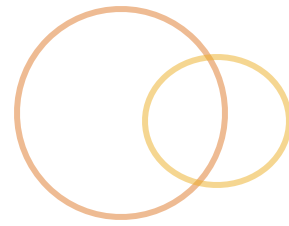


Modifications to java.util

Not so many changes



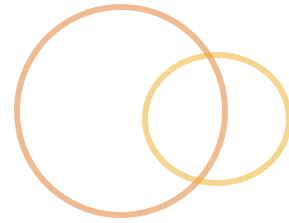
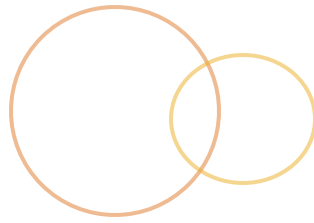
Presentation Topics



In this presentation we will cover:

- 🕒 What's changed
- 🕒 Working with formatted data

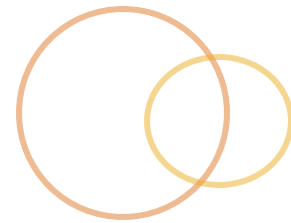
Objectives



When you are done, you should be able to:

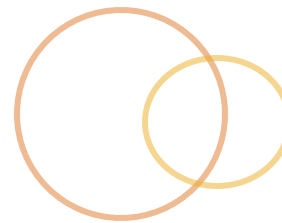
- 🕒 Identify what changed in the `java.util` package
- 🕒 Describe how to generate formatted output

Overview of java.util



- Package containing:
 - Collections Framework
 - Event models
 - Date and Time facilities
 - Internationalization
 - Miscellaneous classes

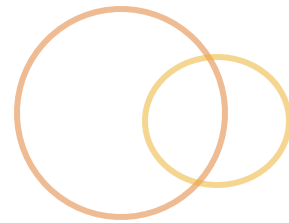
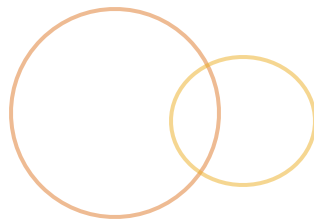
What's changed?



⦿ Not much introduced outside of Collections Framework

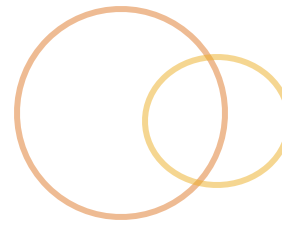
- ⦿ New methods added to existing classes
- ⦿ Few new classes
 - ⦿ Formatter
 - ⦿ Scanner
 - ⦿ UUID

Formatter



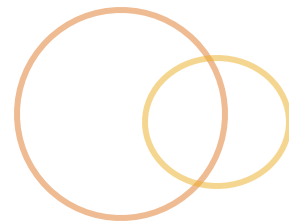
- ⦿ Provides C-like formatted output
- ⦿ Can use for layout justification and alignment
- ⦿ Supports locale centric formatted output:
 - ⦿ Numbers
 - ⦿ Strings
 - ⦿ Date / time
- ⦿ Multiple ways to format
 - ⦿ `Formatter` instance
 - ⦿ `String.format`
 - ⦿ `System.printf`

Formatter Example



```
1 package examples.formatter;
2
3 /**
4  * Basic Formatter example.
5  *
6  * @author developintelligence llc
7  * @version 1.0
8  */
9 public class BasicFormatOutputExample {
10
11     public static void main(String[] args) {
12         System.out.printf("The %s is %d years old\n", "dog", 11);
13         System.out.printf("The %s is %d years old\n", "child", 7);
14         System.out.printf("Your bank account has $%,.2f for its balance", 345.8956);
15     }
16 }
17
```

Formatted Input

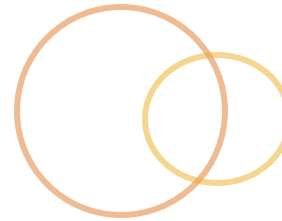
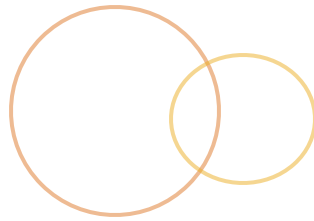
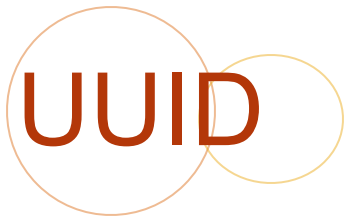


- ⦿ Provides simple text-input Scanner
- ⦿ Can parse primitives and strings
- ⦿ Supports regular expressions

Scanner Example

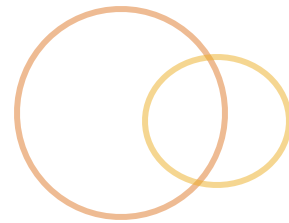
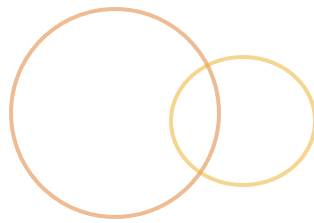


```
1  package examples.scanner;
2
3  import java.io.File;
4  import java.io.FileNotFoundException;
5  import java.util.Scanner;
6
7  /**...*/
19 public class CsvScanner {
20
21     public static void main(String[] args) {
22         if (args.length == 0) {
23             System.out.println("CsvScanner needs a csv file name as a argument.");
24             System.exit(0);
25         }
26
27         try {
28             Scanner fileScanner = new Scanner(new File(args[0]));
29             fileScanner.useDelimiter(",\\s*");
30             while (fileScanner.hasNext()) {
31                 String field = fileScanner.next();
32                 System.out.print(field + " ");
33             }
34             System.out.flush();
35             System.out.println("\nCsvScanner processing next row..");
36         } catch (FileNotFoundException e) {
37             e.printStackTrace();
38         }
39     }
40 }
41
```



- ⦿ Class for creating immutable Universally Unique Identifiers
- ⦿ Represented as 128-bit number

Summary



- ◎ `Formatter` used for formatted String generation
- ◎ `Scanner` used to parse text into typed representations