



Ca' Foscari
University
of Venice

Object oriented programming


Module 1

Pietro Ferrara

pietro.ferrara@unive.it



Eiffel tower

- 18,038 metallic parts
- 5,300 workshop drawings
- 50 engineers and designers
- 150 workers in the Levallois-Perret factory
- Between 150 and 300 workers on the construction site
- 2,500,000 rivets 
- 7,300 tonnes of iron
- 60 tonnes of paint
- 2 years, 2 months and 5 days of construction
- History here

La tour Eiffel prise du Champ-de-Mars - Exposition universelle de 1889
© Parisienne de photographie - Neurdein / Roger-Viollet
https://artsandculture.google.com/asset/_/hQEdYmo3Gzetsw



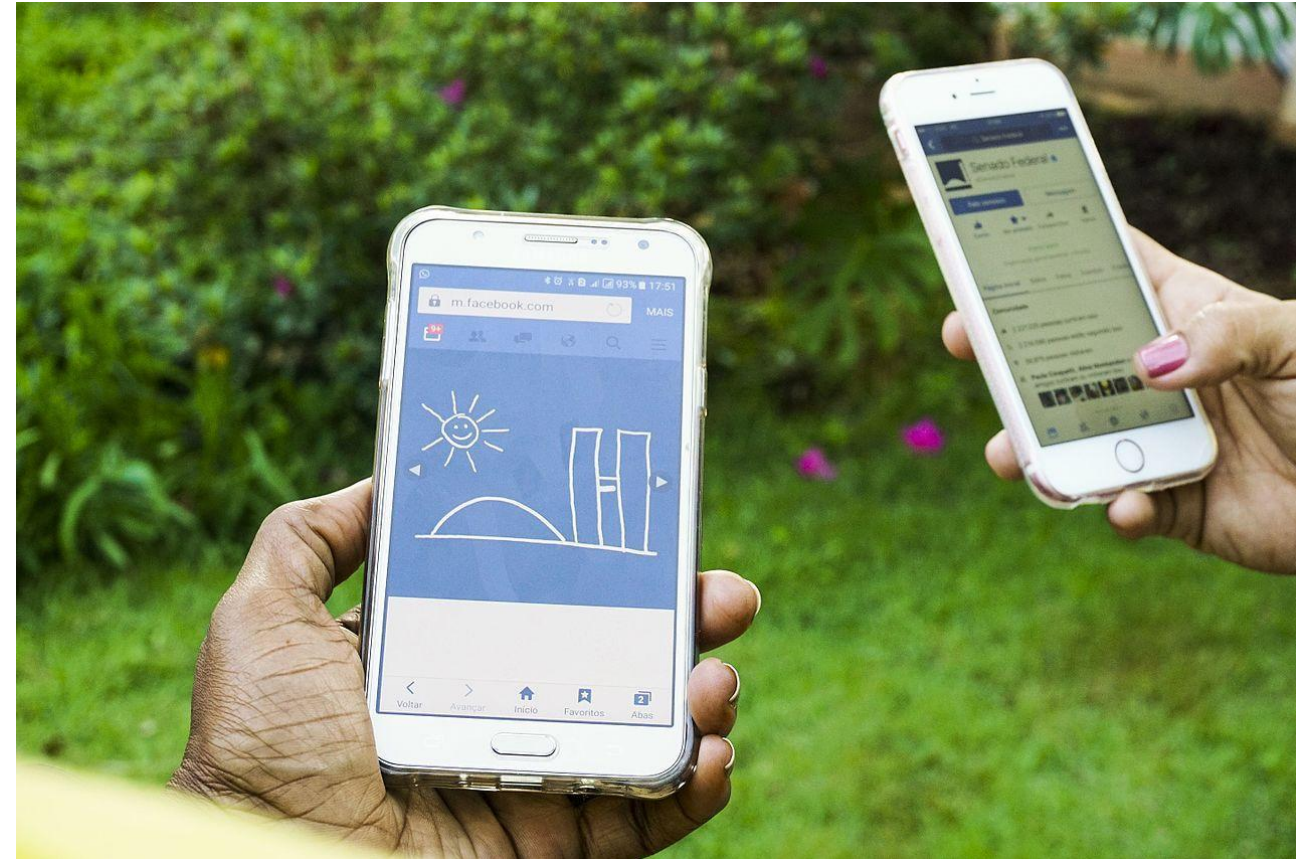


Engineering

- Most of the tower was built somewhere else from some workers
- On the site pieces were only assembled, not constructed!
 - With first temporary rivets and then assembled rivets
 - But still, only a third of the rivets were used on site!
- About 500 workers involved in the construction with different skills
- They “communicated” through some standard means
 - Such as rivets, drawings, etc...
- A worker using some results of other works did not need to know the details of what the others have done!

Smartphone (just as an example)

- ... what do we mean by smart?
- Android operating system
 - <https://source.android.com/>
 - <https://github.com/aosp-mirror>
- 99 GitHub repositories
- “Main” repository
 - ~ 280 branches
 - > 30K commits
 - Almost 150 contributors
 - > 10MLOCs



By Senado Federal - Fotos produzidas pelo Senado, CC BY 2.0,
<https://commons.wikimedia.org/w/index.php?curid=53990377>



Using technology

- <https://www.youtube.com/watch?v=IkRGBf9mUqw>
 - Non ci resta che piangere, 1984

- Everyday as users we rely on some technologies we do not know at all
- The interface of the technology is clear
- The details of the technology are hidden
 - And we should not care at all about them!
- The same must happen for software
 - We rely on code written by others (libraries) using some interface
- ... or do you want to build up an app starting from the operating system???



Ca' Foscari
University
of Venice

Your Eiffel tower

3rd year: how to supervise the construction of the tower
Ingegneria del software, Reti di calcolatori, Sicurezza,
Project management, and much more...

2nd year (now): how to build up more complex systems
packaging and assembling the 1st year basic blocks
Programmazione ad oggetti, Basi di dati, Sistemi operativi,
Algoritmi e strutture dati

1st year: how to build the basic blocks of the tower
Introduzione alla programmazione, Architettura degli
elaboratori, Programmazione e laboratorio





Object oriented

- This course (Java)

*Based on the concept of "objects", which can contain data and code: data in the form of **fields** (often known as attributes or properties), and code, in the form of procedures (often known as **methods**).*

*A feature of objects is that an object's own procedures **can access and often modify the data fields** of itself (objects have a notion of "this" or "self"). OOP languages are diverse, but the most popular ones are **class-based**, meaning that objects are **instances** of classes, which also determine their types.*

programming

- Imperative:
 - Introduction to programming (C)
 - Programming and lab (C++)

*In computer science, imperative programming is a programming paradigm that uses **statements** that **change a program's state**. In much the same way that the imperative mood in natural languages expresses commands, an imperative program consists of **commands for the computer to perform**. Imperative programming focuses on describing **how** a program operates.*



John von Neumann (1903-1954)

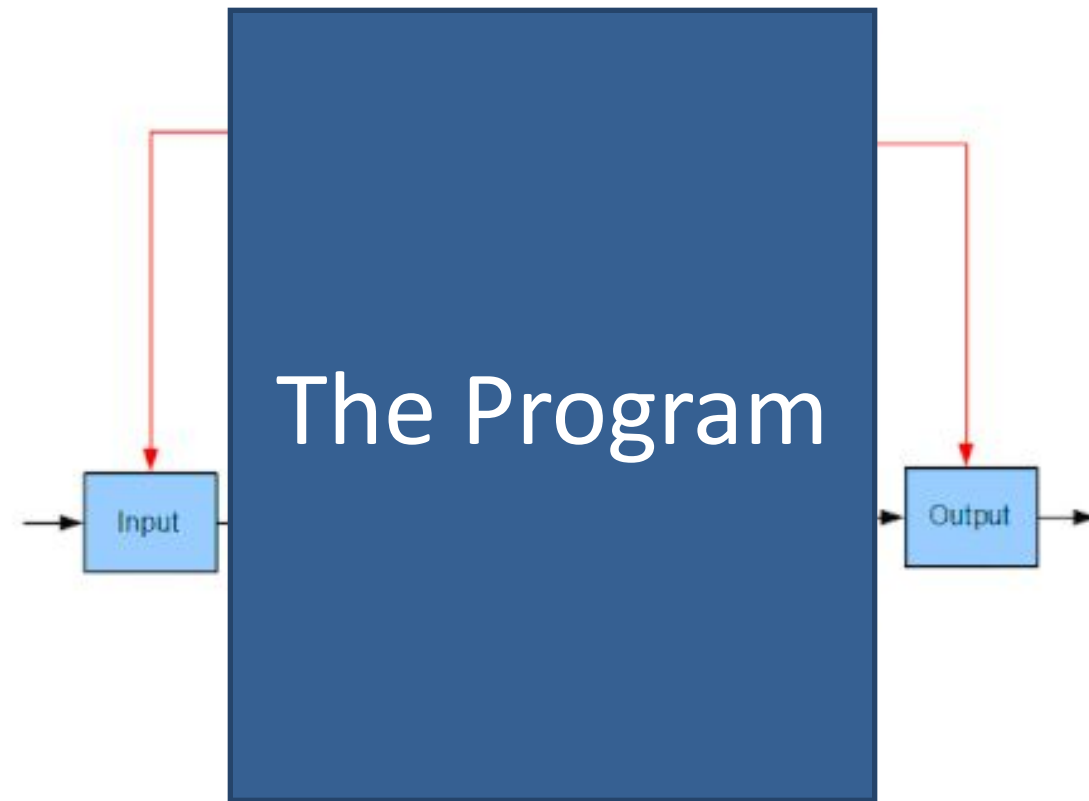
- Mathematician, physicist, computer scientist, and polymath
- His work spanned from pure mathematics to the Manhattan project
 - Produced the first nuclear weapons
- He introduced the first formalization of computer architecture
 - The so-called von “Neumann architecture”





von Neumann architecture

- Central Processing Unit (CPU)
- Main components
 - Control Unit
 - Manage the computations of the CPU
 - An arithmetic & logic unit
 - Execute the logic operations
 - Memory
 - Store operations and data under execution
 - Input
 - Output → Peripherals

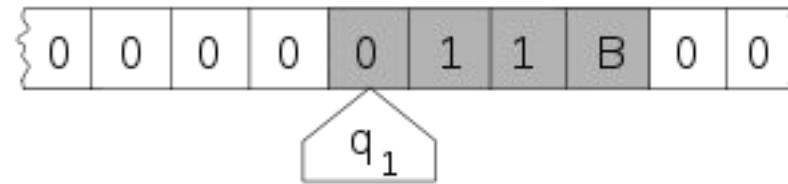




Turing machine

https://en.wikipedia.org/wiki/Turing_machine

- Turing machine (1936) defined by Alan Turing
- Infinite memory tape divided into discrete cells
- Has a "head" over a cell and can read it
- Then the machine can either
 - write a symbol in the cell
 - move the tape one cell left or right
 - proceed to the next instruction or halt the computation
 - Based on the observed symbol
- Any computer program can be executed in a Turing machine





So, what is programming?

- Programming is the ability to perform extremely complex tasks based on a very simple/minimalist/stupid machine
 - (be careful: that's MY definition! 😊)
- Programming languages are much more expressive
 - But at the end they are a machine that executes simple instructions
 - In a mechanical way
- When we develop a program, we need to instruct a mechanical machine on how to perform very complex tasks
 - All the logic stays on our side!
 - Being precise and “exact” is the key to success



Notable examples

- Margaret Hamilton (1936)
- In charge of the Command Module software of the Apollo project
- Picture (1969): all the code of such module
 - And this was essential to allow the man to go to the moon
 - ... and to save some lives! (see Apollo 11)





Notable examples

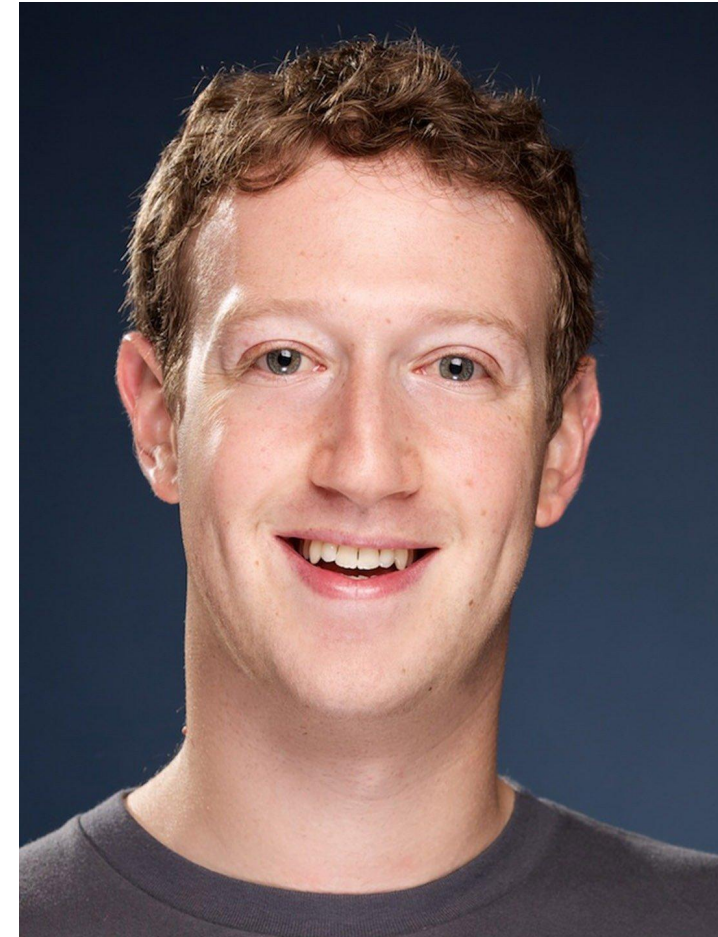
- Childhood friends Bill Gates (1955) and Paul Allen
 - business with their skills in computer programming
- 1972: first company, Traf-O-Data
 - computer to track and analyze automobile traffic data
- March 1975: BASIC interpreter for Altair 8800
- Microsoft officially established on April 4, 1975
- August 1977 agreement with ASCII Magazine in Japan
- January 1979: moved to a new home in Bellevue, Washington
- 1980: enter the operating system business: MS-DOS, Windows 3.1,





Notable examples

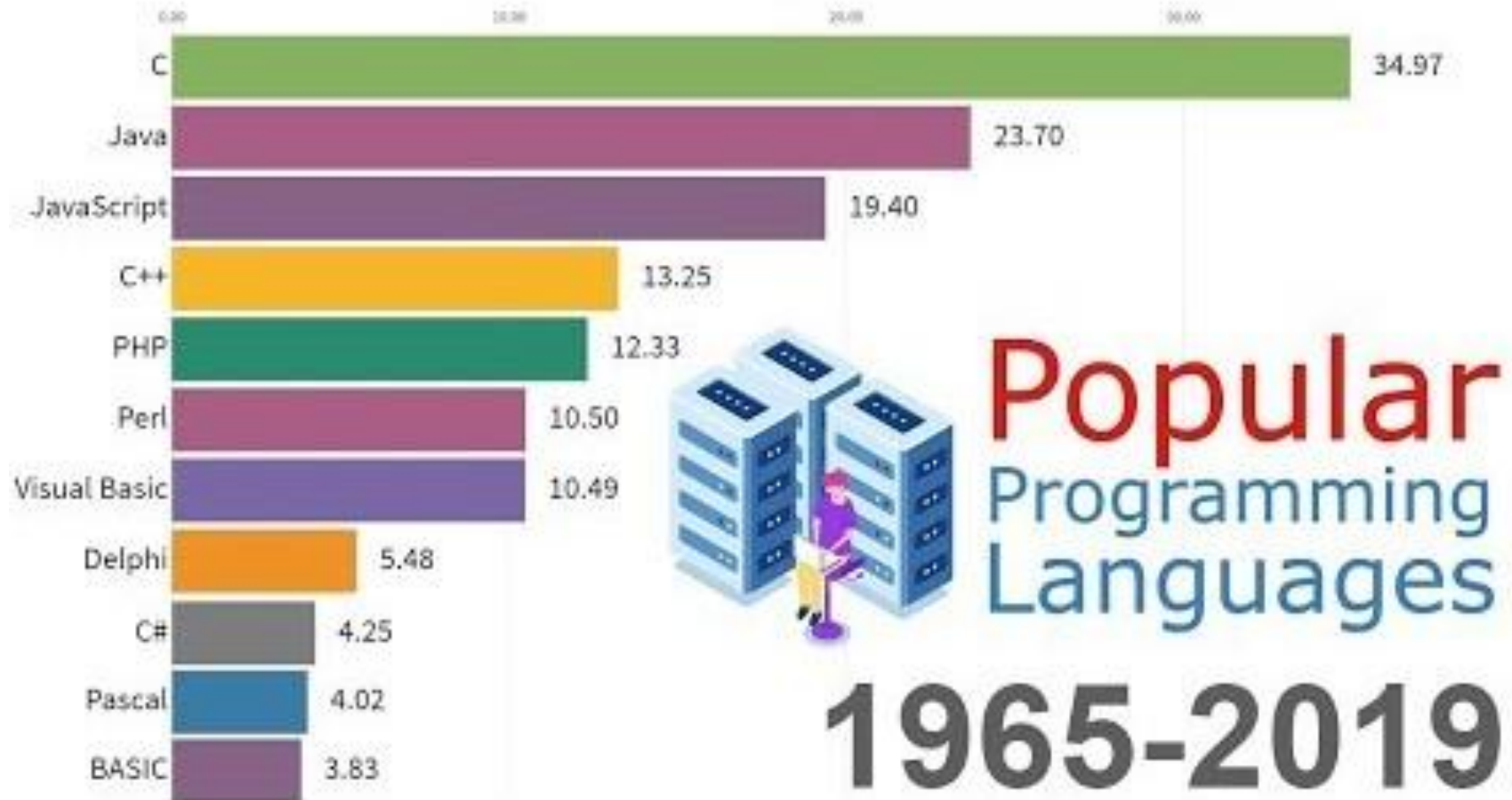
- Mark Zuckerberg (1984)
- 2004: launched the Facebook social network
- 2007: the world's youngest self-made billionaire
- 2012: 1 billion users
- January 2021: 4th-richest person in the world
 - the only < 40 in Forbes' 20 richest people
- Facebook has nowadays quite an impact in any aspect of the society
 - ... from elections to vaccination!





Ca' Foscari
University
of Venice

Most popular programming languages



Funny history of
programming
languages

<https://www.youtube.com/watch?v=Og847HVwRSI>



Takeout 1/2

1. Programming is frustrating
 - We need to make stupid machines clever
2. Programming is fascinating
 - We need to make stupid machines clever
3. Programming has quite an impact nowadays
 - Any activity relies on computers in some forms



Ca' Foscari
University
of Venice

Pietro Ferrara



Unsafe C# code
Aug-Nov 2007: intern

Thesis about Java multithreading

PhD in CS

Thesis about JavaCard

Master in CS

Bachelor in CS

OO programming
and Scala
Postdoc and lecturer

Android and
JavaScript
Research Staff
Member

.NET and Java
Head of R&D

Assistant
professor

Associate
professor

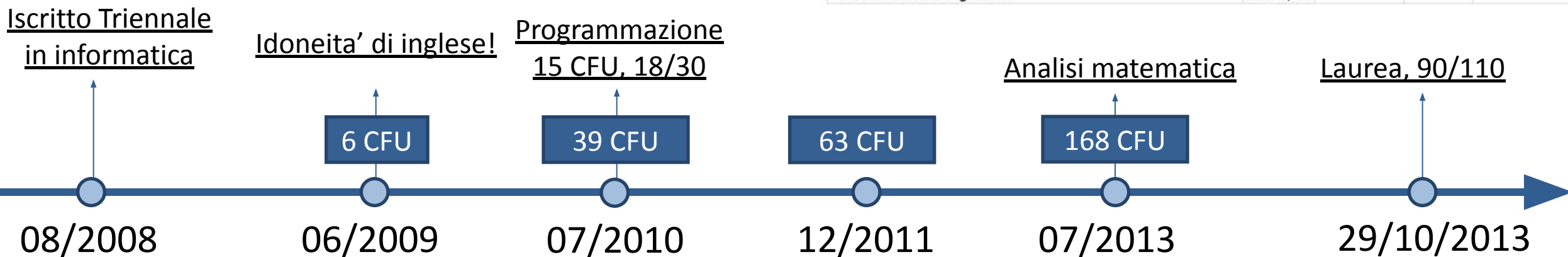




Carriera universitaria

Attività Didattiche	Anno	Peso in crediti	AA Freq.	Voto - Data Esame
CT0004 - ANALISI MATEMATICA	1	9	2008/2009	20 - 06/06/2013
CT0005 - ARCHITETTURA DEGLI ELABORATORI	1	12	2008/2009	22 - 16/06/2010
CT0101 - LINGUA INGLESE	1	6	2008/2009	IDO - 30/06/2009
CT0000 - LOGICA-MATEMATICA (OBBLIGO FORMATIVO AGGIUNTIVO)	1	0	2008/2009	ES - 19/08/2008
CT0103 - MATEMATICA DISCRETA	1	12	2008/2009	28 - 22/06/2010
CT0113 - PROGRAMMAZIONE	1	15	2008/2009	18 - 07/07/2010
CT0178 - LINGUAGGI PER LA RETE: XML	2	6	2010/2011	26 - 08/09/2011
CT0114 - PROGRAMMAZIONE A OGGETTI	2	9	2009/2010	22 - 28/05/2012
CT0153 - PROJECT MANAGEMENT E QUALITA' DEL SOFTWARE	2	6	2009/2010	24 - 13/09/2011
CT0125 - SISTEMI OPERATIVI	2	12	2009/2010	23 - 18/05/2012
CT0002 - ALGORITMI E STRUTTURE DATI	2	9	2009/2010	23 - 25/05/2012
CT0006 - BASI DI DATI	2	12	2009/2010	21 - 17/01/2012

Attività Didattiche	Anno	Peso in crediti	AA Freq.	Voto - Data Esame
CT0091 - INTERAZIONE UOMO-MACCHINA	2	6	2009/2010	23 - 27/05/2011
CT0011 - CALCOLABILITA' E LINGUAGGI FORMALI	3	9	2010/2011	23 - 07/01/2013
CT0158 - COMMERCIO ELETTRONICO	3	6	2011/2012	28 - 22/05/2012
CT0066 - DIRITTO DELL'INFORMATICA	3	6	2010/2011	26 - 06/06/2012
CT0090 - INGEGNERIA DEL SOFTWARE	3	6	2010/2011	26 - 09/01/2012
CT0117 - PROVA FINALE	3	6	2010/2011	IDO - 29/10/2013
CT0119 - RETI DI CALCOLATORI	3	9	2010/2011	27 - 11/09/2013
CT0120 - RICERCA OPERATIVA	3	6	2010/2011	18 - 11/09/2012
CT0128 - STAGE/TIROCCINIO	3	12	2010/2011	AP - 01/02/2013
CT0142 - TECNOLOGIE E APPLICAZIONI WEB	3	6	2010/2011	26 - 24/05/2012
Media Aritmetica degli esami	23.556 / 30			
Media Ponderata degli esami	23.173 / 30			



Jacopo Ferrara, my brother!

<https://www.linkedin.com/in/jacopo-ferrara/>



Carriera lavorativa

IMA Servizi

- manutenzione distributori di benzina
- strutturare un ufficio IT
- Gestione crm della Infor EAM
- diversi portali web per preventivazione, controllo distributori
- Php, db relazionale in PostreSql, JavaScript,

Pellegrini

- noleggio stampanti
- consulente Arxivar
- software documentale
- ASP, C#

Previnet

- settore assicurativo
- sviluppatore Web in Previmedical
- manutenzione di processo e portali
- Perl, JavaScript, HTML, Struts, ...

Alpenite

- consulenza ecommerce
- certificato Salesforce Commerce Cloud
- technical leader progetto Illy
- architect per Bulgari

IMA Servizi

Stage, assunzione

Fallimento IMA Servizi

Pellegrini, assunzione

Previnet

Assunzione

Alpenite

Assunzione

2012

2016


2017

2018




Do we still need developers?

Laurea triennale

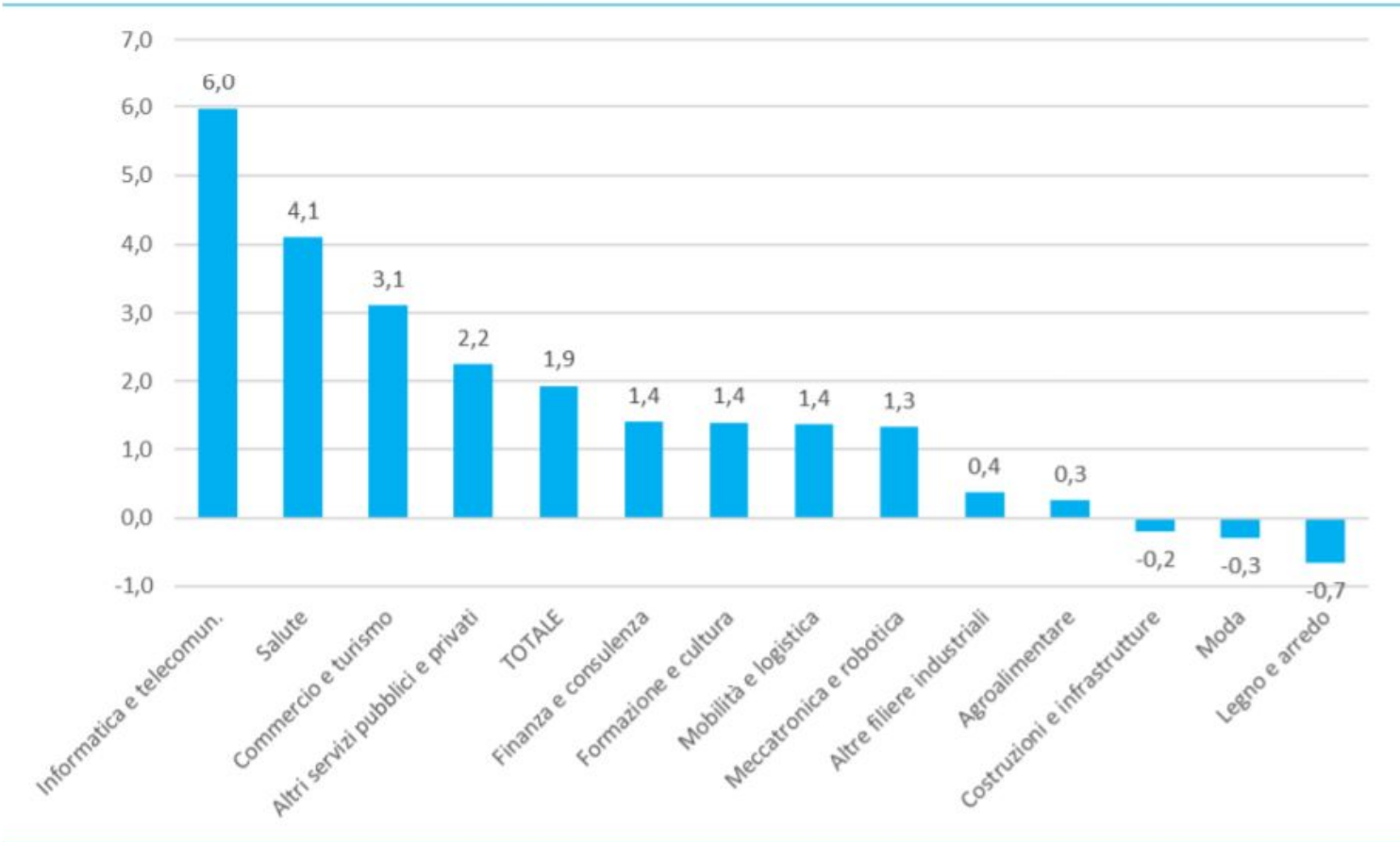
3. Condizione occupazionale	Collettivo selezionato
Condizione occupazionale e formativa (%) 	
Lavorano e non sono iscritti ad una laurea di secondo livello	42,9
Lavorano e sono iscritti ad una laurea di secondo livello	12,1
Non lavorano e sono iscritti ad una laurea di secondo livello	40,0
Non lavorano, non sono iscritti ad una laurea di secondo livello e non cercano	2,6
Non lavorano, non sono iscritti ad una laurea di secondo livello ma cercano	2,5

Laurea magistrale

3. Condizione occupazionale	Collettivo selezionato
Condizione occupazionale (%) 	
Lavorano	78,2
Non lavorano e non cercano	18,4
Non lavorano ma cercano	3,4

Do we still need developers?

FIGURA 7 – VARIAZIONI PERCENTUALI DEGLI STOCK DI OCCUPATI PREVISTE NEL 2024 RISPETTO AL 2021, PER FILIERA – SCENARIO A



In particolare, si prevede (...) nella filiera dell'“informatica e telecomunicazioni” (...) attesa **un'ulteriore accelerazione** della trasformazione digitale proprio per le conseguenze economiche della crisi sanitaria.

Un rilevante **ostacolo** alla crescita di questa filiera potrebbe essere però rappresentata **dall'elevata difficoltà di reperimento di molte delle figure richieste**, come evidenziato nel caso delle **professioni con competenze digitali di elevato profilo**.



Takeout 2/2

1. Failing an exam is frustrating
2. Failing many exams in a row is more frustrating
3. But failure is part of the learning process
 - Not passing an exam means that one specific day you did not perform well enough to pass the exam
 - It does not mean you are lazy, stupid, will fail the exam forever, etc..
4. Briefly, never give up!
 - Because we need more and more skilled developers
 - And you are one of them!

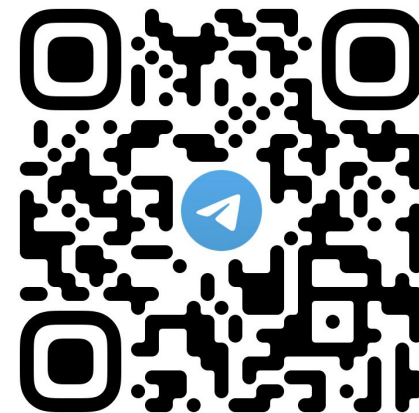


Communicating with professor

- Moodle Forum
 - Then the same forum, then the same forum...
- Obviously, you are still free to send me emails!
 - But please use the forum if it is a generic questions about the content of the course, OOP, etc..
 - And for questions about exams, grades, etc.. please always give a look to these slides before asking, usually the answer is already here
- E-mail address: pietro.ferrara@unive.it
- Linkedin: <https://www.linkedin.com/in/pietroferrara/>
 - Feel free to connect with me!



- Alessio Campanelli
 - 878170@stud.unive.it
- Weekly exercise sessions
 - Links and announcements through our Moodle space
- Some online exercise sessions
- An optional project to practice some coding during the course
- Telegram group for Q&A
 - <https://t.me/+uUxC-Ifi0yM1MDBk>





Exercises, project, and exam

- 8/9 exercises during the course in HackerRank
 - At weeks 3/4 to 11/12
 - They are optional but highly recommended
 - They will be automatically corrected by the system
- Project during the course
 - Starting in ~ a month
 - Optional but highly recommended
 - Probably about playing some card games
- Written exam at the end of the course
- In order to record the grade, you need to pass both mod. 1 and 2



Module 1 and 2

- 1 exam session in January, 2 in June, 1 in September
- Only in January, you can take only module 1
- Only in the first session of June, you can take only module 2
 - If you passed module 1 in January
 - You cannot take only module 2 in the other sessions
- In all the other sessions you'll need to take together the two modules
- Previously the rules were different
 - If you passed module 1 in 2022 you are still allowed to take only module 2 in all 2024 exam sessions



1. Encapsulation e abstraction

- a. Classi e oggetti, campi e metodi
- b. Static e final
- c. Aliasing
- d. Information hiding, attributi di visibilit 
- e. Moduli
- f. Documentazione del codice, Javadoc e file jar
- g. Java Virtual Machine e Java bytecode



2. Polimorfismo

- a. Estensione di classi, overriding e overloading
- b. Abstract e final
- c. Polimorfismo, subtyping, principio di sostituzione
- d. Tipi statici e dinamici
- e. Ereditarietà singola e multipla
- f. Classi astratte, interfacce
- g. Dispatching statico e dinamico
- h. Tipi generici



3. Java in action

- a. Gerarchia delle classi Java, classe Object
- b. Tipi native types, autoboxing, stringhe
- c. Eccezioni
- d. Annotazioni
- e. Reflection
- f. Gestione delle librerie, Gradle
- g. ? Il pattern MVC (Model-View-Controller)
- h. ? Spring
- i. ? Record
- j. ? Streams



- My lecture notes!
- Optional:
 - Ken Arnold, James Gosling, David Holmes: The Java Programming Language, 4th Edition
 - Ken Arnold, James Gosling, David Holmes: Il linguaggio Java: manuale ufficiale, quarta edizione
 - Timothy Budd, An Introduction to Object-Oriented Programming, 3rd edition
 - Available in the library (BAS)
 - Not needed, just if you want to deepen some topics for your interests



Additional material

- We must provide additional material to some students
 - Working
 - With difficulties in learning
 - See here <https://www.unive.it/pag/42819/>
- As additional material, those students will have access to the recording of my lectures in the Moodle space
 - Note that I do not/cannot decide who is allowed to have access
 - In case you think you should have access but you don't have it, contact directly Ca' Foscari offices



Why in English?

- The course (and the CS bachelor programme) is in Italian
- Why slides and textbook are in English?
- English is the lingua franca of Computer Science
 - for software developers as well!
- All the documentation of libraries is in English
- Most of the forums are in English
- If you google in English, you get quite better answers!
- So... get used to English if you want to be a developer 😊
 - Even more if you want to be a scientist...



Technologies

- We will cover various technologies during this course
 - IDE (IntelliJ IDEA in particular)
 - Java
 - Git[Hub]
 - Premium account of GitHub for free for students and professors, <https://education.github.com/benefits>
 - Maven and Gradle (just a small bit)
 - Stackoverflow or ChatGPT? 😊



Ca' Foscari
University
of Venice

Just a couple of quick questions

Your wooclap poll will be displayed here



Install the **Chrome** or
Firefox extension



Make sure you are in
presentation mode

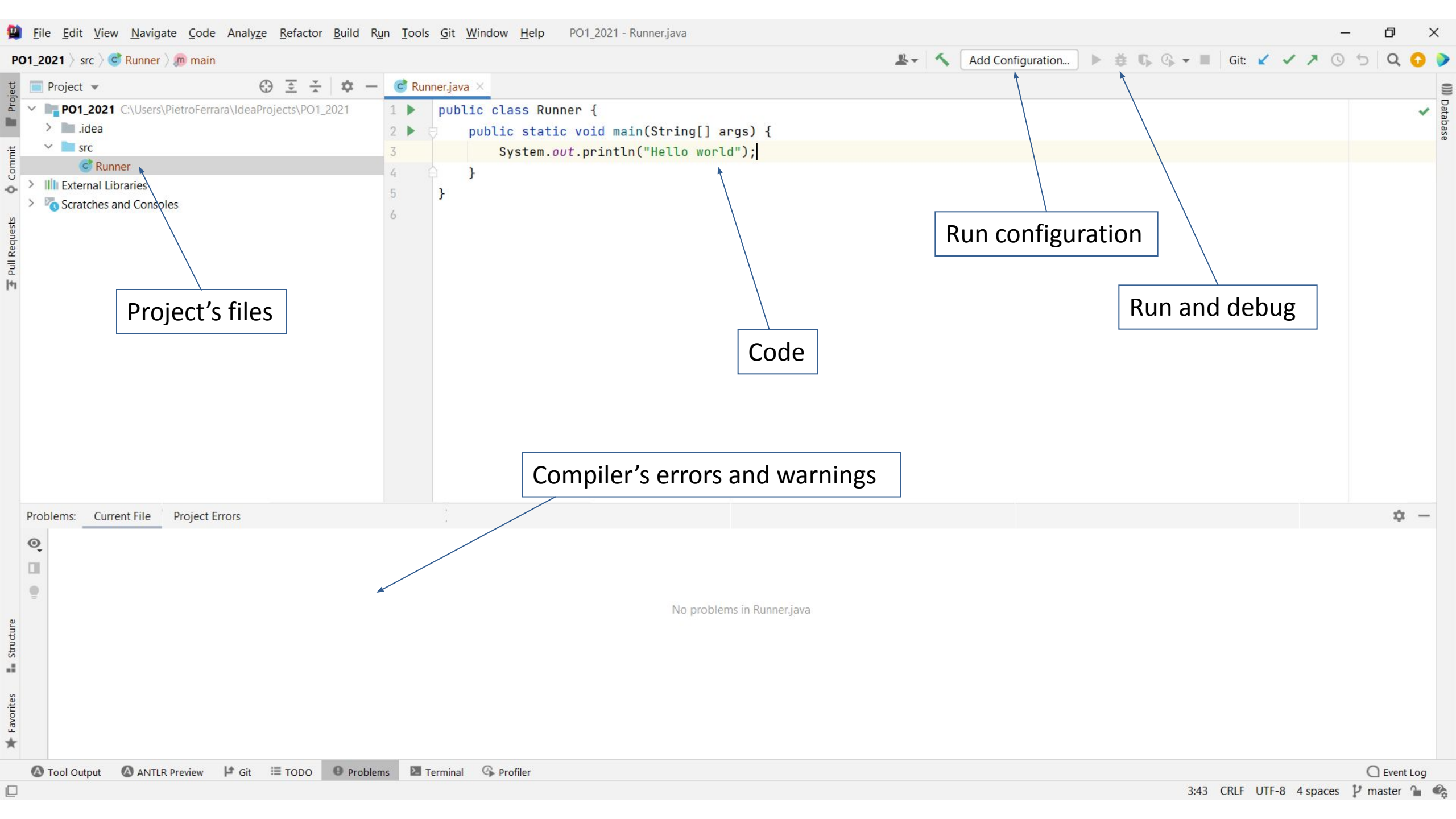
wooclap

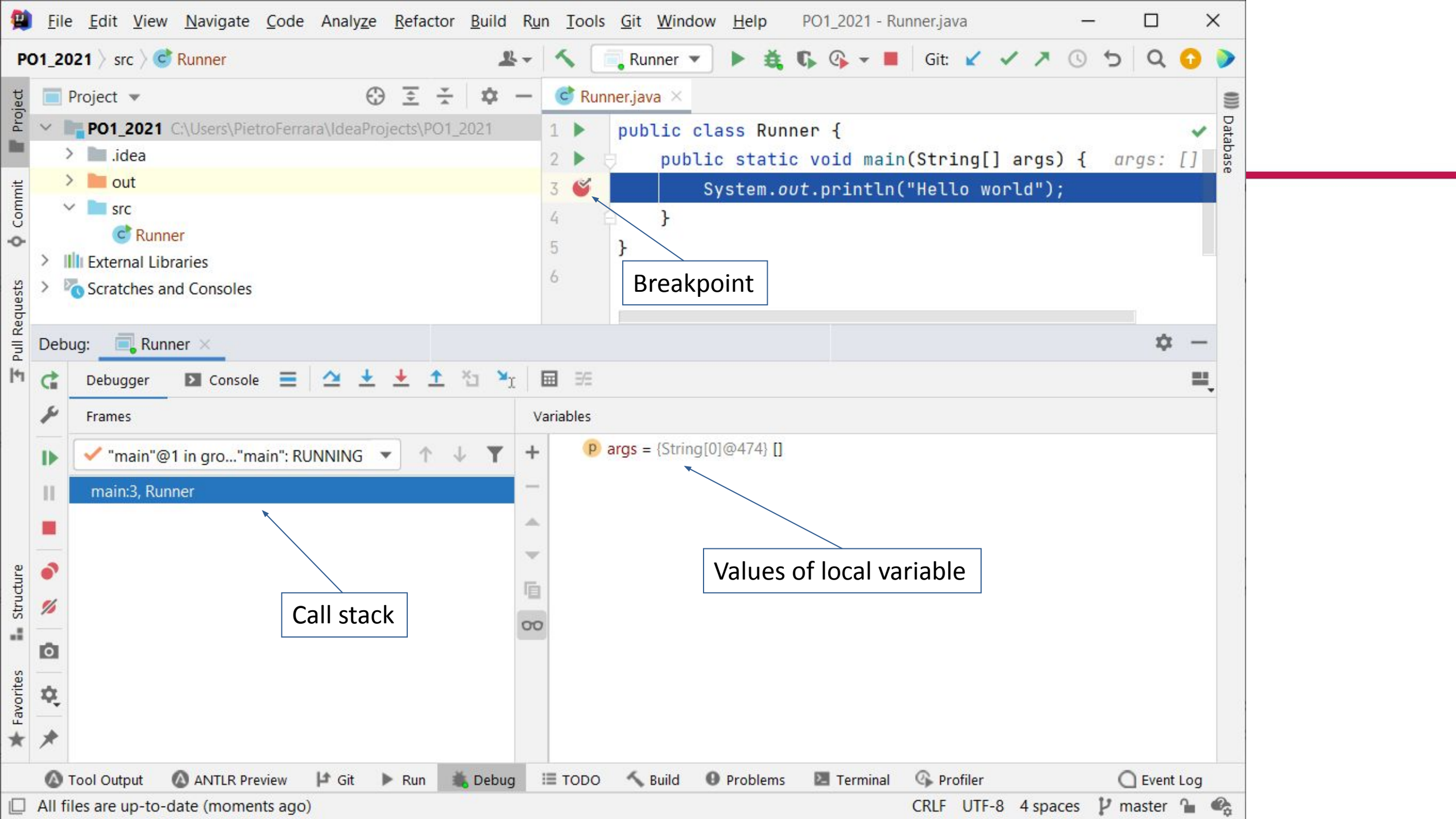
<https://app.wooclap.com/events/PO12324> (questions from 1 to 2)



Integrated Development Environment

- IDEs are a fundamental tool for developers nowadays
- There are many different IDEs
 - Eclipse, IntelliJ IDEA, NetBeans, Visual Studio [Code]
- They are (more or less) the same
- I will use IntelliJ IDEA
 - Ultimate Edition for free for students and professors
 - <https://www.jetbrains.com/community/education/#students>
- Demo
 - https://github.com/pietroferrara/PO1_2023







Ca' Foscari
University
of Venice

Make (a lot of) practice
with Java and IDEs

Well, more for your future career as a
software developer than for this course...