Comp 401
Basics Of Scanning and Java

Instructor: Prasun Dewan (FB 150, dewan@unc.edu)

Programming Overview through Example

- Problem
- Algorithm
- Representation
- Code

Scanning Problem

- Scanning image for text.
- Scanning frequencies for radio stations.
- Finding words in a sentence
- Finding identifiers, operators, in a program

Scanning Problem

(First) Argument or option to the program

Program name

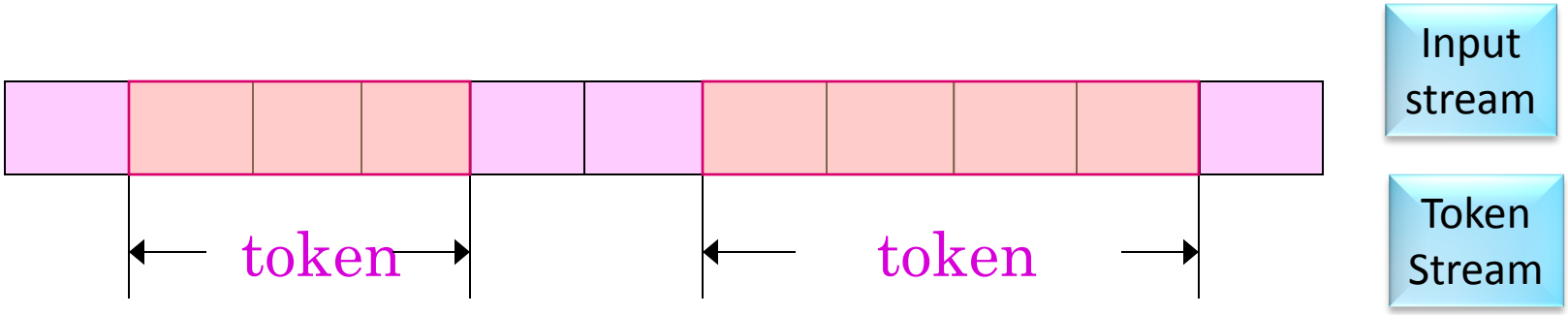
```
D:\dewan_backup\Java\JavaTeaching\bin>java lectures.scanning.AnUpperCasePrinter
"John F. Kennedy"
Upper Case Letters:
JFK
```

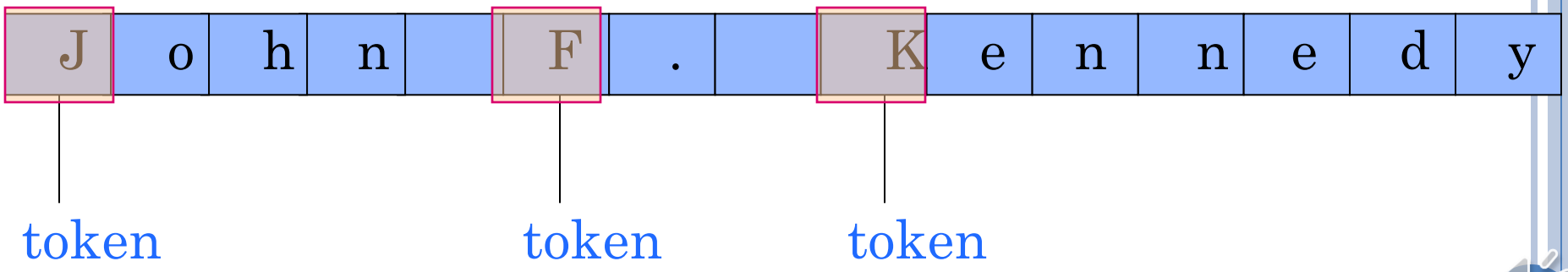
output

Algorithm

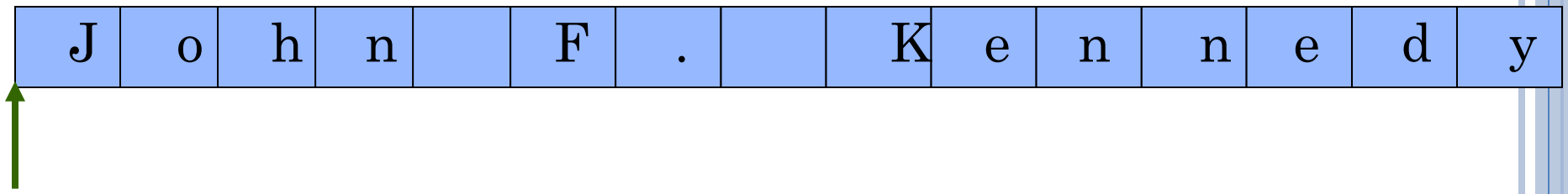
- Description of solution to a problem.
- Can be in any “language”
  - graphical
  - natural or programming language
  - natural + programming language (pseudo code)
- Can describe solution to various levels of detail

Problem


Input stream
Token Stream
token
token

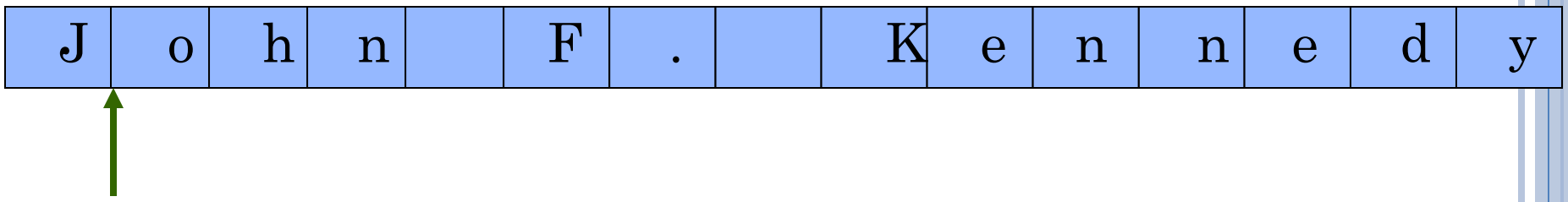

J o h n F . K e n n e d y
token
token
token

Algorithm



Output: J

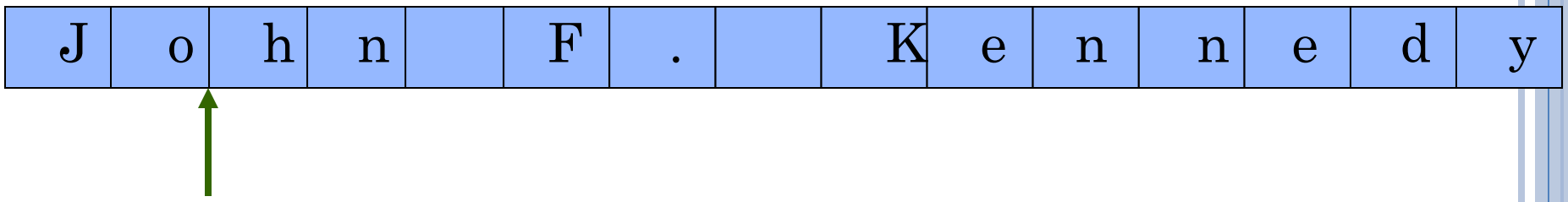
Algorithm



marker 1

Output: J

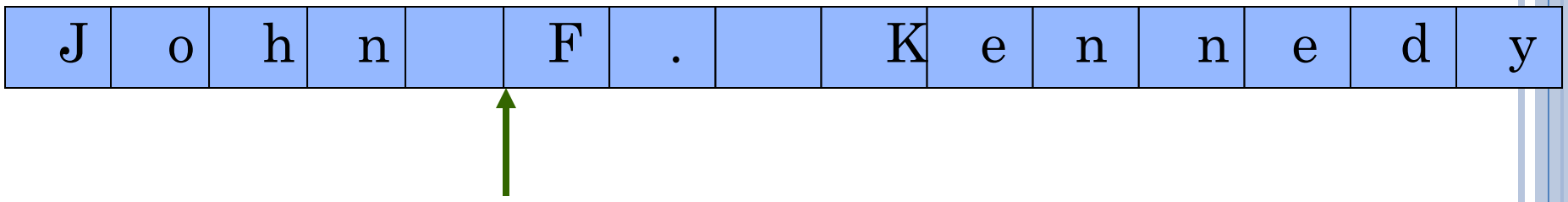
Algorithm



marker 2

Output: J

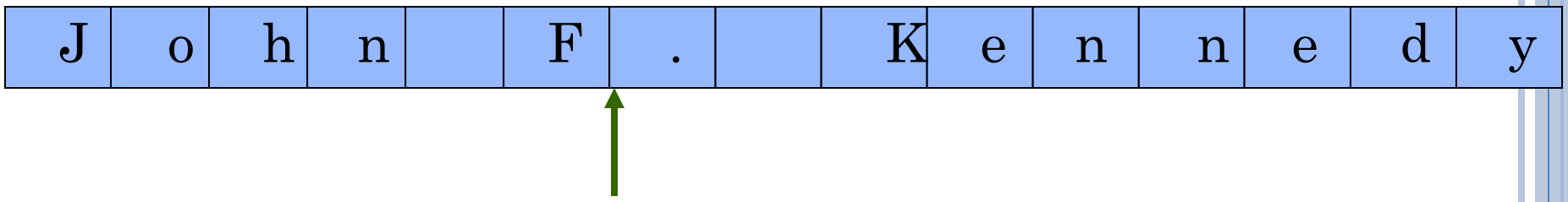
Algorithm



marker 5

Output: JF

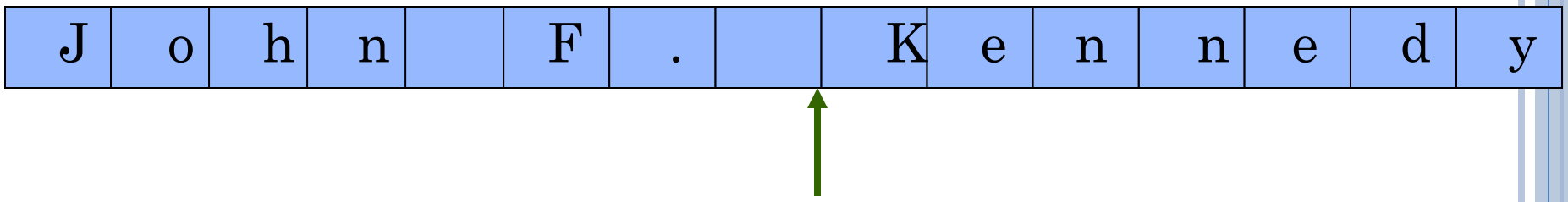
Algorithm



marker 6

Output: JF

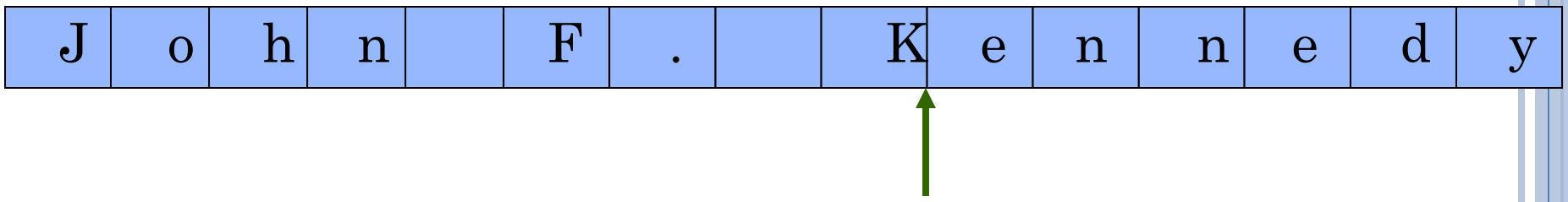
Algorithm



marker 8

Output: JFK

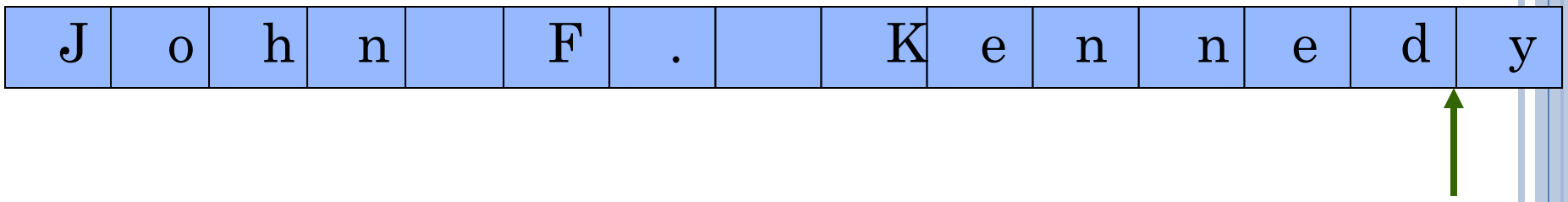
Algorithm



marker 9

Output: JFK

Algorithm



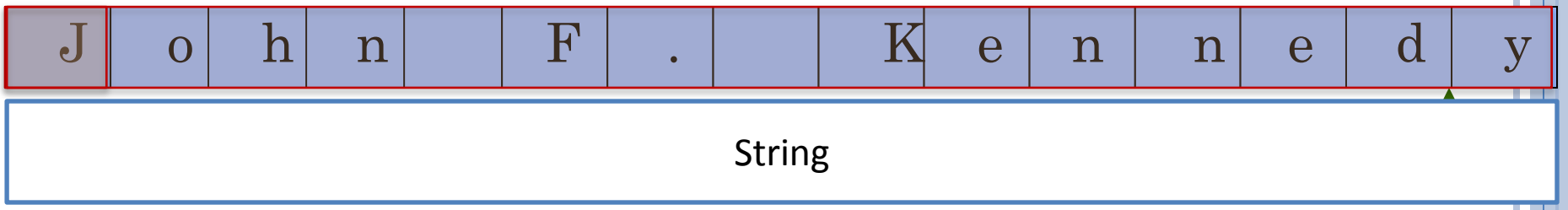
marker 14

Output: JFK

Demo

Scanning Algorithm Demo

Choosing Representation: Data Types



marker 14 int

Output: JFK char

Java Program Structure

```
D:\dewan_backup\Java\JavaTeaching\bin>java lectures.scanning.AnUpperCasePrinter
"John F. Kennedy"
Upper Case Letters:
JFK
```

```
package lectures.scanning;
public class AnUpperCasePrinter {
  public static void main(String[] args) {
    if (args.length != 1) {
      System.out.println("Illegal number of argum
+ ". Terminating program.");
      System.exit(-1);
    }
    String scannedString = args[0];
    System.out.println("Upper Case Letters:");
    int index = 0;
    while (index < scannedString.length()) {
      char nextLetter = scannedString.charAt(index);
      if (Character.isUpperCase(nextLetter))
        System.out.print(nextLetter);
      index++;
    }
    System.out.println();
  }
}
```

Must have this procedure header in executable program

Predefined internal library operations

Print on new vs. previous line

Java Program Structure

```
D:\dewan_backup\Java\JavaTeaching\bin>java lectures.scanning.AnUpperCasePrinter
"John F. Kennedy"
Upper Case Letters:
JFK
```

```
package lectures.scanning;
public class AnUpperCasePrinter {
  public static void main(String[] args) {
    if (args.length != 1) {
      System.out.println("Illegal number of arguments:" + args.length
+ ". Terminating program.");
      System.exit(-1);
    }
    String scannedString = args[0];
    System.out.println("Upper Case Letters:");
    int index = 0;
    while (index < scannedString.length()) {
      char nextLetter = scannedString.charAt(index);
      if (nextLetter >= 'A' && nextLetter <= 'Z')
        System.out.print(nextLetter);
      index++;
    }
    System.out.println();
  }
}
```

Characters are ordered

Reading Input

Library package

Library class

External library, not part of language

```
package ...res.scan
import java.util.Scanner;
public class AConsoleReadingUpperCasePrinter {
  public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    String scannedString = scanner.nextLine();
    System.out.println("Upper Case Letters:");
    int index = 0;
    while (index < scannedString.length()) {
      char nextLetter = scannedString.charAt(index);
      if (nextLetter >= 'A' && nextLetter <= 'Z')
        System.out.print(nextLetter);
      index++;
    }
   System.out.println();
  }
}
```

Omitted Import

```
package lectures.scanning;

public class AConsoleReadingUpperCasePri          {
  public static void main(String[]          ) {
    Scanner scanner = new Scanner(System.in);
    String scannedString = scanner.nextLine();
    System.out.println("Upper Case Letters:");
    int index = 0;
    while (index < scannedString.length()) {
      char nextLetter = scannedString.charAt(index);
      if (nextLetter >= 'A' && nextLetter <= 'Z')
        System.out.print(
      index++;
    }
   System.out.println();
  }
}
```

Error

In Eclipse press CTRL-SHIFT-O to automatically import all used classes not in the same package

If class is in more than one package, Eclipse gives a choice

In future code, package names and imports are omitted

Reading Input

Library package

Library class

External library, not part of language

```
package ...res.scan
import java.util.Scanner;
public class AConsoleReadingUpperCasePrinter {
  public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    String scannedString = scanner.nextLine();
    System.out.println("Upper Case Letters:");
    int index = 0;
    while (index < scannedString.length()) {
      char nextLetter = scannedString.charAt(index);
      if (nextLetter >= 'A' && nextLetter <= 'Z')
        System.out.print(nextLetter);
      index++;
    }
   System.out.println();
  }
}
```

Omitted Import

```
package lectures.scanning;

public class AConsoleReadingUpperCasePr
  public static void main(String[
    Scanner scanner = new Scanner(System.in);
    String scannedString = scanner.nextLine();
    System.out.println("Upper Case Letters:");
    int index = 0;
    while (index < scannedString.length()) {
      char nextLetter = scannedString.charAt(index);
      if (nextLetter >= 'A' && nextLetter <= 'Z')
        System.out.print(
      index++;
    }
   System.out.println();
  }
}
```

Error

In Eclipse press CTRL-SHIFT-O to automatically import all used classes not in the same package

If class is in more than one package, Eclipse gives a choice

In future code, package names and imports are omitted

More on Packages and Imports

110 and 401 (look on you own)	Packages	PowerPoint	PDF	Objects Chapter			

Decomposition?

```
package lectures.scanning;
import java.util.Scanner;
public class AConsoleReadingUpperCasePrinter {
  public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    String scannedString = scanner.nextLine();
    System.out.println("Upper Case Letters:");
    int index = 0;
    while (index < scannedString.length()) {
      char nextLetter = scannedString.charAt(index);
      if (nextLetter >= 'A' && nextLetter <= 'Z')
        System.out.print(nextLetter);
      index++;
    }
   System.out.println();
  }
}
```

Monolithic!

Modular Reading Input

```
package lectures.scanning;
import java.util.Scanner;
public class AModularConsoleReadingUpperCasePrinter {
  public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    String scannedString = scanner.nextLine();
    scanAndPrint(scannedString);
  }
  public static void scanAndPrint(String scannedString) {
    System.out.println("Upper Case Letters:");
    int index = 0;
    while (index < scannedString.length()) {
      char nextLetter = scannedString.charAt(index);
      if (nextLetter >= 'A' && nextLetter <= 'Z')
        System.out.print(nextLetter);
      index++;
    }
    System.out.println();
  }
}
```

Monolithic Arg Scanner

```
package lectures.scanning;
public class AnUpperCasePrinter {
  public static void main(String[] args) {
    if (args.length != 1) {
      System.out.println("Illegal number of arguments:" + args.length
+ ". Terminating program.");
      System.exit(-1);
    }
    String scannedString = args[0];
    System.out.println("Upper Case Letters:");
    int index = 0;
    while (index < scannedString.length()) {
      char nextLetter = scannedString.charAt(index);
      if (nextLetter >= 'A' && nextLetter <= 'Z')
        System.out.print(nextLetter);
      index++;
    }
    System.out.println();
  }
}
```

Code Reuse in Modular Arg Scanner

```
package lectures.scanning;
public class AModularUpperCasePrinter {
  public static void main(String[] args) {
    if (args.length != 1) {
      System.out.println("Illegal number of arguments:" +
args.length
      + ". Terminating program.");
      System.exit(-1);
    }
    AModularConsoleReadingUpperCasePrinter.scanAndPrint(args[0]);
  }
}
```

SUBSTRING

```
static void subString() {
 System.out.println("hello world".substring(4, 7));
 System.out.println("hello world".substring(4, 4));
 System.out.println("hello world".substring(7, 4));
}
```

s.substring(beginIndex, endIndex)

→ s.charAt(beginIndex) .. s.charAt(endIndex-1)

“hello world”.substring(4,7)	“o w”
“hello world”.substring(4,4)	“”
“hello world”.substring(7,4)	StringIndexBounds Exception

Console Class, Exceptions, parseInt and Comments

```
public class Console {
  static Scanner scanner = new Scanner(System.in);
  public static int readInt() {
    try { // program block that can cause an error
      return Integer.parseInt(scanner.nextLine());
    } catch (Exception e) { // program block that handles the error
      e.printStackTrace();
      System.out.println(e);
      return 0;
    }
  }
  public static String readString() {
  try {
    return scanner.nextLine();
   } catch (Exception e) {
     System.out.println(e);
     return "";
   }
}
```

For Loops, Arrays and Comments

```
// fill array with numbers input by the user
System.out.println("Number of Strings:");
int numElements = Console.readInt(); // reads the next line as integer
System.out.println("Please enter " + numElements + " strings");
String[] strings = new String[numElements]; // dynamic array
for (int elementNum = 0; elementNum < numElements; elementNum++)
  strings[elementNum] = Console.readString();
```

Single line comment

```
/*
 * This loop prints  the array input in the previous loop
 */
for (int elementNum = 0; elementNum < strings.length; elementNum++)
  System.out.println(strings[elementNum]);
```

multi line comment

```
// print 0th array element
String s = strings[0]; // unsafe
for (int i = 0; i < s.length(); i++)
  System.out.println(s.charAt(i));
```

Difference in syntax: arrays built into language, strings are library

Comments vs. Program Decomposition

```
// fill array with numbers input by the user
System.out.println("Number of Strings:");
int numElements = Console.readInt(); // reads the next line as integer
System.out.println("Please enter " + numElements + " strings");
String[] strings = new String[numElements]; // dynamic array
for (int elementNum = 0; elementNum < numElements; elementNum++)
  strings[elementNum] = Console.readString();
```

```
/*
 * This loop prints  the array input in the previous loop
 */
for (int elementNum = 0; elementNum < strings.length; elementNum++)
  System.out.println(strings[elementNum]);
```

```
// print 0th array element
String s = strings[0]; // unsafe
for (int i = 0; i < s.length(); i++)
  System.out.println(s.charAt(i));
```

Clean self explaining code first choice, comments on messy code second choice

Program Decomposition

Comment→long identifier name

```
public static void modularReadAndPrintStrings() () {
  String[] strings = readStrings(readNumStrings());
  printStrings(strings);
  printString(strings[0]);// unsafe
}
```

Eclipse CTRL-SPACE will complete name for you

```
public static int readNumStrings() {
  System.out.println("Number of Strings:");
  return Console.readInt(); // reads the next line as integer
}
```

```
public static String[] readStrings(int numElements) {
  System.out.println("Please enter " + numElements + " strings");
  String[] strings = new String[numElements]; // dynamic array
  for (int elementNum = 0; elementNum < numElements; elementNum++)
    strings[elementNum] = Console.readString();
  return strings;
}
```

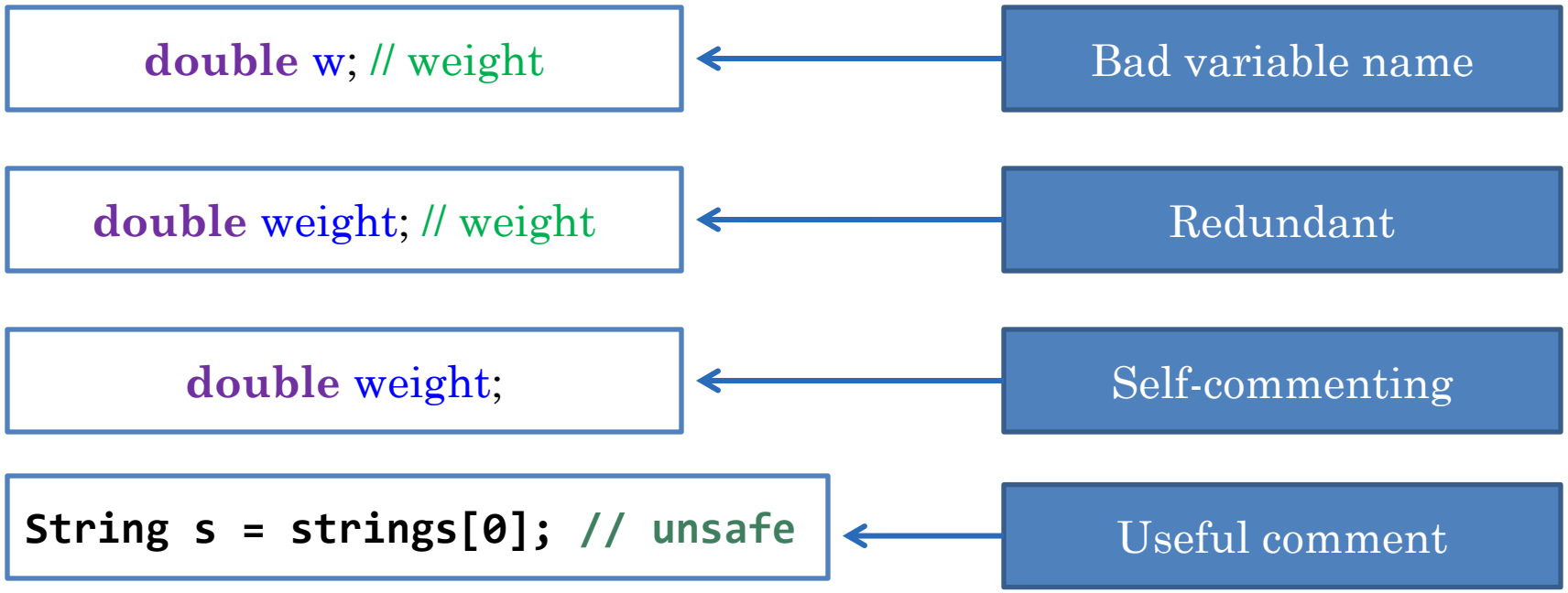
Useful information assuming the reader does not know it

Method Decomposition (Contd.)

```
public static void printStrings(String[] strings) {
  for (int elementNum = 0; elementNum < strings.length;
           elementNum++)
  System.out.println(strings[elementNum]);
}
```

```
public static void printString(String s) {
  for (int i = 0; i < s.length(); i++)
    System.out.println(s.charAt(i));
}
```

What to Write in a Comment?



Java Case Conventions

Start Class Names With Upper Case Letters

- ~~anUppercasePrinter~~
- AnUppercasePrinter

Start Variable and Method Names With Lower Case Letters

- weight
- ~~Weight~~
- ~~Square~~
- square

Start Variable and Method Names With Lower Case Letters Each New Word in the Name Starts with a Capital Letter

- ~~convertinches~~
- convertToInches
- AnUppercasePrinter
- ~~Anuppercaseprinter~~

Description by Example vs. More Formal Description

Based on your knowledge of previous programming, can write a basic program and start on the first assignment

Need a more formal, definition-based introduction to understand what is really going on