Input

AP Computer Science

Credit: Slides are modified with permission from Barry Wittman at Elizabethtown College

Input

Using Scanner

- There are three parts to using **`Scanner`** for input
  1. Include the appropriate import statement to be able to use a **`Scanner`** object
  2. Create a specific **`Scanner`** object with a name you choose
  3. Use the object you create to read in data

Importing Scanner

- People have written all kinds of useful Java code
- By **importing** this code, we can use it to help solve our problems
- To import code, you type **`import`** and then the name of the package or class
- To import **`Scanner`**, type the following at the top of your program (before the **`class`**!)

```
import java.util.Scanner;
```

Creating a Scanner object

- Once you have imported the **Scanner** class, you have to create a **Scanner** object
- To do so, declare a reference of type **Scanner**, and use the new keyword to create a new **Scanner** with **System.in** as a parameter like so:

```
Scanner scan = new Scanner(System.in);
```

- You can use any name you want, here it is **scan**
- Does not make any sense? For now, that is okay.

Using a Scanner object

- Now you have a **`Scanner`** object you can use to read in data
- It has a method to read in the next piece of data user types
- You have to know if the data is an **`int`**, a **`double`**, or a **`String`**

nextInt()

- To use Scanner to bring in an **`int`** value you need to use the **`nextInt()`** method
- The variable you assign to should be an **`int`**

```
int years;
years = scan.nextInt();
```

nextDouble()

- To use Scanner to bring in a **`double`** value you need to use the **`nextDouble()`** method
- The variable you assign to should be a **`double`**

```
double distance;
distance = scan.nextDouble();
```

next()

- To use Scanner to bring in a **`String`** value you need to use the **`next()`** method
- The variable you assign to should be a **`String`**

```
String name;
name = scan.next();
```

nextLine()

- To use Scanner to bring in a multi-word **String** value you need to use the **nextLine()** method
- The variable you assign to should be a **String**

```
String sentence;
sentence = scan.nextLine();
```

nextLine() Problems

- **`nextLine()`** works great until you use another Scanner method preceding **`nextLine()`**

```
int age;
System.out.println("Enter your age: ");
age = scan.nextInt();
String name;
System.out.print("What is your name?");
name = scan.nextLine();
System.out.println("Your age is " + age
+ " Your name is: " + name);
```

- In this instance, the name will never be inputted into your program

nextLine() Solution

- Here is the solution:

```
int age;
System.out.println("Enter your age: ");
age = scan.nextInt();
String name;
scan.nextLine(); //dummy nextLine()
System.out.print("What is your name?");
name = scan.nextLine();
System.out.println("Your age is " + age
+ " Your name is: " + name);
```

- Now the program will work properly. Remember the `nextLine()` issue!

Putting it all together

```java
import java.util.Scanner;

public class Age
{
  public static void main(String args[])
  {
    Scanner scan = new Scanner(System.in);
    System.out.println("What is your age?");
    int years;
    years = scan.nextInt();
    years = years * 2;
    System.out.print("Your age doubled is ");
    System.out.println(years);
  }
}
```