

Opening Problem

Find the sum of integers from 1 to 10, from 20 to 30, and from 35 to 45, respectively.

int sum = 0:
for (int i = 1; i <= 10; i++)
sum += i;
System.out.println("Sum from 1 to 10 is " + sum);
sum = 0;
for (int i = 20; i <= 30; i++)
sum += 1;
<pre>System.out.println("Sum from 20 to 30 is " + sum);</pre>
sum = 0;
for (int i = 35; i <= 45; i++)
sum += 1;
System.out.println("Sum from 35 to 45 is " + sum);

Defining Methods

✤ A method is a collection of statements that are grouped together to perform an operation.



Defining Methods *Method signature* is the combination of the method name and the parameter list. The variables defined in the method header are known as *formal parameters*. When a method is invoked, you pass a value to the parameter. This value is referred to as *actual parameter or argument*. A method may return a value. The <u>returnValueType</u> is the data type of the value the method returns. If the method does not return a value, the <u>returnValueType</u> is the keyword <u>void</u>. For example, the <u>returnValueType</u> in the <u>main</u> method is <u>void</u>.





Uploaded By: Haneen

Case Study: Converting Hexadecimals to Decimals

Write a method that converts a hexadecimal number into a decimal number.

ABCD =>

A*16^3 + B*16^2 + C*16^1+ D*16^0

= ((A*16 + B)*16 + C)*16+D

= ((10*16 + 11)*16 + 12)*16+13 = ?



Uploaded By: Haneen

Scope of Local Variables

A local variable: a variable defined inside a block (e.g. method, loop).

Scope: the part of the program where the variable can be referenced.

The scope of a local variable starts from its declaration and continues to the end of the block that contains the variable.

A local variable must be declared before it can be used.

Scope of Local Variables

You can declare a local variable with the same name multiple times in different **non-nesting** blocks in a method, but you cannot declare a local variable twice in nested blocks.

```
It is fine to declare i in two
                                      It is wrong to declare i in
non-nesting blocks
                                      two nesting blocks
public static void method1() {
                                         public static void method2()
  int x = 1;
                                           int i = 1;
  int y = 1;
                                           int sum = 0;
  for (int i = 1; i < 10; i++) {
                                           for (int i = 1; i < 10; i++)
  sum += i;</pre>
    x += i;
                                           }
  for (int i = 1; i < 10; i++) {
                                         1
    y += i;
                                                                  10
```

STUDENTS-HUB.com

Uploaded By: Haneen

9







- Class constants:
 - Pl
 - E
- Class methods:
 - Trigonometric Methods
 - Exponent Methods
 - Rounding Methods
 - min, max, abs, and random Methods



13

Returns <mark>e</mark> raised to the power of a.	Examples:	
log(double a) Returns the natural logarithm of a	Math.exp(1)	returns 2.71
	Math.log(2.71)	returns 1.0
Iog10(double a) Beturns the 10 based logarithm of a	Math.pow(2, 3)	returns 8.0
Returns the 10-based logarithm of a.	Math.pow(3, 2)	returns 9.0
Pow(double a, double b) Returns a raised to the power of b.	Math.pow(3.5, 2.5)	returns 22.91
• cart(double a)	Math.sqrt(4)	returns 2.0
Returns the square root of a.	Math.sqrt(10.5)	returns 3.24

Rounding Methods

- double ceil(double x) x rounded up to its nearest integer. This integer is returned as a double value.
- double floor(double x) x is rounded down to its nearest integer. This integer is returned as a double value.
- double rint(double x) x is rounded to its nearest integer. If x is equally close to two integers, the even one is returned as a double.
- int round(float x) Return (int)Math.floor(x+0.5).
- \$ long round(double x) Return (long)Math.floor(x+0.5).

16

min, max, and abs

max(a, b) and min(a, b)

Returns the maximum or minimum of two parameters.

abs(a)

Returns the absolute value of the parameter.

* random()

Returns a random double value in the range [0.0, 1.0).

Examples:				
Math.max(2, 3)	returns 3			
Math.max(2.5, 3)	returns 3.0			
Math.min(2.5, 3.6)	returns 2.5			
Math.abs(-2)	returns 2			
Math.abs(-2.1)	returns 2.1			

17

The random Method\$ Generates a random double value greater than or equal to 0.0 and less than 1.0 (0 <= Math.random() < 1.0)</p> (int) (Math.random() * 10) → Returns a random integer between 0 and 9. 50 + (int) (Math.random() * 50) → Returns a random integer between 50 and 99. In general: a + Math.random() * b → Returns a random number between a and a + b, excluding a + b.

STUDENTS-HUB.com

Uploaded By: Haneen



