

MODULE 2 – TYPES, VARIABLES AND EXPRESSIONS

ITSE 1003

Introduction to Programming Languages

Goals

- Basic data types
- Operators (part 1)
- Play!
- Variables
- Operators (part 2)
- Play!
- Homework ... even more play!

Java data types (primitive)

Type	Size (bits)	Values
boolean	Platform specific	true or false
char	16	'\u0000' to '\uFFFF' (0 to 65535)
byte	8	-128 to +127
short	16	-32,768 to +32,767
int	32	-2,147,483,648 to +2,147,483,647
long	64	-9,223,372,036,854,775,808 to +9,223,372,036,854,775,807
float	32	-3.4028234663852886e+38 to -1.40129846432481707e-45 and +1.40129846432481707e-45 to +3.4028234663852886e+38
double	64	-1.7976931348623157e+308 to -4.94065645841246544e-324 and +4.94065645841246544e-324 to +1.7976931348623157e+308

Java data types (class)

- String
 - We used String in the first program, we just did not make it obvious
- Any class, built in or created

Go live

- Look at some code examples for primitive and String

Operators (part 1)

- Arithmetic

Operation	Operator	Example
Addition	+	$1 + 2 \Rightarrow 3$
Subtraction	-	$2 - 1 \Rightarrow 1$
Multiplication	*	$2 * 3 \Rightarrow 6$
Division	/	$6 / 2 \Rightarrow 3$
Modulo	%	$3 \% 2 \Rightarrow 1$
Plus	+	$+3 \Rightarrow$ positive 3
Minus	-	$-3 \Rightarrow$ negative 3

- String

Operation	Operator	Example
Concatenation	+	"You" + " first" \Rightarrow "You first"

Operators (part 1)

- Operator precedence
 - Unary +, -
 - *, /, %
 - +, -
- Examples
 - $4 * 2 + 1 \Rightarrow 9$
 - $4 + 2 * 1 \Rightarrow 6$
 - $-4 + 2 * 1 \Rightarrow -2$
- Precedence can be overridden with parentheses
 - $4 * (2 + 1) \Rightarrow 12$
 - $(4 + 2) * 1 \Rightarrow 8$

Go live

- Look at some code examples for expressions

Activity: Play!

- Experiment with
 - Primitives
 - Expressions, including precedence and parentheses
 - Concatenation
 - Printing the primitives and the results of expressions
- For example
 - Calculate (and print) net sale of items priced at \$25, \$60, \$100
 - Calculate sales tax on a sale of the same items
 - What if there are 3 @ \$25, 2 @ \$60, 4 @ \$100

Variables

- Represent a named memory location holding data
- Variables can make intent much clearer
- Name (identifier) must be valid
 - Series of letters (a-zA-Z), digits (0-9), underscore (_), dollar sign (\$)
 - Cannot begin with a digit
 - Cannot contain blanks (spaces)
- Can be used anywhere **literals** can be used
 - We used literals in the previous examples

NOTE: There are often conventions for variable naming.

Variables

- Declaration must happen before it can be used

- `type name = initial_value;`

- Examples

- `int a = 12;`

- `byte z = 124;`

- `String ss = "Right now!";`

NOTE: The '=' is another **operator**

- Assignment
- Lowest precedence

- Java is a ***strongly typed*** language

- A variable can never change type

- This is not true of other languages, e.g., PHP, Javascript

Go live

- Replace literals with variables

Activity: Play!

- Replace the literals in your code with variables

Operators (part 2)


- Arithmetic (variables only)

Operation	Operator	Explanation
Increment prefix	<code>++x</code>	Increment x before x used
Increment postfix	<code>x++</code>	Increment x after x used
Decrement prefix	<code>--x</code>	Decrement x before x used
Decrement postfix	<code>x--</code>	Decrement x before x used

Operators (part 2)

- Assignment (variables only)

Operation	Operator	Example
Assignment	=	x = 4;
Addition assignment	+=	x += 4;
Subtraction assignment	-=	x -= 4;
Multiplication assignment	*=	x *= 4;
Division assignment	/=	x /= 4;
Modulo assignment	%=	x %= 4;



NOTE: There are other assignment operators covered in more advanced courses.

Operators (part 2)

- Operator precedence
 - ++, -- highest
 - *, /, % next
 - +, - next
 - Assignment last
- Precedence can be overridden with parentheses
- Variables can be declared without initial value
 - `type name;`
 - Example
 - `int aww;`
- Use assignment operator to give variable a value
 - `aww = 98;`

Go live

- Demonstrate new operators

Activity: Play!

- Write your own program with
 - Variables
 - Whatever operators you chose
- Some possibilities
 - Investment growth
 - Start with initial principle and annual interest rate
 - Display balance for every year for 5 years
 - Loan amortization
 - Start with a loan amount, a 6 month term, some rational interest rate
 - Calculate payment to pay off loan in 6 months, including interest
 - Display principle balance for each of 6 months

Homework

- Approximate the value of pi (π) using the formula:

$$\pi = 4 \sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1}$$

- $\pi = 4 * (1 - 1/3 + 1/5 - 1/7 \dots)$
 - Where $i = 0, 1, 2, \dots n$, where n is as big as you can tolerate
 - As n gets larger, the precision gets better
 - Print the approximation after every subtraction and addition
-
- NOTE: this will converge very slowly, so don't be disappointed if you never get to 3.14159....