


Welcome to CS 101!

Introduction to Java Programming

Jamal Abdul Nasir

Previous Lectures to watch?


Playlist name: Introduction to Java Programming



Introduction to Java Programming

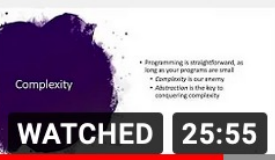
≡ SORT

≡



[Lecture-1] CS-101 Introduction to Java Programming
Learn IT Skills

≡



[Lecture-2] CS-101 Introduction to Java Programming
Learn IT Skills

Course Learning Objectives

- Computational Thinking (Decomposition into sub-problems)
- Functionality/Behavior (Working solution of these sub-problems)
- Control Structures (Methods, loops, conditions)
- Data Abstraction (Variables, parameters, arrays, classes)
- Code Quality (programs that are well-written, readable, maintainable, and conform to established standards)

Abstraction (Detail Removal)

- Modern user interface: Right pedal is “accelerate”, left is “decelerate”



Maps for directions

Abstraction: Generalization

- “The process of formulating general concepts by abstracting common properties of instances.”
- Extensible shower rods
- Adjustable hats
- Single recipe for <fruit> milkshake
- Feeding animals on a farm
 - To feed <animal>, put <animal> food in <animal> dish

Algorithm

- Computational thinking is to divide problems into sub-problems
- Then algorithm is a **step-by-step** procedure for solving each sub-problem

What is programming?

- **program:** A set of instructions to be carried out by a computer.
- **program execution:** The act of carrying out the instructions contained in a program.
- **programming language:** A systematic set of rules used to describe computations in a format that is editable by humans.
 - We will be studying a programming language called Java

Your first Java program!

```
public class Hello {  
    public static void main(String[] args) {  
        System.out.println("Hello, world!");  
    }  
}
```

- File must be named `Hello.java`

Running a program

- Write it
- Compile it
- Execute it

Running a program

```
public class Hello {  
    public static void main(String[] args) {  
        System.out.println("Hello, world!");  
    }  
}
```



Structure of a Java program

```
public class name {  
    public static void main(String[] args) {  
        statement;  
        statement;  
        ...  
        statement;  
    }  
}
```

class: a program

method: a named group of statements

statement: a command to be executed

Legal Identifier

- `_myName`
- `TheCure`
- `ANSWER_IS_42`
- `$bling$`
- `Age`
- `Full_name`

Illegal Identifier

- me+u
- 49ers
- side-swipe
- Ph.D's

Keywords

- **keyword:** An identifier that you cannot use because it already has a reserved meaning in Java.

abstract	default	if	private	this
boolean	do	implements	protected	throw
break	double	import	public	throws
byte	else	instanceof	return	transient
case	extends	int	short	try
catch	final	interface	static	void
char	finally	long	strictfp	volatile
class	float	native	super	while
const	for	new	switch	
continue	goto	package	synchronized	



System.out.println

System.out.println

- A statement that prints a line of output on the console.
 - pronounced "print-linn"
- Three ways to use `System.out.println`:
 - `System.out.println("text");`
Prints the given message as output.
 - `System.out.println(text);`
Prints the value of variable 'text'
 - `System.out.println();`
Prints a blank line of output.



Strings and escape sequences

Strings

- A sequence of text characters.
 - Starts and ends with a " (quotation mark character).
 - The quotes do not appear in the output.
 - Examples:
`"hello"`
`"This is a string. It's very long!"`

Strings

- Restrictions:

- May not span multiple lines.

```
"This is not  
a legal String."
```

- May not contain a " character.

```
"This is not a "legal" String either."
```

Escape sequences

- **escape sequence:** A special sequence of characters used to represent certain special characters in a string.

<code>\t</code>	tab character
<code>\n</code>	new line character
<code>\"</code>	quotation mark character
<code>\\</code>	backslash character

Escape sequences

- Example:

```
System.out.println("\\hello\\nhow\\tare \\\"you\\\"?\\\\\\\\\");
```

- Output:

```
\\hello
```

```
how      are  "you"?\\
```

Question

- What is the output of the following `println` statements?

```
System.out.println("\ta\tb\tc");  
a           b           c
```

```
System.out.println("\\\\\\" );  
\\
```

Question

- What is the output of the following `println` statements?

```
System.out.println("' '");
```

```
System.out.println("\\""\\"""");
```

Question

- What is the output of the following `println` statement?

```
System.out.println("C:\nin\the downward spiral");
```

```
C:  
in      he downward spiral
```


Question

- Write a `println` statement to produce this output:

/ \ // \\ /// \\\

Question

- Write a `println` statement to produce this output:

```
/ \ // \ \ /// \ \
```

- `println` statement to produce the line of output:

```
System.out.println("/ \ // \ \ \ \ /// \ \ \ \ \ \");
```

What println statements will generate this output? (Add your answers in the comments)

- [question-1] What `println` statements will generate this output?

```
This quote is from  
Very famous Irish poet Oscar Wilde:
```

```
"When I was young  
I thought  
that money was the most important thing in life;  
now that I am old  
I know that it is."
```

The End

- Practice makes a man perfect!
- If you have questions/comments: Comment on Youtube Video.
- Please like this video, share with your friends and keep Learning!!!
Bye!