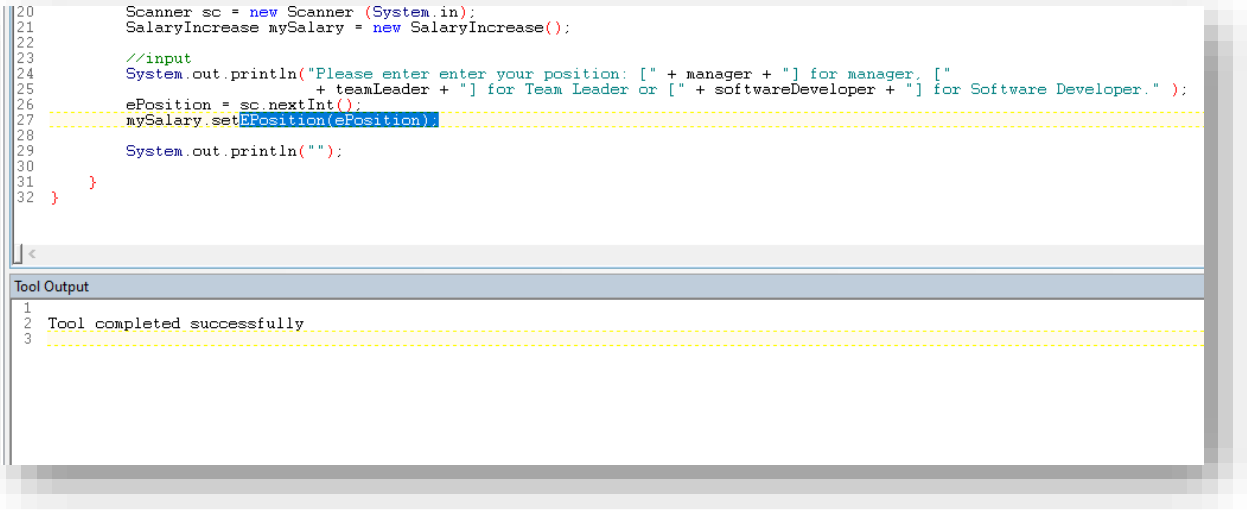
- **Question 1D: A description of five examples of compilation errors and bugs that you have encountered during development and how you fixed them. The report should include screenshots of the five compilation errors and bugs.**



```
        //objects
        Scanner sc = new Scanner (System.in);
        SalaryIncrease mySalary = new SalaryIncrease();

        //input
        System.out.println("Please enter enter your position: [" + manager + "] for manager, ["
                           + teamLeader + "] for Team Leader or [" + softwareDeveloper + "] for Software Developer." );
        ePosition = sc.nextInt();
        mySalary.setEPosistion(int ePosition);

        System.out.println("");

    }
}
```

```
ool Output
F:\Documentos pessoais\NCI\Software Development\Java\CA Software development\SalaryIncreaseApp.java:27: error: '.class' expected
                mySalary.setEPosistion(int ePosition);
                                       ^
F:\Documentos pessoais\NCI\Software Development\Java\CA Software development\SalaryIncreaseApp.java:27: error: ';' expected
                mySalary.setEPosistion(int ePosition);
                                           ^
2 errors
```

**1st Error:** spelling mistake on line 27, supposed to be “mySalary.setEPosistion(ePosistion);”



```
        Scanner sc = new Scanner (System.in);
        SalaryIncrease mySalary = new SalaryIncrease();

        //input
        System.out.println("Please enter enter your position: [" + manager + "] for manager, ["
                           + teamLeader + "] for Team Leader or [" + softwareDeveloper + "] for Software Developer." );
        ePosition = sc.nextInt();
        mySalary.setEPosition(ePosition);

        System.out.println("");

    }
}
```

```
Tool Output

Tool completed successfully
```

**1st Error solution:** the spelling mistake and an extra “int” between the brackets was fixed on line 27

```
27          mySalary.setEPosition(ePosition);
28
29          System.out.println("Please enter the number of years you worked in the company:");
30          years = sc.nextInt();
31          mySalary.setYears(years);
32
33          System.out.println("Please enter your salary: ")
34          salary = sc.nextDouble();
35          mySalary.setSalary(salary);
36
37      }
38  }
```

Tool Output

```
1  F:\Documentos pessoais\NCI\Software Development\Java\CA Software development\SalaryIncreaseApp.java:33: error: ';' expected
2                  System.out.println("Please enter your salary: ")
3                                                                  ^
4  1 error
5
6  Tool completed with exit code 1
7
```

**2nd Error:** on line 33 error: ';' expected

```
28
29          System.out.println("Please enter the number of years you worked in the company:");
30          years = sc.nextInt();
31          mySalary.setYears(years);
32
33          System.out.println("Please enter your salary: ");
34          salary = sc.nextDouble();
35          mySalary.setSalary(salary);
36
37      }
38  }
```

Tool Output

```
1
2  Tool completed successfully
3
```

**2nd Error solution:** fixed by adding the ";" on line 33

```
33
34          System.out.println("Please enter your salary: ");
35          salary = sc.nextDouble();
36          mySalary.setSalary(salary);
37
38          //compute
39          mySalary.compute();
40
41          if (ePosition == manager){
42              position = "Manager";
43          } else if (ePosition == teamLeader){
44              position = "Team Leader";
45          } else {
46              position = "Software Developer";
47          }
48
49          //output
```

Tool Output

```
1  F:\Documentos pessoais\NCI\Software Development\Java\CA Software development\SalaryIncreaseApp.java:43: error: 'else' without 'if'
2                  else if (ePosition == teamLeader){
3                  ^
4  F:\Documentos pessoais\NCI\Software Development\Java\CA Software development\SalaryIncreaseApp.java:54: error: reached end of file while p
5  }
```

**3rd Error:** on line 43, 'else' without 'if'  else if (ePosition == teamLeader){

```
32          mySalary.setYears(years);
33
34          System.out.println("Please enter your salary: ");
35          salary = sc.nextDouble();
36          mySalary.setSalary(salary);
37
38          //compute
39          mySalary.compute();
40
41          if (ePosition == manager){
42              position = "Manager";
43          } else if (ePosition == teamLeader){
44              position = "Team Leader";
45          } else {
46              position = "Software Developer";
47          }
48
49          //output
```

Tool Output

```
F:\Documentos pessoais\NCI\Software Development\Java\CA Software development\SalaryIncreaseApp.java:22: error: cannot find symbol
                SalaryIncreease mySalary = new SalaryIncrease();
                ^
  symbol:   class SalaryIncreease
  location: class SalaryIncreaseApp
1 error

Tool completed with exit code 1
```

**3rd Error solution:** to fix the previous error I realised I forgot the curly bracket on line 43 which was fixed by just adding it. Although it showed me another error, error…

```
21          Scanner sc = new Scanner (System.in);
22          SalaryIncreease mySalary = new SalaryIncrease();
23
24          //input
25          System.out.println("Please enter enter your position: [" + manager + "] for manager, ["
26                                  + teamLeader + "] for Team Leader or [" + softwareDeveloper + "] for Software Developer." );
27          ePosition = sc.nextInt();
28          mySalary.setEPosition(ePosition);
29
30          System.out.println("Please enter the number of years you worked in the company:");
31          years = sc.nextInt();
32          mySalary.setYears(years);
33
34          System.out.println("Please enter your salary: ");
35          salary = sc.nextDouble();
36          mySalary.setSalary(salary);
37
38          //compute
39          mySalary.compute();
40
```

Tool Output

```
F:\Documentos pessoais\NCI\Software Development\Java\CA Software development\SalaryIncreaseApp.java:22: error: cannot find symbol
                SalaryIncreease mySalary = new SalaryIncrease();
                ^
  symbol:   class SalaryIncreease
  location: class SalaryIncreaseApp
1 error
```

**4th Error:** spelling mistake on line 22, cannot find symbol SalaryIncreaease mySalary = new SalaryIncrease();

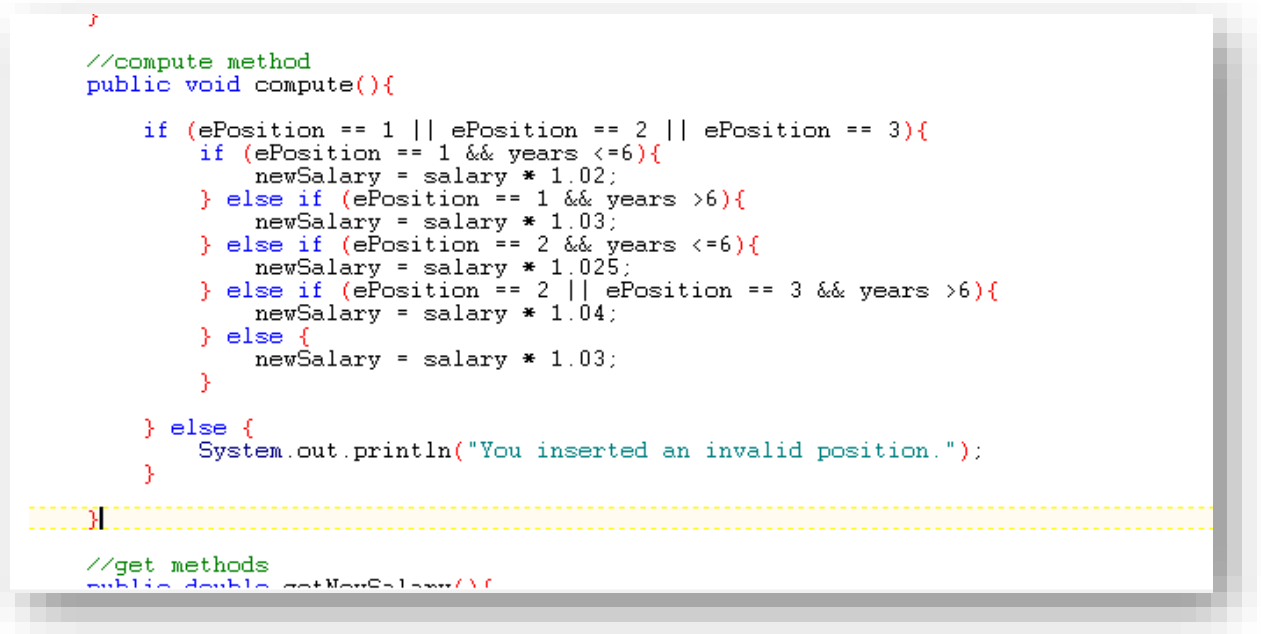
```
15          //constants
16          final int manager = 1;
17          final int teamLeader = 2;
18          final int softwareDeveloper = 3;
19
20          //objects
21          Scanner sc = new Scanner (System.in);
22          SalaryIncrease mySalary = new SalaryIncrease();
23
24          //input
25          System.out.println("Please enter enter your position: [" + manager + "] for manager, ["
26                              + teamLeader + "] for Team Leader or [" + softwareDeveloper + "] for Software Developer." );
27          ePosition = sc.nextInt();
28          mySalary.setEPosition(ePosition);
```

```
Tool Output
1
2  Tool completed successfully
3
```

**4th Error solution:** On line 22, “SalaryIncrease mySalary = new SalaryIncrease();”

```
C:\WINDOWS\system32\cmd.exe
Please enter enter your position: [1] for manager, [2] for Team Leader or [3] for Software Developer.
0
Please enter the number of years you worked in the company:
5
Please enter your salary:
2000
You inserted an invalid position.
Being a Software Developer who worked for 5 years, your new salary will be: 0.0
Press any key to continue . . .
```

**5th Error:** “You inserted an invalid position” should make the app stop but it keep working even though I inserted an invalid position.



```
    }

    //compute method
    public void compute(){

        if (ePosition == 1 || ePosition == 2 || ePosition == 3){
            if (ePosition == 1 && years <=6){
                newSalary = salary * 1.02;
            } else if (ePosition == 1 && years >6){
                newSalary = salary * 1.03;
            } else if (ePosition == 2 && years <=6){
                newSalary = salary * 1.025;
            } else if (ePosition == 2 || ePosition == 3 && years >6){
                newSalary = salary * 1.04;
            } else {
                newSalary = salary * 1.03;
            }

        } else {
            System.out.println("You inserted an invalid position.");
        }

    }

    //get methods
    public double getNewSalary(){
```

**5th Error solution:** I had to extract the first "if statement" from the compute method in the instantiable class and use as a condition on the app class instead. This way if the right position is chosen the rest of the application works normally, else it show a message informing the user it is an invalid position.

**code for instantiable class SalaryIncrease.java Question 1D:**

```
/*
 *SalaryIncrease.java
 *@Patryck Brenner
 *2nd of Dec 2020
*/


public class SalaryIncrease{
        //Variables
        private int ePosition, years;
        private double salary, newSalary;

        //constructor
        public SalaryIncrease(){
```

```
        ePosition = 0;
        years = 0;
        salary = 0.0;
        newSalary = 0.0;
}

//set methods
public void setEPosition(int ePosition){
        this.ePosition = ePosition;
}

public void setYears (int years){
        this.years = years;
}

public void setSalary (double salary){
        this.salary = salary;
}

//compute method
public void compute(){


                if (ePosition == 1 && years <=6){
                        newSalary = salary * 1.02;
                } else if (ePosition == 1 && years >6){
                        newSalary = salary * 1.03;
                } else if (ePosition == 2 && years <=6){
                        newSalary = salary * 1.025;
                } else if (ePosition == 2 || ePosition == 3 && years >6){
```

```
                newSalary = salary * 1.04;
            } else {
                newSalary = salary * 1.03;
            }

    }

    //get methods
    public double getNewSalary(){
        return newSalary;
    }


}
```

**code for main class SalaryIncreaseApp.java Question 1D:**

```
/*
 *SalaryIncreaseApp.java
 *@Patryck Brenner
 *2nd of Dec 2020
*/

import java.util.*;
public class SalaryIncreaseApp{
    public static void main (String args[]){
        //variables
        int ePosition, years;
        double salary, newSalary;
        String position = "";
```

```
		//constants
		final int manager = 1;
		final int teamLeader = 2;
		final int softwareDeveloper = 3;

		//objects
		Scanner sc = new Scanner (System.in);
		SalaryIncrease mySalary = new SalaryIncrease();

		//input
		System.out.println("Please enter enter your position: [" + manager + "] for
manager, ["
                                                + teamLeader + "] for Team Leader or ["
+ softwareDeveloper + "] for Software Developer." );
		ePosition = sc.nextInt();

		if (ePosition == 1 || ePosition == 2 || ePosition == 3){

			mySalary.setEPosition(ePosition);

			System.out.println("Please enter the number of years you worked in
the company:");
			years = sc.nextInt();
			mySalary.setYears(years);

			System.out.println("Please enter your salary: ");
			salary = sc.nextDouble();
			mySalary.setSalary(salary);

			//compute
```

```
                    mySalary.compute();

                    if (ePosition == manager){
                        position = "Manager";
                    } else if (ePosition == teamLeader){
                        position = "Team Leader";
                    } else {
                        position = "Software Developer";
                    }

                    //output
                    newSalary = mySalary.getNewSalary();
                    System.out.println("Being a " + position + " who worked for " + years +
" years, your new salary will be: " + newSalary);

                }else {
                        System.out.println("You inserted an invalid position.");
                }

        }
}
```

**Q 2I.** The Computing Society asks for your help to create an application to encode sentences. The application prompts the user to enter one piece of text (at least one sentence). The text can contain only letters, spaces (i.e. ' '), dots (i.e. '.'), exclamation marks (i.e. '!') and question marks (i.e. '?'). Each sentence ends with either a dot, an exclamation mark or a question mark. (Please note that you are not required to validate the input). Next, the application uses the given text to create the corresponding encoded text. The encoded text is created using the following rules:

- If the character is a vowel then immediately after the vowel add that vowel's position in the original sentence
- Each space is replaced by a plus character (i.e. '+')
- Each dot is replaced by an exclamation mark (i.e. '!')
- Each question mark (i.e. '?') is replaced by two question marks
- Each exclamation mark (i.e. '!') is replaced by a question mark (i.e. '?') followed by an exclamation mark (i.e. '!')

1. Develop an **instantiable class** for this application which contains:
   A class definition
   Suitable data members (instance variables)
   A constructor
   A setter method to set the given text
   A compute method to create the encoded text
   A getter method to return the encoded text

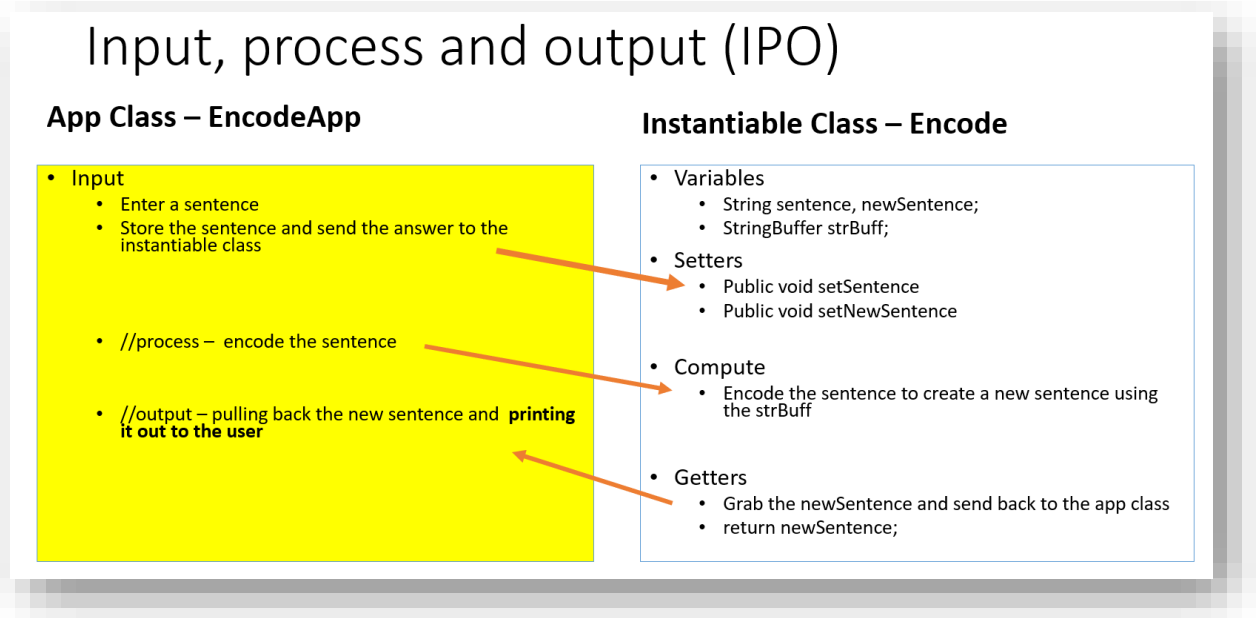
Name the instantiable class **Encoder.**

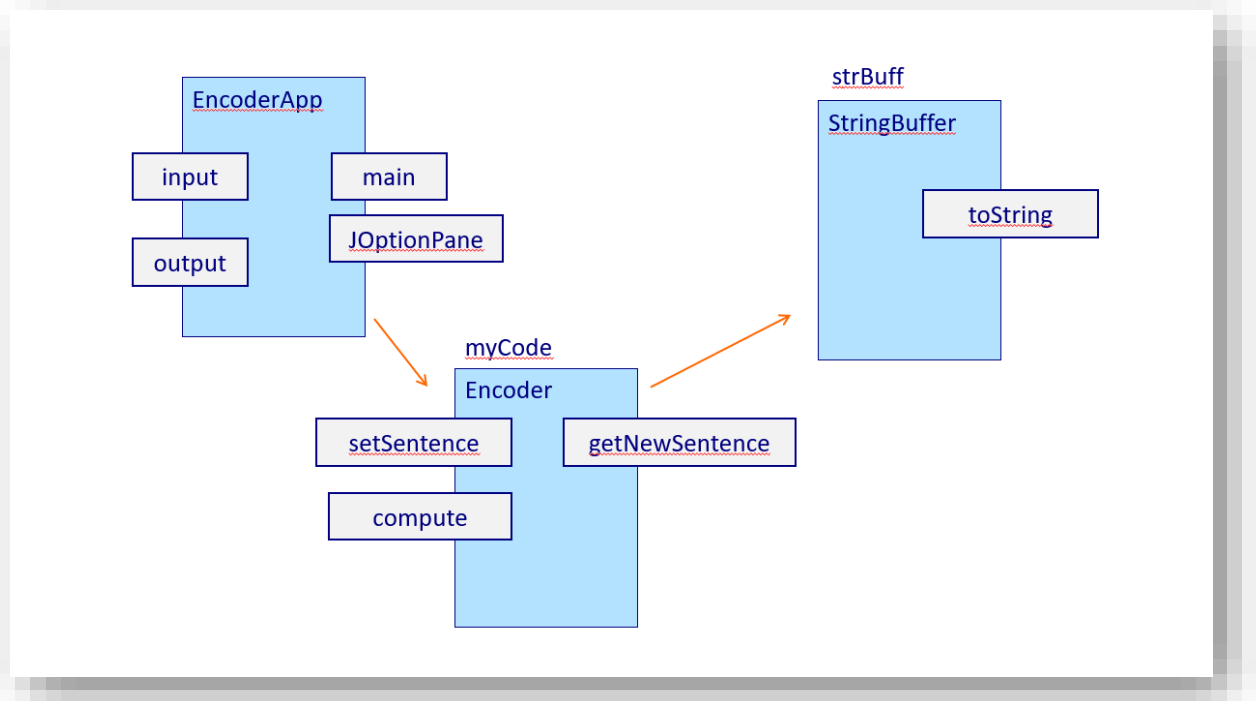
For example, if the instantiable class receives:

- The text "YOU are your best thing." then the compute method should create the encoded text "YO2U3+a5re7+yo10u11r+be15st+thi21ng!"
- The text "Omar learned Java. Did John learn C?" then the compute method should generate the encoded text "O1ma3r+le7a8rne11d+Ja15va17!+Di21d+Jo25hn+le30a31rn+C??"

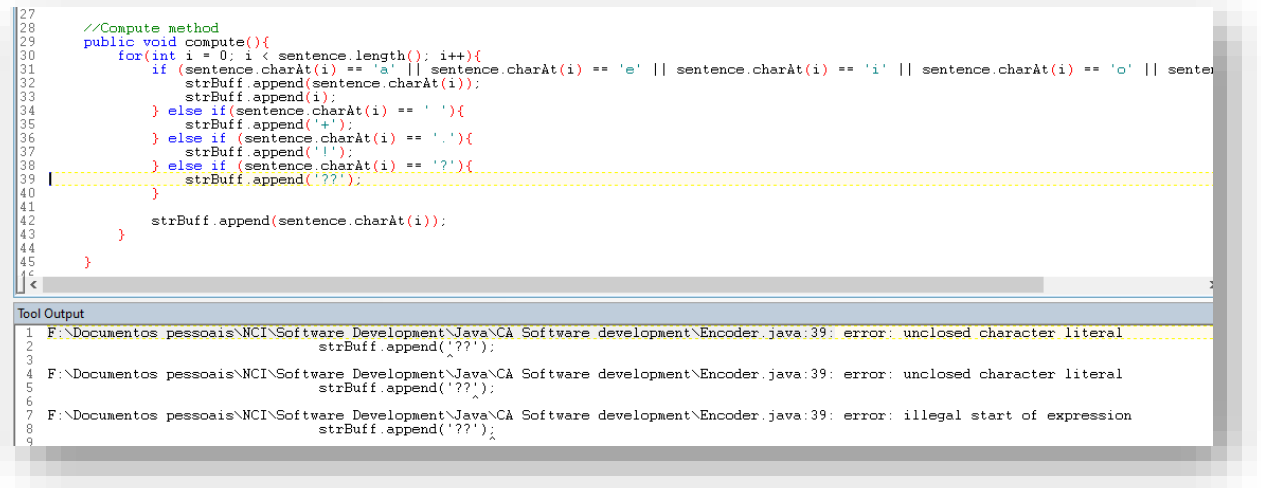
2. **Develop an application** that uses the instantiable class *Encoder* (the instantiable class previously developed) to encode a piece of text. The application will display the encoded text on the screen. In the application class, please add a short comment for each method of the *Encoder* class that you use/call in your application to explain why that method is needed.
Name the application class **EncoderApp.**

- **Question 2I: A description of the input, main processing, and output (IPO)**

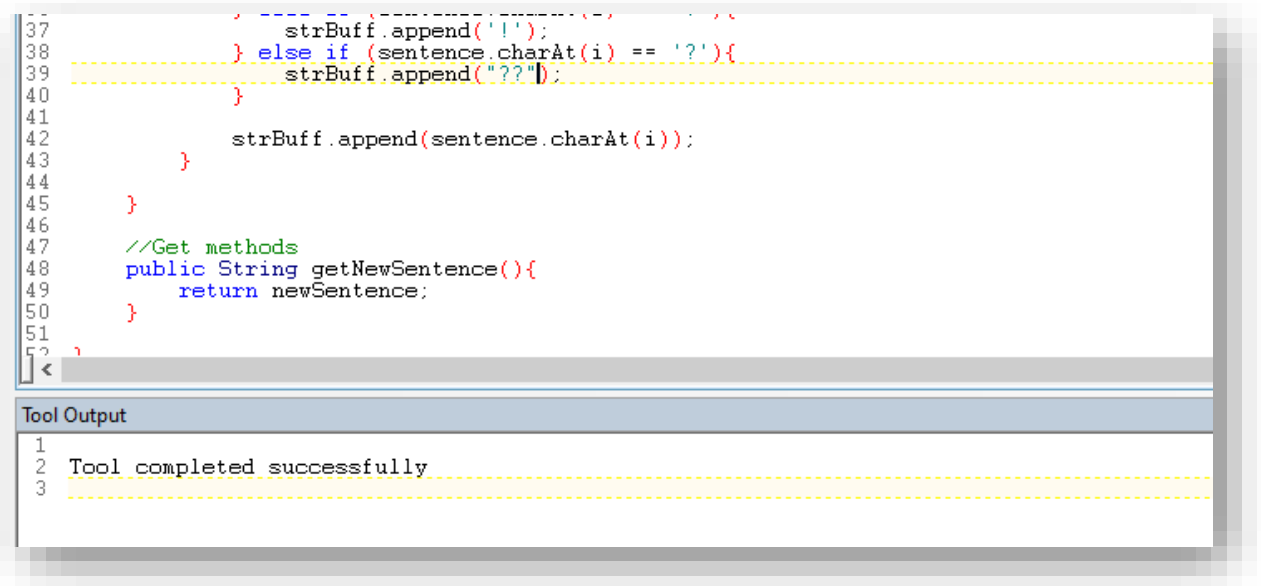


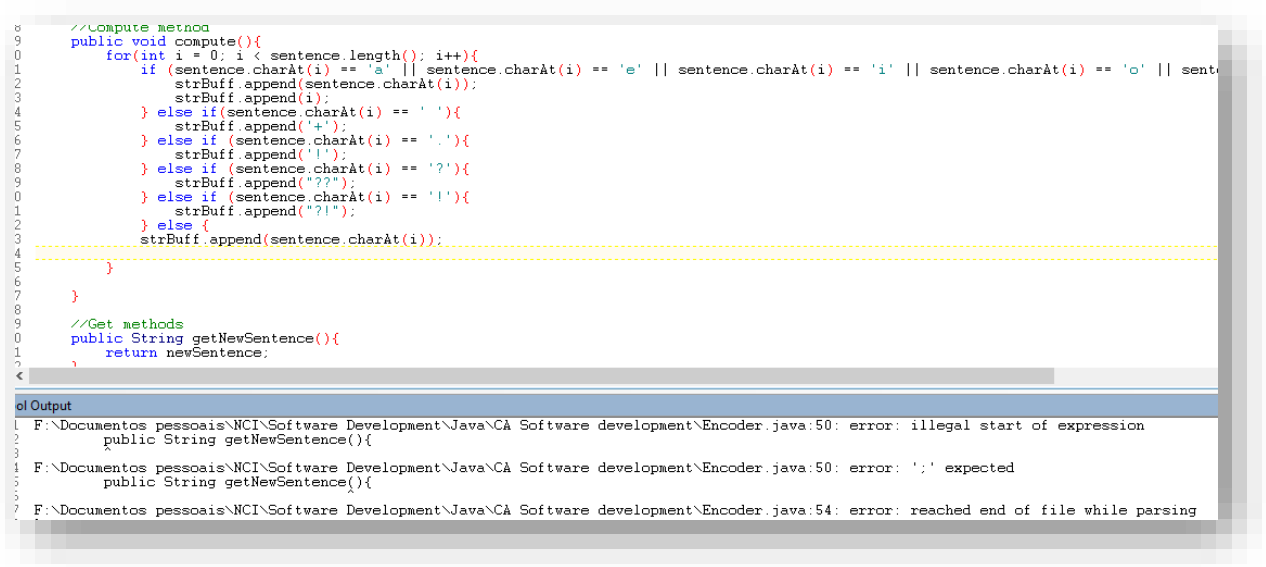
- **Question 2I: A description of five examples of compilation errors and bugs that you have encountered during development and how you fixed them. The report should include screenshots of the five compilation errors and bugs:**



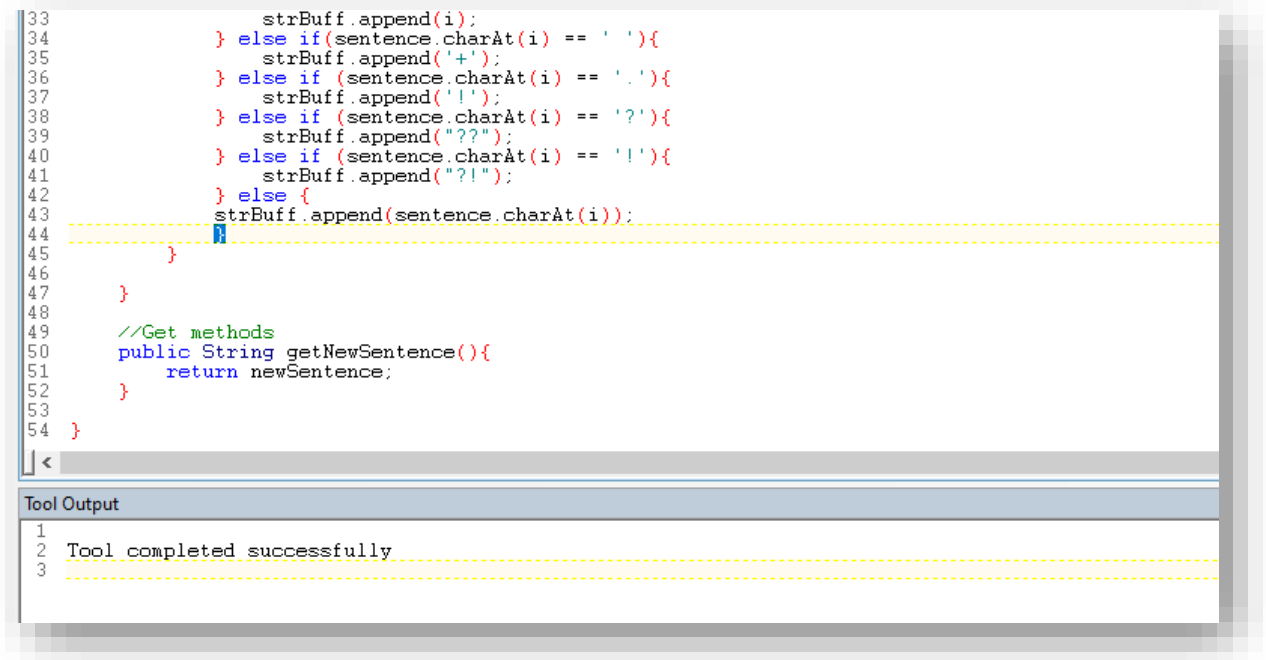

**1st Error:** unclosed character literal strBuff.append('??'); caused 8 more errors



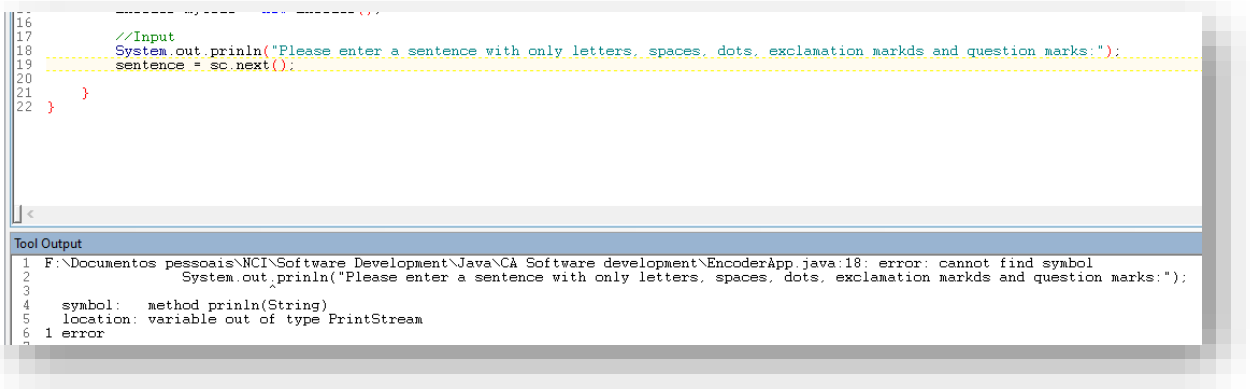

**1st Error solution:** the error was caused because it is not allowed using single quotes when appending more than one word to the StringBuffer. It was solved when I changed the single quotes by double quotes.




**2nd Error:** On the line 54, error: reached end of file while parsing } between other 3 erros

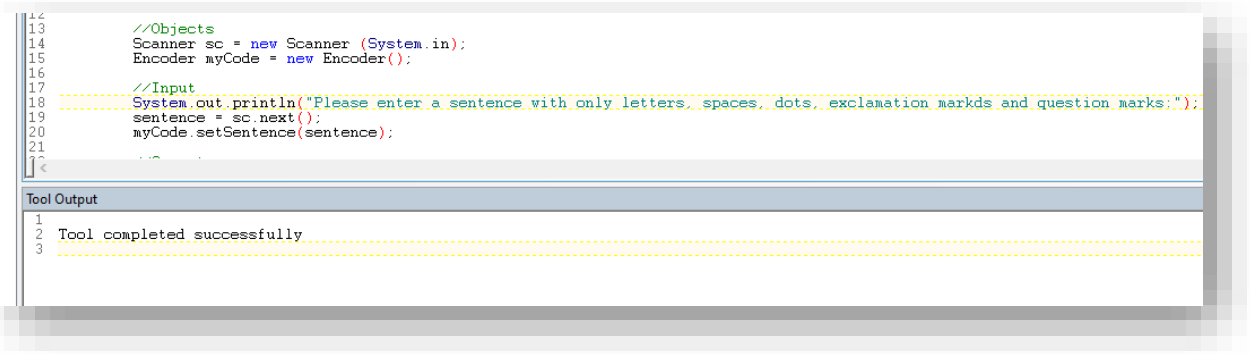



**2nd Error solution:** By closing the curly bracket on the “else” statement on line 44 it managed to solve 2 other false errors caused by it.

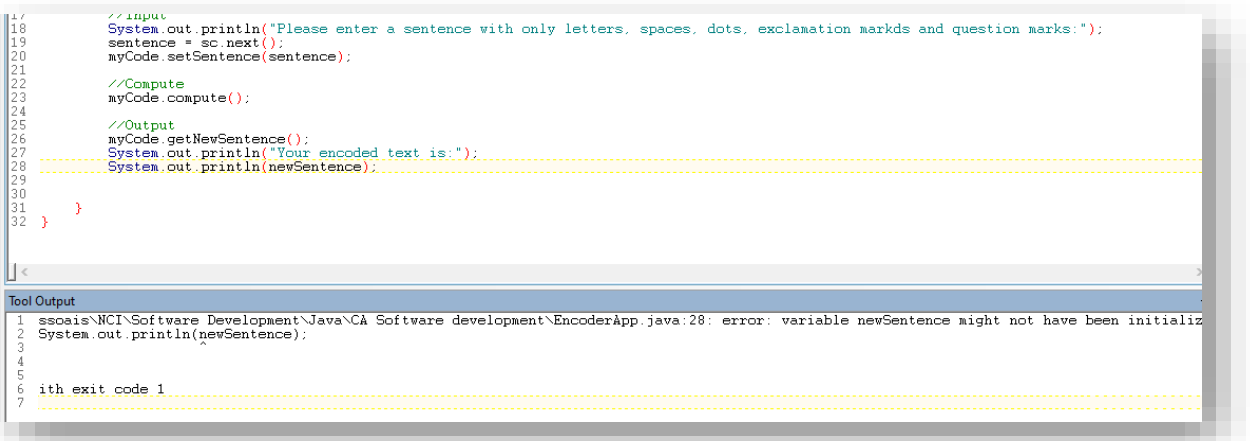



**3rd Error:** cannot find symbol

System.out.prinln("Please enter a sentence with only letters, spaces, dots, exclamation markds and question marks:");

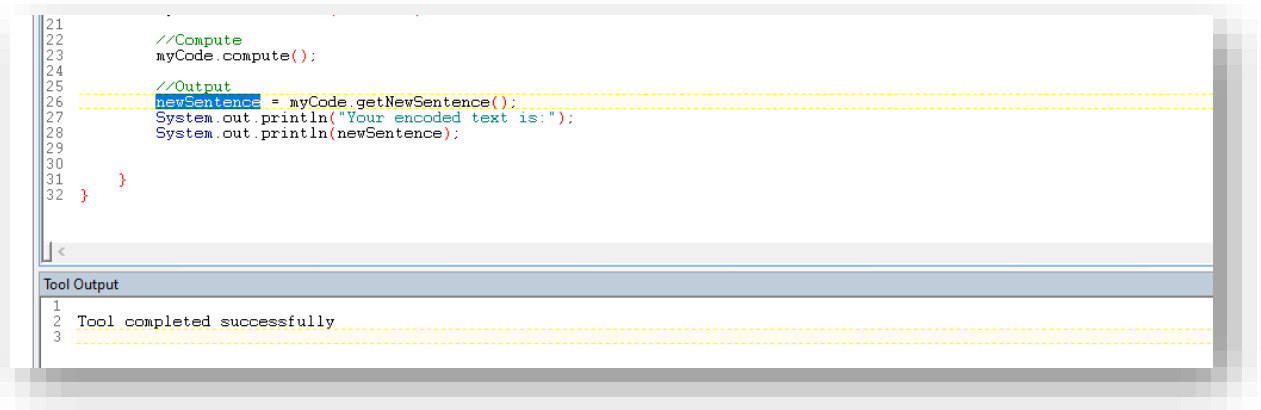



**3rd Error solution:** This was only a spelling mistake on line 18 which was solved by adding ‘t’ to “prin**t**ln”.

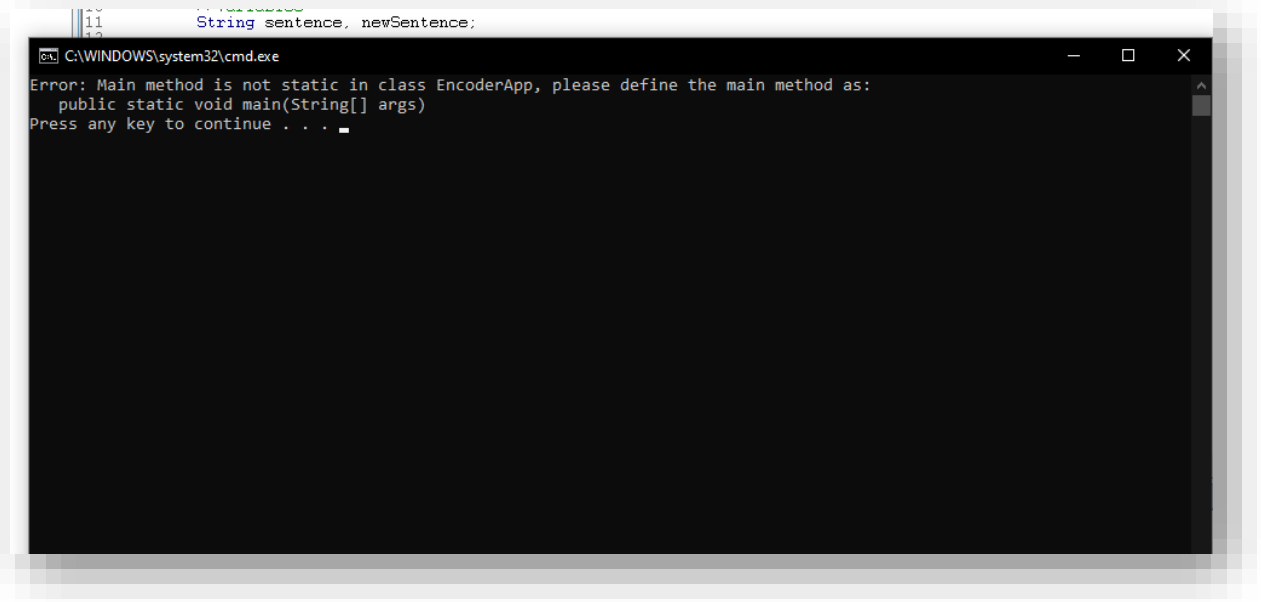



**4th Error:** variable newSentence might not have been initialized

System.out.println(newSentence);




**4th Error solution:** There was no value set for the variable “newSentence”. The error was fixed when the value getNewSentence() was set to the variable newSentence.



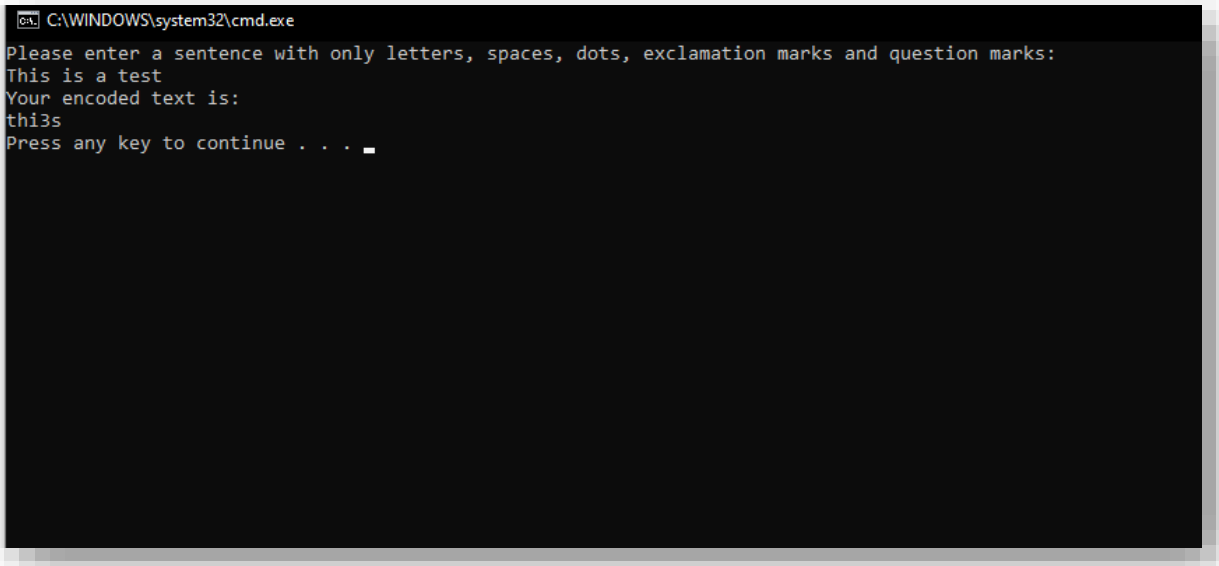

**5th Error:** Main method is not static in class

```
6
7  import java.util.*;
8  public class EncoderApp{
9      public static void main (String args[]){
10         //Variables
11         String sentence, newSentence;
12
13         //Objects
14         Scanner sc = new Scanner (System.in);
15         Encoder myCode = new Encoder();
16
17         //Input
18         System.out.println("Please enter a sentence with only letters, spaces, dots
19         sentence = sc.next();
20         myCode.setSentence(sentence);
21
22         //Compute
23         myCode.compute();
24
25         //Output
26         newSentence = myCode.getNewSentence();
27         System.out.println("Your encoded text is:");
28         System.out.println(newSentence);
29
```

Tool Output

```
1
2  Tool completed successfully
3
```

**5th Error solution:** Although the tool was completed successfully, the app was not recognizing the Main method because I forgot to write **static** on the main method. Once the "satatic" was added to the main method everything worked perfectly.

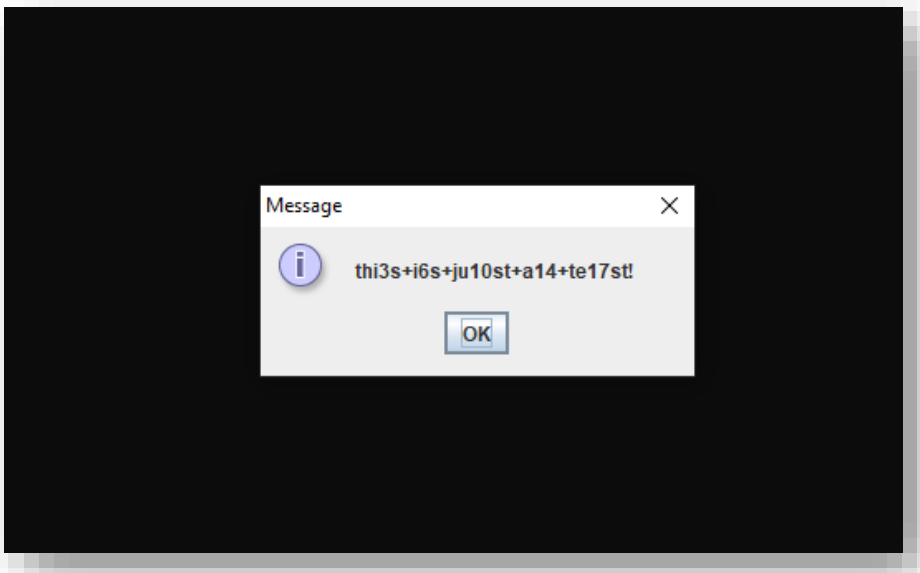



**6th Error:** This was the hardiest to identify, somehow it would not transform all my sentence in a StringBuffer and back to String. It was collecting only the first word and doing the alteration.

```
6
7  import javax.swing.JOptionPane;
8  public class EncoderApp{
9      public static void main (String args[]){
10         //Variables
11         String sentence, newSentence;
12
13         //Objects
14         Encoder myCode = new Encoder();
15
16         //Input
17         sentence = JOptionPane.showInputDialog(null, "Please enter a sentence with only letters, spaces, dots, exclamation marks and que
18         myCode.setSentence(sentence);
19
20         //Compute
21         myCode.compute();
22
23         //Output
24         newSentence = myCode.getNewSentence();
25         JOptionPane.showMessageDialog(null, newSentence);
26
27
28     }
29 }
```

Tool Output

```
1
2  Tool completed successfully
3
```




**6th Error solution:** To fix that I had to import JOptionPane instead of Scanner and do it all again, this way I ran the test once more and it worked fine.

**Code for instantiable class Encode.java Question 2I:**

```
/*
 *Encoder.java
 *@Patryck Brenner
 *2nd of Dec 2020
*/

public class Encoder{
	//Variables
	private String sentence, newSentence;
	private StringBuffer strBuff;

	//Constructor
	public Encoder(){
		sentence = "";
		newSentence = "";
		strBuff = new StringBuffer();
	}

	//Set methods
	public void setSentence(String sentence){
		this.sentence = sentence;
	}

	public void setNewSentence(String newSentence){
		this.newSentence = newSentence;
	}

	//Compute method
	public void compute(){
```

```
		sentence = sentence.toLowerCase();
		for(int i = 0; i < sentence.length(); i++){
			if (sentence.charAt(i) == 'a' || sentence.charAt(i) == 'e' ||
sentence.charAt(i) == 'i' || sentence.charAt(i) == 'o' || sentence.charAt(i) == 'u'){
				strBuff.append(sentence.charAt(i));
				strBuff.append(i + 1);
			} else if(sentence.charAt(i) == ' '){
				strBuff.append('+');
			} else if (sentence.charAt(i) == '.'){
				strBuff.append('!');
			} else if (sentence.charAt(i) == '?'){
				strBuff.append("??");
			} else if (sentence.charAt(i) == '!'){
				strBuff.append("?!");
			} else {
			strBuff.append(sentence.charAt(i));
			}
			newSentence = strBuff.toString();
		}




	}

	//Get methods
	public String getNewSentence(){
		return newSentence;
	}

}
```

**Code for main class EncodeApp.java Question 2I:**

```
/*
 *EncoderApp.java
 *@Patryck Brenner
 *2nd of Dec 2020
*/

import javax.swing.JOptionPane;
public class EncoderApp{
	public static void main (String args[]){
		//Variables
		String sentence, newSentence;

		//Objects
		Encoder myCode = new Encoder();

		//Input
		sentence = JOptionPane.showInputDialog(null, "Please enter a sentence with only letters, spaces, dots, exclamation marks and question marks:");
		myCode.setSentence(sentence);

		//Compute
		myCode.compute();

		//Output
		newSentence = myCode.getNewSentence();
		JOptionPane.showMessageDialog(null, newSentence);

	}
}
```