public class Lesson11\_01\_ReadFile {  
 public static void main(String[] args) {  
 /\*\*  
 \* создайте текстовый файл, рядом с этим файлом  
 \* заполните его строками. Конец строки - перевод строки.  
 \*/  
 try (Scanner scanner = new Scanner(  
 new FileInputStream(  
 "src/lesson/lesson\_11/lineInput.txt"))) {  
 String[] lineArray = new String[5];  
 for(int i = 0; scanner.hasNext(); i++){  
 lineArray[i] = scanner.nextLine();  
 }  
 for(int i = 0; i < lineArray.length; i++){  
 System.out.println(lineArray[i]);  
 }  
 } catch (FileNotFoundException e) {  
 e.printStackTrace();  
 }  
 }  
}

public class Lesson11\_01\_ReadFile {  
 public static void main(String[] args) {  
 /\*\*  
 \* создайте текстовый файл, рядом с этим файлом  
 \* заполните его строками. Конец строки - перевод строки.  
 \*/  
 try (Scanner scanner = new Scanner(  
 new FileInputStream(  
 "src/lesson/lesson\_11/lineInput.txt"))) {  
 String[] lineArray = new String[5];  
 for(int i = 0; scanner.hasNext(); i++){  
 lineArray[i] = scanner.nextLine();  
 }  
 for(int i = 0; i < lineArray.length; i++){  
 System.out.println(lineArray[i]);  
 }  
 } catch (FileNotFoundException e) {  
 e.printStackTrace();  
 }  
 }  
}

public class Lesson11\_01\_ReadFile {  
 public static void main(String[] args) {  
 /\*\*  
 \* создайте текстовый файл, рядом с этим файлом  
 \* заполните его строками. Конец строки - перевод строки.  
 \*/  
 try (Scanner scanner = new Scanner(  
 new FileInputStream(  
 "src/lesson/lesson\_11/lineInput.txt"))) {  
 String[] lineArray = new String[5];  
 for(int i = 0; scanner.hasNext(); i++){  
 lineArray[i] = scanner.nextLine();  
 }  
 for(int i = 0; i < lineArray.length; i++){  
 System.out.println(lineArray[i]);  
 }  
 } catch (FileNotFoundException e) {  
 e.printStackTrace();  
 }  
 }  
}

public class Lesson11\_01\_ReadFile {  
 public static void main(String[] args) {  
 /\*\*  
 \* создайте текстовый файл, рядом с этим файлом  
 \* заполните его строками. Конец строки - перевод строки.  
 \*/  
 try (Scanner scanner = new Scanner(  
 new FileInputStream(  
 "src/lesson/lesson\_11/lineInput.txt"))) {  
 String[] lineArray = new String[5];  
 for(int i = 0; scanner.hasNext(); i++){  
 lineArray[i] = scanner.nextLine();  
 }  
 for(int i = 0; i < lineArray.length; i++){  
 System.out.println(lineArray[i]);  
 }  
 } catch (FileNotFoundException e) {  
 e.printStackTrace();  
 }  
 }  
}

public class Lesson11\_02\_PositiveNumber\_ReadFile {  
 public static void main(String[] args) {  
 /\*\*  
 \* создайте текстовый файл, рядом с этим файлом  
 \* заполните его числами. Разделители между числами: проблел, перевод строки и тд.  
 \*/  
 try(Scanner scanner = new Scanner(  
 new FileInputStream(  
 "src/lesson/lesson\_11/numberInput.txt"))) {  
 scanner.useLocale(Locale.US);  
  
 System.out.print("Введите число: ");  
 int number = scanner.nextInt();  
  
 if (number > 0) {  
 System.out.println("Число " + number + " - положительное");  
 }  
 }catch (Exception e){  
 e.printStackTrace();  
 }  
 }  
}

public class Lesson11\_02\_PositiveNumber\_ReadFile {  
 public static void main(String[] args) {  
 /\*\*  
 \* создайте текстовый файл, рядом с этим файлом  
 \* заполните его числами. Разделители между числами: проблел, перевод строки и тд.  
 \*/  
 try(Scanner scanner = new Scanner(  
 new FileInputStream(  
 "src/lesson/lesson\_11/numberInput.txt"))) {  
 scanner.useLocale(Locale.US);  
  
 System.out.print("Введите число: ");  
 int number = scanner.nextInt();  
  
 if (number > 0) {  
 System.out.println("Число " + number + " - положительное");  
 }  
 }catch (Exception e){  
 e.printStackTrace();  
 }  
 }  
}

public class Lesson11\_02\_PositiveNumber\_ReadFile {  
 public static void main(String[] args) {  
 /\*\*  
 \* создайте текстовый файл, рядом с этим файлом  
 \* заполните его числами. Разделители между числами: проблел, перевод строки и тд.  
 \*/  
 try(Scanner scanner = new Scanner(  
 new FileInputStream(  
 "src/lesson/lesson\_11/numberInput.txt"))) {  
 scanner.useLocale(Locale.US);  
  
 System.out.print("Введите число: ");  
 int number = scanner.nextInt();  
  
 if (number > 0) {  
 System.out.println("Число " + number + " - положительное");  
 }  
 }catch (Exception e){  
 e.printStackTrace();  
 }  
 }  
}

public class Lesson11\_02\_PositiveNumber\_ReadFile {  
 public static void main(String[] args) {  
 /\*\*  
 \* создайте текстовый файл, рядом с этим файлом  
 \* заполните его числами. Разделители между числами: проблел, перевод строки и тд.  
 \*/  
 try(Scanner scanner = new Scanner(  
 new FileInputStream(  
 "src/lesson/lesson\_11/numberInput.txt"))) {  
 scanner.useLocale(Locale.US);  
  
 System.out.print("Введите число: ");  
 int number = scanner.nextInt();  
  
 if (number > 0) {  
 System.out.println("Число " + number + " - положительное");  
 }  
 }catch (Exception e){  
 e.printStackTrace();  
 }  
 }  
}

public class Lesson11\_03\_ReadAndWrite {  
 public static void main(String[] args) {  
  
 try(Scanner scanner = new Scanner(  
 new FileInputStream(  
 "src/lesson/lesson\_11/lineInput.txt"));  
 PrintWriter writer = new PrintWriter("src/lesson/lesson\_11/lineOutput.txt")){  
 while(scanner.hasNext()){  
 String line = scanner.nextLine();  
 writer.println(line.toUpperCase());  
 }  
 } catch (FileNotFoundException e) {  
 e.printStackTrace();  
 }  
 }  
}

public class Lesson11\_03\_ReadAndWrite {  
 public static void main(String[] args) {  
  
 try(Scanner scanner = new Scanner(  
 new FileInputStream(  
 "src/lesson/lesson\_11/lineInput.txt"));  
 PrintWriter writer = new PrintWriter("src/lesson/lesson\_11/lineOutput.txt")){  
 while(scanner.hasNext()){  
 String line = scanner.nextLine();  
 writer.println(line.toUpperCase());  
 }  
 } catch (FileNotFoundException e) {  
 e.printStackTrace();  
 }  
 }  
}

public class Lesson11\_03\_ReadAndWrite {  
 public static void main(String[] args) {  
  
 try(Scanner scanner = new Scanner(  
 new FileInputStream(  
 "src/lesson/lesson\_11/lineInput.txt"));  
 PrintWriter writer = new PrintWriter("src/lesson/lesson\_11/lineOutput.txt")){  
 while(scanner.hasNext()){  
 String line = scanner.nextLine();  
 writer.println(line.toUpperCase());  
 }  
 } catch (FileNotFoundException e) {  
 e.printStackTrace();  
 }  
 }  
}

public class Lesson11\_03\_ReadAndWrite {  
 public static void main(String[] args) {  
  
 try(Scanner scanner = new Scanner(  
 new FileInputStream(  
 "src/lesson/lesson\_11/lineInput.txt"));  
 PrintWriter writer = new PrintWriter("src/lesson/lesson\_11/lineOutput.txt")){  
 while(scanner.hasNext()){  
 String line = scanner.nextLine();  
 writer.println(line.toUpperCase());  
 }  
 } catch (FileNotFoundException e) {  
 e.printStackTrace();  
 }  
 }  
}

public class Lesson11\_04\_StringBuilder {  
 public static void main(String[] args) {  
 StringBuilder builder = new StringBuilder();  
 for (int i = 1; i <= 100; i++) {  
 builder.append(i).append(", ");  
 }  
 builder.delete(builder.length() - 2, builder.length());  
 System.*out*.println(builder.toString());  
 }  
}

public class Lesson11\_04\_StringBuilder {  
 public static void main(String[] args) {  
 StringBuilder builder = new StringBuilder();  
 for (int i = 1; i <= 100; i++) {  
 builder.append(i).append(", ");  
 }  
 builder.delete(builder.length() - 2, builder.length());  
 System.*out*.println(builder.toString());  
 }  
}

public class Lesson11\_04\_StringBuilder {  
 public static void main(String[] args) {  
 StringBuilder builder = new StringBuilder();  
 for (int i = 1; i <= 100; i++) {  
 builder.append(i).append(", ");  
 }  
 builder.delete(builder.length() - 2, builder.length());  
 System.*out*.println(builder.toString());  
 }  
}

public class Lesson11\_04\_StringBuilder {  
 public static void main(String[] args) {  
 StringBuilder builder = new StringBuilder();  
 for (int i = 1; i <= 100; i++) {  
 builder.append(i).append(", ");  
 }  
 builder.delete(builder.length() - 2, builder.length());  
 System.*out*.println(builder.toString());  
 }  
}

public class Lesson11\_05\_Url {  
 public static void main(String[] args) {  
 final String url = "http://SomeServerName/abcd/";  
 final String prefix = "://";  
 final String postfix = "/";  
 int prefixIndex = url.indexOf(prefix);  
 int endPrefixEndex = prefixIndex + prefix.length();  
 int postfixIndex = url.indexOf(postfix, endPrefixEndex);  
 if(postfixIndex == -1){  
 postfixIndex = url.length();  
 }  
 String serverName = url.substring(endPrefixEndex, postfixIndex);  
 System.*out*.printf("Url: %s. ServerName: %s", url, serverName);  
 }  
}

public class Lesson11\_05\_Url {  
 public static void main(String[] args) {  
 final String url = "http://SomeServerName/abcd/";  
 final String prefix = "://";  
 final String postfix = "/";  
 int prefixIndex = url.indexOf(prefix);  
 int endPrefixEndex = prefixIndex + prefix.length();  
 int postfixIndex = url.indexOf(postfix, endPrefixEndex);  
 if(postfixIndex == -1){  
 postfixIndex = url.length();  
 }  
 String serverName = url.substring(endPrefixEndex, postfixIndex);  
 System.*out*.printf("Url: %s. ServerName: %s", url, serverName);  
 }  
}

public class Lesson11\_05\_Url {  
 public static void main(String[] args) {  
 final String url = "http://SomeServerName/abcd/";  
 final String prefix = "://";  
 final String postfix = "/";  
 int prefixIndex = url.indexOf(prefix);  
 int endPrefixEndex = prefixIndex + prefix.length();  
 int postfixIndex = url.indexOf(postfix, endPrefixEndex);  
 if(postfixIndex == -1){  
 postfixIndex = url.length();  
 }  
 String serverName = url.substring(endPrefixEndex, postfixIndex);  
 System.*out*.printf("Url: %s. ServerName: %s", url, serverName);  
 }  
}

public class Lesson11\_05\_Url {  
 public static void main(String[] args) {  
 final String url = "http://SomeServerName/abcd/";  
 final String prefix = "://";  
 final String postfix = "/";  
 int prefixIndex = url.indexOf(prefix);  
 int endPrefixEndex = prefixIndex + prefix.length();  
 int postfixIndex = url.indexOf(postfix, endPrefixEndex);  
 if(postfixIndex == -1){  
 postfixIndex = url.length();  
 }  
 String serverName = url.substring(endPrefixEndex, postfixIndex);  
 System.*out*.printf("Url: %s. ServerName: %s", url, serverName);  
 }  
}

public class Lesson11\_06\_CalcSubstringCount {  
 public static void main(String[] args) {  
 try (Scanner scanner = new Scanner(  
 new FileInputStream(  
 "src/lesson/lesson\_11/lineInput.txt"))) {  
 StringBuilder builder = new StringBuilder();  
 while (scanner.hasNext()) {  
 builder.append(scanner.nextLine()).append(System.*lineSeparator*());  
 }  
 String text = builder.toString();  
 final String substring = "123";  
 int countSubstring = 0;  
 int indexOfSubstring = text.indexOf(substring);  
 while (indexOfSubstring != -1) {  
 countSubstring++;  
 indexOfSubstring = text.indexOf(substring, indexOfSubstring + substring.length());  
 }  
 System.*out*.printf("substring: %s, count: %d, text: %s.", substring, countSubstring, text);  
 } catch (FileNotFoundException e) {  
 e.printStackTrace();  
 }}}

public class Lesson11\_06\_CalcSubstringCount {  
 public static void main(String[] args) {  
 try (Scanner scanner = new Scanner(  
 new FileInputStream(  
 "src/lesson/lesson\_11/lineInput.txt"))) {  
 StringBuilder builder = new StringBuilder();  
 while (scanner.hasNext()) {  
 builder.append(scanner.nextLine()).append(System.*lineSeparator*());  
 }  
 String text = builder.toString();  
 final String substring = "123";  
 int countSubstring = 0;  
 int indexOfSubstring = text.indexOf(substring);  
 while (indexOfSubstring != -1) {  
 countSubstring++;  
 indexOfSubstring = text.indexOf(substring, indexOfSubstring + substring.length());  
 }  
 System.*out*.printf("substring: %s, count: %d, text: %s.", substring, countSubstring, text);  
 } catch (FileNotFoundException e) {  
 e.printStackTrace();  
 }}}

public class Lesson11\_06\_CalcSubstringCount {  
 public static void main(String[] args) {  
 try (Scanner scanner = new Scanner(  
 new FileInputStream(  
 "src/lesson/lesson\_11/lineInput.txt"))) {  
 StringBuilder builder = new StringBuilder();  
 while (scanner.hasNext()) {  
 builder.append(scanner.nextLine()).append(System.*lineSeparator*());  
 }  
 String text = builder.toString();  
 final String substring = "123";  
 int countSubstring = 0;  
 int indexOfSubstring = text.indexOf(substring);  
 while (indexOfSubstring != -1) {  
 countSubstring++;  
 indexOfSubstring = text.indexOf(substring, indexOfSubstring + substring.length());  
 }  
 System.*out*.printf("substring: %s, count: %d, text: %s.", substring, countSubstring, text);  
 } catch (FileNotFoundException e) {  
 e.printStackTrace();  
 }}}

public class Lesson11\_06\_CalcSubstringCount {  
 public static void main(String[] args) {  
 try (Scanner scanner = new Scanner(  
 new FileInputStream(  
 "src/lesson/lesson\_11/lineInput.txt"))) {  
 StringBuilder builder = new StringBuilder();  
 while (scanner.hasNext()) {  
 builder.append(scanner.nextLine()).append(System.*lineSeparator*());  
 }  
 String text = builder.toString();  
 final String substring = "123";  
 int countSubstring = 0;  
 int indexOfSubstring = text.indexOf(substring);  
 while (indexOfSubstring != -1) {  
 countSubstring++;  
 indexOfSubstring = text.indexOf(substring, indexOfSubstring + substring.length());  
 }  
 System.*out*.printf("substring: %s, count: %d, text: %s.", substring, countSubstring, text);  
 } catch (FileNotFoundException e) {  
 e.printStackTrace();  
 }}}

public class Lesson11\_07\_Sum {  
 public static void main(String[] args) {  
 final String numbersLine = "1, 2, 3, 5";  
 String [] numbersAsStringArray = numbersLine.split(", ");  
 int [] numbersArray = new int [numbersAsStringArray.length];  
 for(int i = 0; i < numbersArray.length; i++){  
 numbersArray[i] = Integer.*parseInt*(numbersAsStringArray[i]);  
 }  
 int sum = 0;  
 for(int i = 0; i < numbersArray.length; i++){  
 sum += numbersArray[i];  
 }  
 System.*out*.printf("numbersLine: %s, sum: %d", numbersLine, sum);  
 }  
}

public class Lesson11\_07\_Sum {  
 public static void main(String[] args) {  
 final String numbersLine = "1, 2, 3, 5";  
 String [] numbersAsStringArray = numbersLine.split(", ");  
 int [] numbersArray = new int [numbersAsStringArray.length];  
 for(int i = 0; i < numbersArray.length; i++){  
 numbersArray[i] = Integer.*parseInt*(numbersAsStringArray[i]);  
 }  
 int sum = 0;  
 for(int i = 0; i < numbersArray.length; i++){  
 sum += numbersArray[i];  
 }  
 System.*out*.printf("numbersLine: %s, sum: %d", numbersLine, sum);  
 }  
}

public class Lesson11\_07\_Sum {  
 public static void main(String[] args) {  
 final String numbersLine = "1, 2, 3, 5";  
 String [] numbersAsStringArray = numbersLine.split(", ");  
 int [] numbersArray = new int [numbersAsStringArray.length];  
 for(int i = 0; i < numbersArray.length; i++){  
 numbersArray[i] = Integer.*parseInt*(numbersAsStringArray[i]);  
 }  
 int sum = 0;  
 for(int i = 0; i < numbersArray.length; i++){  
 sum += numbersArray[i];  
 }  
 System.*out*.printf("numbersLine: %s, sum: %d", numbersLine, sum);  
 }  
}

public class Lesson11\_07\_Sum {  
 public static void main(String[] args) {  
 final String numbersLine = "1, 2, 3, 5";  
 String [] numbersAsStringArray = numbersLine.split(", ");  
 int [] numbersArray = new int [numbersAsStringArray.length];  
 for(int i = 0; i < numbersArray.length; i++){  
 numbersArray[i] = Integer.*parseInt*(numbersAsStringArray[i]);  
 }  
 int sum = 0;  
 for(int i = 0; i < numbersArray.length; i++){  
 sum += numbersArray[i];  
 }  
 System.*out*.printf("numbersLine: %s, sum: %d", numbersLine, sum);  
 }  
}