Java Basics I

Christian Rodríguez Bustos Object Oriented Programming

Java Basics



Agenda

Last Class

Exercise

Why Java?

Java programs structure

Java basic

applications Java Basics



Last Class Exercise

1. Do the Eclipse HelloWord!! or NetBeans HelloWord!! 2. Modify the “Multidimensional array use example ”code in order to:

– print the main diagonal of the next two multidimensional arrays

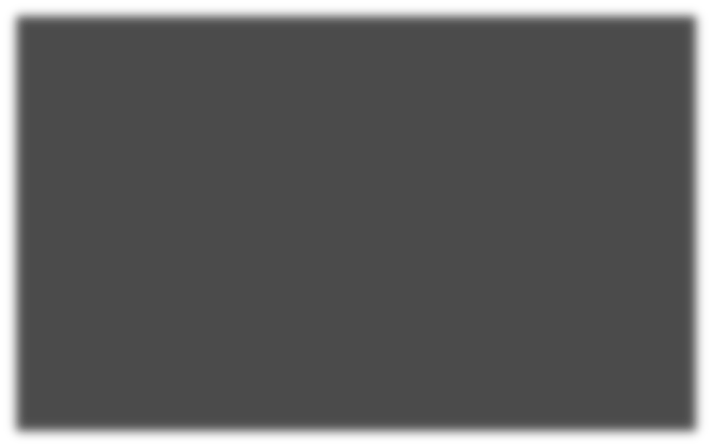
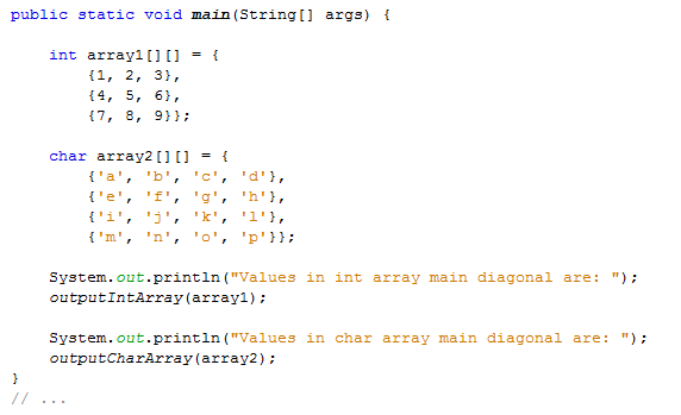
| a | b | c | d |
| --- | --- | --- | --- |
| e | f | g | h |
| i | j | k | l |
| m | n | o | p |

| 1 | 2 | 3 |
| --- | --- | --- |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

Numbers array

Letters array

Java Basics

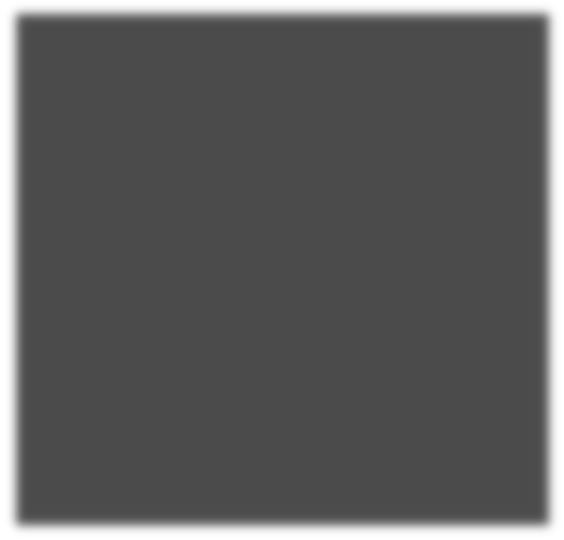
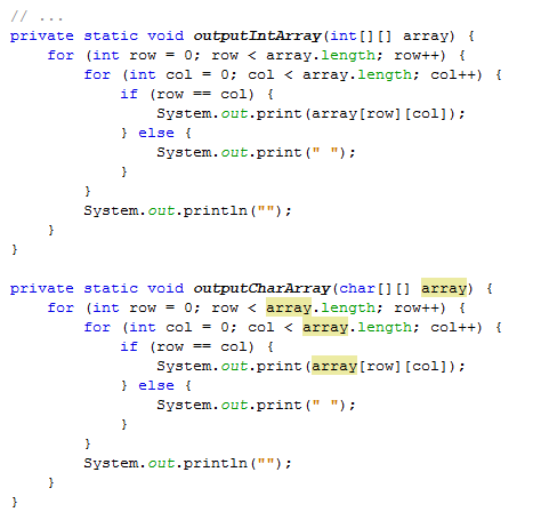


Int Array

Char Array



Java Basics



Method for

printing int

arrays

Method for

printing

char arrays

Why Java?

Java advantages Java acronyms

Java Basics



Java Is Free!

Java Is an Open Standard

Java advantages

Java Is

Architecture

Neutral

Java Provides

“One-Stop

Shopping”

Java Is Object

Oriented from

the Ground Up

Practice Makes

Perfect

Java Basics



Java Is Architecture Neutral

Java compiler  for Windows 

Java compiler  for Linux 

Platform

independent

Java source

code

Java compiler for Solaris 

C++ compiler  for Windows 

C++ compiler  for Linux 

Platform

independent

C++ source

code

C++ compiler for Solaris 

Windows version Linux version Solaris version 

Platform independent bytecode

(.class file) 

Java Basics

Java Is Architecture Neutral

The Java Virtual Machine (JVM)

converts the compiled Java byte code to machine code.JVM A 

JVM B 

Platform independent bytecode

(.class file)

The same bytecode can be understood by different JVMs

In theory, bytecode is forward compatible with newer versions of the JVM

JVM C JVM D 



Java Basics

Java Provides “One-Stop Shopping”

Java language provides an extensive set of application

programming interfaces (APIs)



java.io: Used for file system access

java.sql: The JDBC API, used for communicating with relational

databases in a vendor-independent fashion

java.awt: The Abstract Windowing Toolkit, used for GUI development

javax.swing: Swing components, also used for GUI development

And there are many more …



Java Basics

Java Is Object-Oriented from the Ground Up

Primitive or simple data types are still just single pieces of information

Object-oriented objects are complex types that have multiple pieces of information and specific properties(or attributes) and behaviors(methods).

Java Basics

Java Is Object-Oriented from the Ground Up

All data, with the exception of a few 

primitive types are objects.

All of the GUI building blocks windows,

buttons, text input fields, scroll bars, lists,

menus, and so on are objects.

All functions are associated with objects

and are known as methods there can be

no “free-floating” functions as there are in

C/C++.

Java Basics



Practice Makes Perfect

Java taken the best features of

C++,Eiffel, Ada, and Smalltalk

Added some capabilities and

features not found in those

languages.

Features that had proven to be

most troublesome in those

earlier languages were

eliminated.

Java Basics



Java Is an Open Standard

Java Basics



Java Is Free!

Java Basics



Java Is Free!

Java Is an Open Standard

Java advantages

Java Is

Architecture

Neutral

Java Provides

“One-Stop

Shopping”

Java Is Object

Oriented from

the Ground Up

Practice Makes

Perfect

Java Basics



Java acronyms Java Basics

JDK: Java Developer Kit: Develop and execution

JRE: Java Runtime

Environment: Execution

JVM: Java Virtual Machine 

Java acronyms

Java EE

Enterprise Edition

Java SE

Standard Edition

Java ME

Micro Edition

Java Basics



Java programs structure

File Structure

Classes

Methods

Java Basics



A source code file holds one class definition 

Same name

Put a class in a source file !!!

Java Basics

A class holds one or more methods

Method 1 

Method 2

Method 3

Put methods in a class !!! 

Java Basics

A method holds statements

Method 1 

Statements

Method 2

Statements

Put statements in a method !!! 

Java Basics

File structure

Source fileClassMethod

Method

Put a class in a source file !!!

Put methods in a class !!!

Put statements in a method !!!

Java Basics



Define who can access the class:

public means everyone can access it

Class definition

This is a

class Class name



Closing

brace

Java Basics

Opening

brace



Class names Should be nouns

• Should be nouns, in mixed case with the first letter of each internal word

capitalized.

• Try to keep your class names simple and descriptive.

• Use whole words, avoid acronyms and abbreviations. • Java is case sensitive.

Java Basics

Good Examples:

• class **SoccerPlayer** {…} • class **Person** {…}

Bad Examples

• class **XYZ** {…}

• class **PERSON** {…}

• class **soccerplayer** {…}

Methods definition

Define who can access the class:

public means everyone can access it

We will see this later ☺



Method name

Parameters to the

method

Return

type

Closing

brace

Java Basics

Opening

brace



Methods names Should be verbs

Good Examples:

• private static void **play**(int coinValue) {…}

• public static void **moveToRight**(int steps) {…}

• public static void **getDirection**() {…}

Bad Examples

• public static void **person**() {…}

• public static void **PLAY**() {…}

• public static void **soccerplayer**() {…} Java Basics

Should be verbs (behaviors), in mixed case with the first letter lowercase, with the first letter of each internal word capitalized.



The main method is where your program start to run 

Is not necessary a

main method in a

class



Java Basics

Java basic applications

Comments, operators and precedence

System.out and System.in



Comments improve readability of source code



Most of the times ;)





A good source code do not 

required comments



What is the best comment in source code you have ever encountered

Java Basics

Self explanatory code vs Commented code

Self explanatory

Commented code



Java Basics

Precedence of arithmetic operators Java Basics



Precedence of arithmetic operators



??????



Java Basics

Precedence of arithmetic operators





Java Basics

System.out is the "standard" java output stream

This stream is already open and ready to accept output data.

Print an object

System.out.print(Object object);

Print an object using a specific format

System.out.print**f**(String format, Object object);

Print an object in a new line

System.out.print**ln**(Object object);



Java Basics

System.out is the "standard" java output stream

System.out.**print**(); 

Print without moving cursor to

the next line

System.out.**println**(); 

Print moving cursor to the next

line



Java Basics

System.out is the "standard" java output stream 



Java Basics

System.out is the "standard" java output stream

System.out.**printf**();

Print without moving cursor to the next line



Java Basics

Printing formats: Numbers with zeros 

This is the format

Java Basics



Printing formats: Letters

This is the format 

Java Basics



Printing formats: Dates

Remember 

to import 

the Date

Class

This is the format 

Java Basics



Printing formats: Floats





This is the format 

Java Basics

Printing formats resources

See more formats on

http://www.java2s.com/Code/JavaAPI/java.lang/System. out.printf.htm



Java Basics

Escape sequences

Java Basics



Escape sequences: new line and tab examples

Java Basics

Escape sequences: slash and double quote examples Java Basics

System.in is used to read user inputs

System.in and Scanner class allow us to read values typed by the user

First we need to import 

the Scanner class at the

beginning of our source

code file



Java Basics

System.in: reading strings example



Creating the

scanner



Reading

an integer



Java Basics

System.in: reading strings example



Creating the

scanner



Reading

a String



Java Basics

System.in: reading posibilities Java Basics



References

• [Barker] J. Barker, Beginning Java Objects: From Concepts To Code, Second Edition, Apress, 2005.

• [Deitel] H.M. Deitel and P.J. Deitel, Java How to Program: Early Objects Version, Prentice Hall, 2009.

• [Sierra] K. Sierra and B. Bates, Head First Java, 2nd Edition, O'Reilly Media, 2005.

• Code Conventions for the Java Programming Language, available at http://java.sun.com/docs/codeconv/CodeConventions.pdf



Java Basics