Getting Input from Keyboard

Sang Shin
Michèle Garoche
www.javapassion.com
"Learn with Passion!"

Objectives

At the end of the lesson, the student should be able to:

- Create an interactive Java program that gets input from the keyboard
- Use the BufferedReader class to get input from the keyboard using a console
- Use the JOptionPane class to get input from the keyboard using a graphical user interface

Getting Input from the Keyboard

- Two methods of getting input:
  - BufferedReader class
  - JOptionPane class
    - graphical user interface

BufferedReader Class

Using BufferedReader Class

- BufferedReader class
  - Found in the java.io package
  - Used to get input

Steps to get Input

1. Add this at the top of your code:

```
import java.io.*;
```

2. Add this statement:

```
BufferedReader dataIn = new BufferedReader( new
  InputStreamReader(System.in) );
```

Steps to get Input

3. Declare a temporary String variable to get the input, and invoke the *readLine()* method to get input from the keyboard. You have to type it inside a try-catch block.

```
try{
   String temp = dataIn.readLine();
}catch( IOException e ){
    System.out.println("Error in getting input");
}
```

Sample Program

```
1    import java.io.BufferedReader;
2    import java.io.InputStreamReader;
3    import java.io.IOException;
4
5    public class GetInputFromKeyboard {
6
7       public static void main( String[] args ){
8          BufferedReader dataIn = new BufferedReader(
9                                       new InputStreamReader( System.in) );
10
11         String name = "";
12         System.out.print("Please Enter Your Name:");
13         try{
14            name = dataIn.readLine();
15         }catch( IOException e ){
16            System.out.println("Error!");
17         }
18         System.out.println("Hello " + name +"!");
19      }
20   }
```

Studying the Sample Program

- The lines,

```
import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.io.IOException;
```

  indicate that we want to use the classes BufferedReader, InputStreamReader and IOException which are inside the java.io package.

- These statements can also be written as,

```
import java.io.*;
```

Studying the Sample Program

- The Java Application Programming Interface (API) contains hundreds of predefined classes that you can use in your programs. These classes are organized into what we call packages.

- Packages contain classes that have related purpose.

Studying the Sample Program

- The statement,

```
public class GetInputFromKeyboard {
```

means we declare a class named GetInputFromKeyboard

```
public static void main( String[] args ){
```

- The next statement declares the main method.

Studying the Sample Program

- The statement,

```
BufferedReader dataIn = new BufferedReader(new InputStreamReader
                                                ( System.in) );
```

declares a variable named `dataIn`, with the class type `BufferedReader`.

  - Don't worry about what the syntax means for now. We will cover more about classes and declaring classes later in the course.

Studying the Sample Program

- The statement,

```
String name = "";
```

declares a String type variable name.

- The next statement,

```
System.out.print("Please Enter Your Name:");
```

outputs a String “Please Enter Your Name:” on the screen

Studying the Sample Program

- The given block defines a try-catch block.

```
try{
   name = dataIn.readLine();
}catch( IOException e ){
   System.out.println("Error!");
}
```

This assures that the possible exceptions that could occur in the statement

```
name = dataIn.readLine();
```

will be caught.

  - We will cover more about exception handling in the latter part of this course.

Studying the Sample Program

- Now going back to the statement,

```
name = dataIn.readLine();
```

  the method call, dataIn.readLine(), gets input from the user and will return it as a String value.

- This value will then be saved to our name variable, which we will use in our final statement to greet the user,

```
System.out.println("Hello " + name + "!");
```

JOptionPane Class

Using JOptionPane Class

- Another way to get input from the user is by using the JOptionPane class which is found in the javax.swing package.

- JOptionPane makes it easy to pop up a standard dialog box that prompts users for a value or informs them of something.

Sample Program

```
1 import javax.swing.JOptionPane;
2
3 public class GetInputFromKeyboard {
4
5   public static void main( String[] args ){
6     String name = "";
7     name=JOptionPane.showInputDialog(“Please enter your name");
8     String msg = "Hello " + name + "!";
9     JOptionPane.showMessageDialog(null, msg);
10  }
11}
```

Sample Program Output

Sample Program

- The statement,

```
import javax.swing.JOptionPane;
```

  indicates that we want to import the JOptionPane class from the javax.swing package.

- This can also written as,

```
import javax.swing.*;
```

Studying the Sample Program

- The statement,

```
name=JOptionPane.showInputDialog(“Please enter your name");
```

  creates a JOptionPane input dialog, which will display a dialog with a message, a textfield and an OK and Cancel buttons as shown in the figure.

- This returns a String which we will save in the name variable.

Studying the Sample Program

- The statement,

```
String msg = "Hello " + name + "!";
```

  creates the welcome message, which we will store in the `msg` variable.

Sample Program

- The statement,

```
JOptionPane.showMessageDialog(null, msg);
```

displays a dialog which contains a message and an OK button.

Summary

- Discussed two ways of getting input from the user by using the classes:
  - BufferedReader
  - JOptionPane
- Brief overview of packages
  - Groups related classes in Java
  - Classes inside packages can be used by importing the package

Thank you!

Check JavaPassion.com Codecamps!
http://www.javapassion.com/codecamps
"Learn with Passion!"