

Other Useful Java Classes

- ① Scanner
- ② Random
- ③ Math

The Scanner class

- allows program to be interactive
- takes input from user
 - e.g. program asks user question using `System.out.println()`
user types in answer
 - e.g. program prints out a list of menu options
user types in selection

- create Scanner object

e.g.

```
Scanner sysin;  
sysin = new Scanner(System.in);  
                        object for keyboard input
```

⋮

```
String userInput;  
userInput = sysin.nextLine();
```

- `nextLine()` grabs entire line as String
- `nextInt()` returns user input as int
- `nextDouble()` " " double

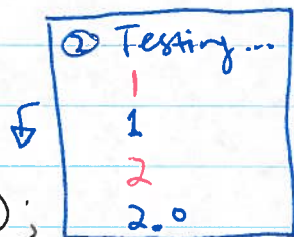
- e.g. `System.out.println("Testing...");`
`System.out.println(sysin.nextLine());`

Testing...

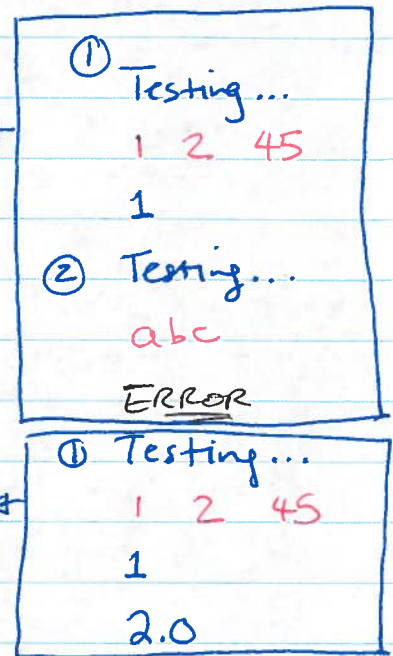
1 2 3 456

"1 2 3 456"

- e.g. `S.O.P("Testing...");`
`System.out.println( sysin.nextInt());`



- e.g. `S.O.P("Testing...");`
`System.out.println( sysin.nextInt());`
`System.out.println( sysin.nextDouble());`



- to use Scanner class, need at beginning of class file:

`import java.util.Scanner;`
↑
name of package
↑
name of class

this is called the **import declaration**

- by default, every class in `java.lang` are imported automatically equivalent to having at top of your files:

`import java.lang.*;`
↑
wildcard, meaning "any" class

- Random class belongs in `java.util`

- Math class belongs in `java.lang`

the Random class

- performs complicated calculations based on a ^{initial} seed value to create a stream of seemingly random values

- e.g. Random generator = new Random();
:

```
System.out.println ( generator.nextInt() );           157
System.out.println ( generator.nextInt(10) );         9
System.out.println ( generator.nextDouble() );       0.7085915
```

- range of values of nextInt(n) produces [0 , n-1]
" nextDouble() " [0 , 1)

the Math class

- predefined math functions such as absolute value, square root, exponentials, trigonometric functions
- unlike other classes where we create an object and call methods, the Math class has **Static methods**
↳ meaning, you call the method directly from the class, without creating a ^{Math} ~~an~~ object

- e.g. Value = Math.cos(90) + Math.sqrt(delta);

- see text for more examples: § 2.6 Scanner
§ 3.2 String
§ 3.4 Random
§ 3.5 Math

if time allows

Pseudorandom Number Generator

- use of computational algorithm to produce long sequences of seemingly random results
but in fact is determined by the seed value and function used
- example of a commonly used generator:

let X_0 = seed value you give it
 P_1, P_2 = some other constants (e.g. set based on seed value)
 N = max value⁺ in the output range (ie, "up to N ")

then

$$x_i = (P_1 x_{i-1} + P_2) \bmod N \quad \text{for } i = 1, 2, 3, \dots$$

e.g. `long x0 = System.currentTimeMillis();` 13769320519;
`long p1, p2, N, xi;`
`p1 = 234;`
`p2 = 83;`
`N = 100;`
`xi = (p1 * x0 + p2) % N;` 81
`x0 = xi;`
`xi = (p1 * x0 + p2) % N;` 37
`x0 = xi;`
`xi = ...` 41
`x0 = xi;`
`xi = ...` 77
`x0 = xi;`
`xi = ...` 1