

Introduzione al corso

Introduzione alle Applicazioni Web

Luigi De Russis

Alberto Monge Roffarello, Juan Pablo Sáenz Moreno



Photo by [Sai Kiran Anagani](#) on [Unsplash](#)





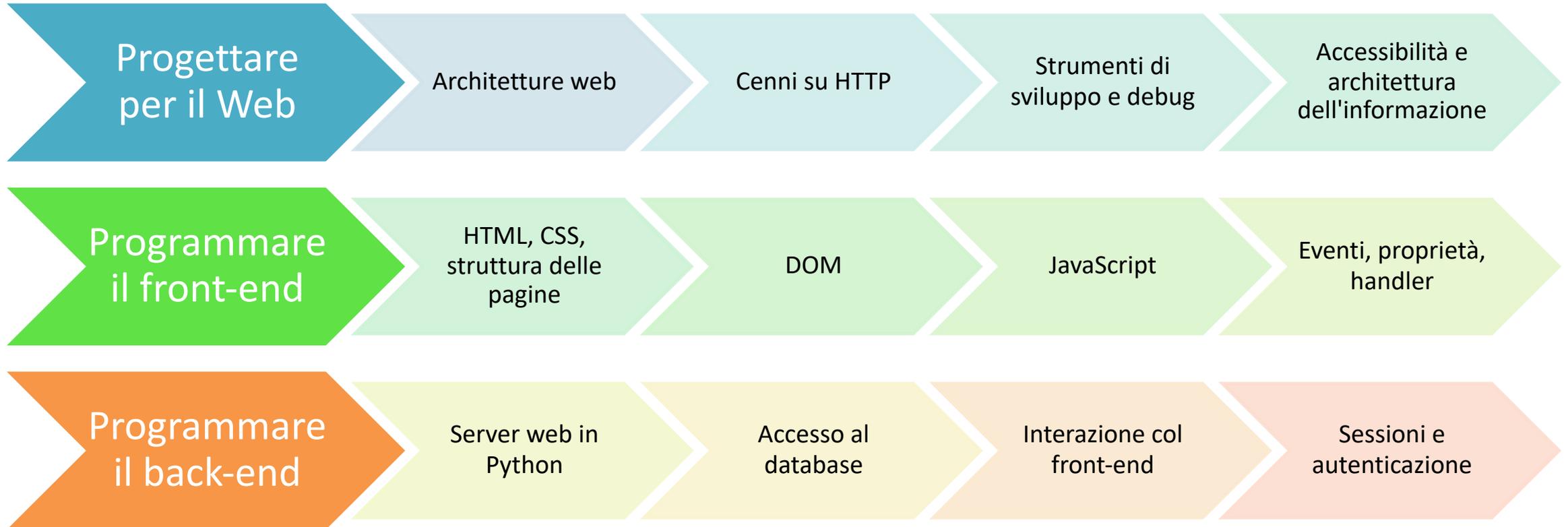
Cosa sperate di imparare in questo corso?

ASPETTATIVE?

Obiettivi

- Conoscere i concetti fondamentali relativi alle architetture web e i protocolli in uso.
- Conoscere e usare le principali tecnologie per il web (HTML, CSS, JavaScript) e saperle applicare a scenari reali.
- Possedere e utilizzare le conoscenze base di un framework in Python per creare applicazioni web.
- Creare un'applicazione web completa, di media complessità, usabile e accessibile.

Che cosa impareremo



Un'occhiata agli argomenti

1. Architetture web e HTTP
2. HTML5 e DOM
3. Progettare per il web e architettura dell'informazione
4. CSS3 e design responsive
5. Flask
6. Form
7. Sessioni
8. Interazione col database
9. Autenticazione
10. JavaScript
11. Manipolazione del DOM



Organizzazione del corso

- Lezioni
 - 3 o 4.5 ore/settimana
 - Interattive
 - Lezioni + esercizi (mix)
 - Video-registrate
- Laboratori
 - 1.5 ore/settimana
 - 2 slot
 - Dalla **prossima** settimana

	LUN	MAR	MER	GIO	VEN
08:30			Lab LABINF		
10:00			Lab LABINF		
11:30					
13:00					
14:30				Lezione 7S	
16:00	Lezione 12				
17:30	(Lezione 12)				



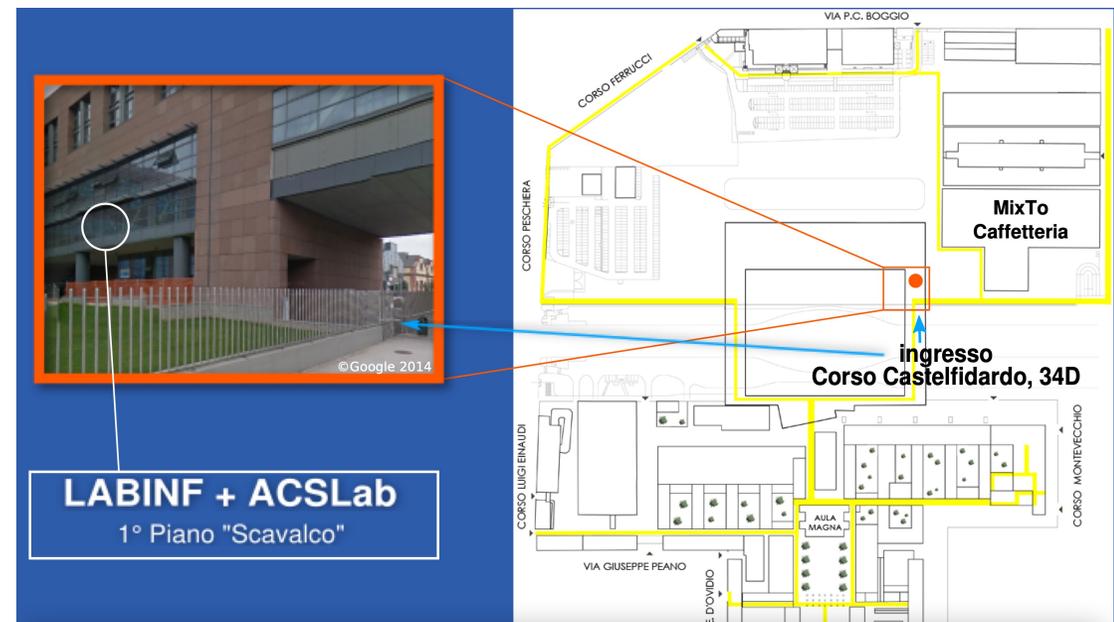
Tenete d'occhio le ~~scale~~^{aule}...
... a loro piace cambiare

Lezioni

- In presenza, (*ogni tanto*) in aule con prese ai banchi
 - Se volete o potete, portate il vostro computer per meglio seguire gli esempi/esercizi
- Video-registrate e rese disponibili dopo ogni slot in aula
 - Non in streaming

Laboratori

- Dal **12 Ottobre 2022**
 - Al LABINF (<https://www.labinf.polito.it>)
- Testi online, qualche giorno in anticipo
- Esercizi da svolgere durante le ore in laboratorio
- Soluzioni disponibili su GitHub
 - Circa una settimana dopo la fine di ogni laboratorio

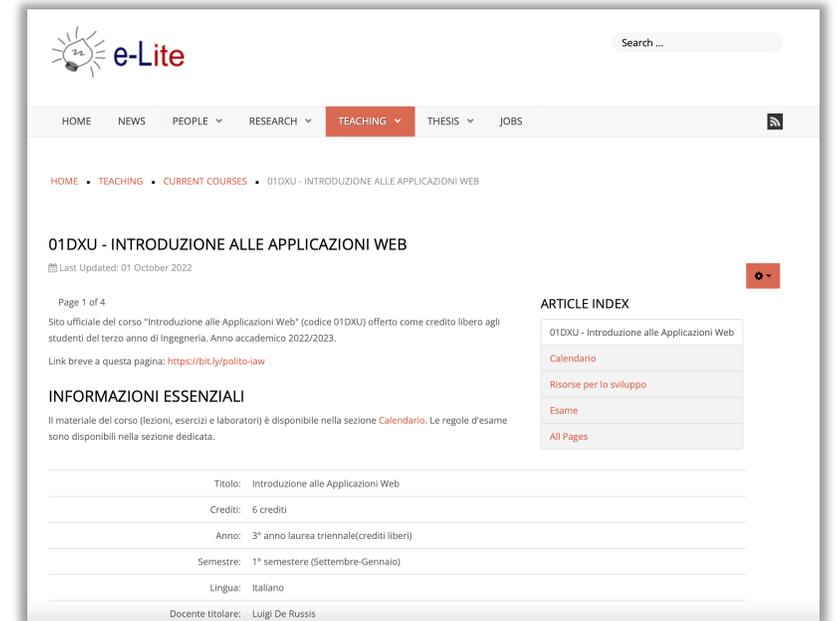


Laboratori

- Due turni
 - 8:30 – 10:00
 - 10:00 – 11:30
- Potete usare i computer del LABINF o i vostri personali
 - come preferite
- Indicate una preferenza su quale turno frequentare
 - entro il **10 Ottobre 2022** (a fine giornata)
 - <https://forms.gle/FmWVWqQh8G7wQ7Lv5>

Materiale

- Sito web del corso
 - <https://bit.ly/polito-iaw>
 - Slide, testi dei laboratori, esercizi, link, esami, ...
 - Calendario delle lezioni e dei lab
- Video-registrazioni (solo per le ore in aula)
 - YouTube https://www.youtube.com/playlist?list=PLs7DWGc_wmwTg9F9SkP6GWtJ5FV36dIxp
 - Portale della Didattica
- GitHub - <https://github.com/polito-iaw-2022>
 - Slide, testi e soluzioni dei lab, esercizi, ...



Comunicazioni



- Useremo **Telegram** per le comunicazioni rapide
 - Tra studenti, con i docenti, ecc.
- Link del gruppo: <https://t.me/+XEHOwq2D01xhYTNk>
- Annunci, promemoria e comunicazioni ufficiali avranno il tag **#news**
- Contattate pure i docenti per feedback e per fare domande
 - Usate il tag **#question**, se scrivete nel gruppo
- Le email sono un'**alternativa** per conversazioni più lunghe, lente e private
 - Sfruttate anche le *ore di ricevimento*

Ore di ricevimento

- **Perché?**

- Un'opportunità per i singoli (o gruppi) per discutere eventuali bisogni o problemi
- Per chiarire informazioni o fare domande sul corso
- Per discutere obiettivi accademici o di carriera
- Per sapere di più su certi argomenti
- ...

- **Quando?**

- **Giovedì 16:30-18:30** nel mio ufficio: [prenota tramite il mio calendario](#)
- Su richiesta, in persona (nel mio ufficio) o da remoto (su Zoom)

Esame

1. Sviluppo di un progetto

- Individuale
- A partire da un documento di specifiche
- 20 giorni di tempo

2. Discussione orale

- Individuale e obbligatoria
- Correzione “live” del progetto
- Domande sul progetto (scelte, chiarimenti, ...)
- *Quando*: il giorno dell’esame (o “da quel giorno”)

Sviluppo di un progetto

Che cosa?

- Sviluppo di un'applicazione web utilizzando
 - HTML, CSS, JavaScript
 - Python e Flask
 - SQLite
- Secondo delle specifiche
 - Pubblicate 20 giorni prima di ogni data d'esame ufficiale

Come?

- Individualmente (cioè, non in gruppo)
- Valutazione dei docenti
 - Lanciando e testando l'applicazione...
 - ... esaminando il codice...
 - ... rispetto alle specifiche rilasciate

Discussione orale

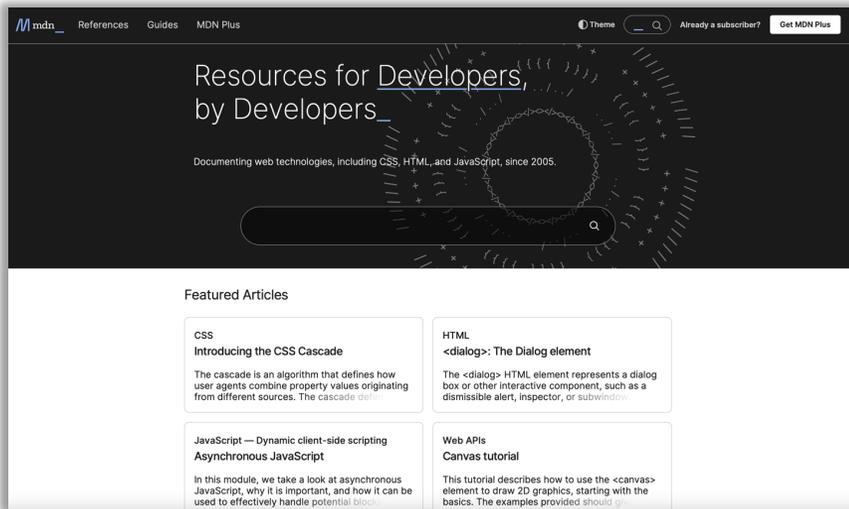
Obiettivo

- Assicurarsi che ogni studente/ssa abbia sviluppato il progetto in *autonomia* (da solo/a)
- Valutare la capacità dello studente/ssa di spiegare il *comportamento esatto* del codice

Criteri di valutazione

- Conoscenza pratica e teorica della progettazione dell'applicazione web consegnata
- Conoscenza pratica e teorica del codice dell'applicazione web
- Prontezza e chiarezza nelle risposte

Risorse (essenziali)



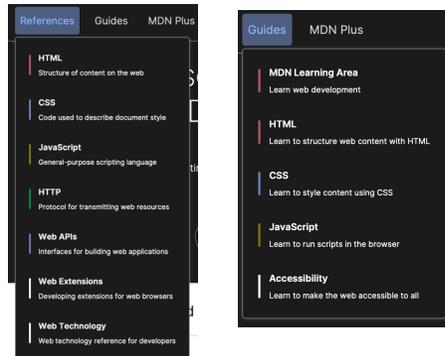
Resources for Developers, by Developers

Documenting web technologies, including CSS, HTML, and JavaScript, since 2005.

Featured Articles

- CSS**
Introducing the CSS Cascade
- HTML**
<dialog>: The Dialog element
- JavaScript** — Dynamic client-side scripting
Asynchronous JavaScript
- Web APIs**
Canvas tutorial

Mozilla Developer Network (MDN)
<https://developer.mozilla.org/>



References Guides MDN Plus

- HTML
- CSS
- JavaScript
- HTTP
- Web APIs
- Web Extensions
- Web Technology

Guides MDN Plus

- MDN Learning Area
- HTML
- CSS
- JavaScript
- Accessibility



Project Links

Donate
PyPI Releases
Source Code
Issue Tracker
Website
Twitter
Chat

Flask

web development,
one drop at a time

Welcome to Flask's documentation. Get started with [Installation](#) and then get an overview with the [Quickstart](#). There is also a more detailed [Tutorial](#) that shows how to create a small but complete application with Flask. Common patterns are described in the [Patterns for Flask](#) section. The rest of the docs describe each component of Flask in detail, with a full reference in the [API](#) section.

Flask depends on the [Jinja](#) template engine and the [Werkzeug](#) WSGI toolkit. The documentation for these libraries can be found at:

- [Jinja documentation](#)
- [Werkzeug documentation](#)

Quick search

Go

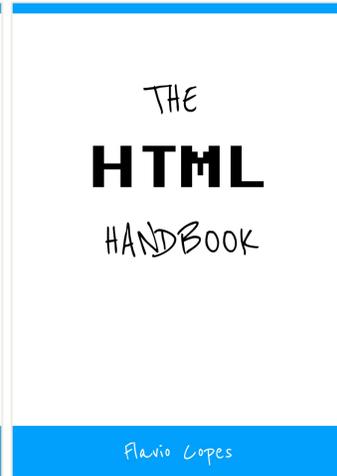
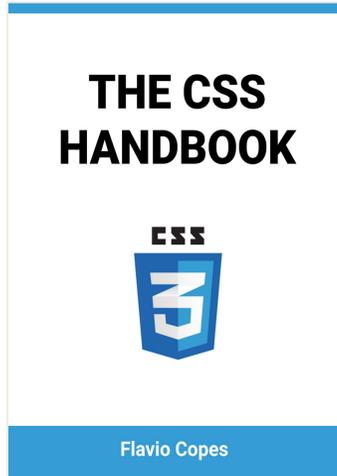
User's Guide

Flask provides configuration and conventions, with sensible defaults, to get started. This section of the documentation explains the different parts of the Flask framework and how they can be used, customized, and extended. Beyond Flask itself, look for community-maintained extensions to add even more functionality.

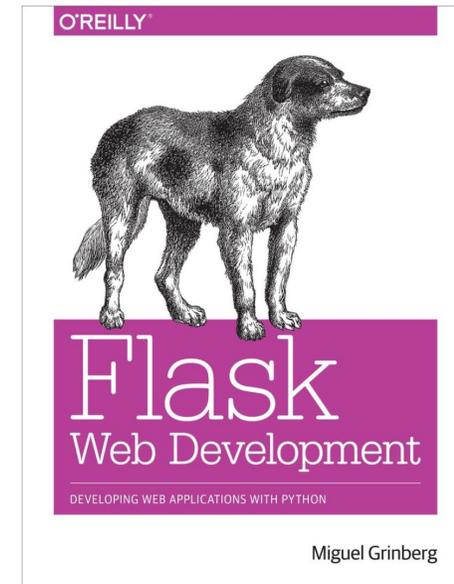
- Installation**
 - Python Version
 - Dependencies
 - Virtual environments
 - Install Flask
- Quickstart**
 - A Minimal Application
 - Debug Mode
 - HTML Escaping
 - Routing
 - Static Files

Documentazione di Flask
<https://flask.palletsprojects.com>

Risorse (libri, online e non)

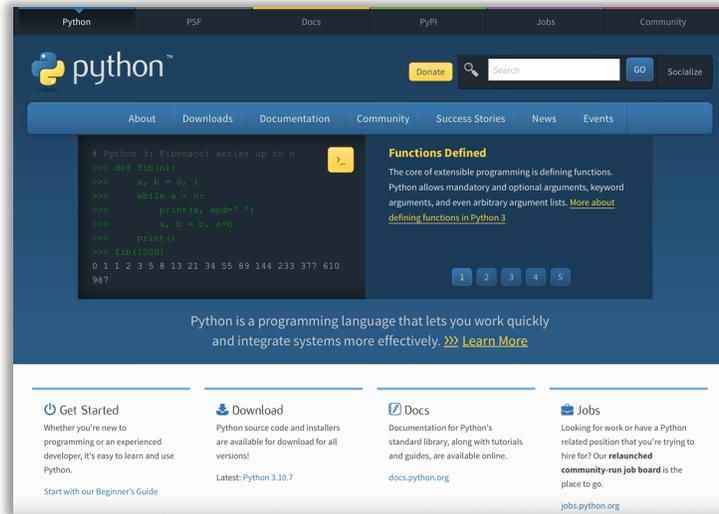


Flavio Copes Handbooks
<https://flaviocopes.com/>

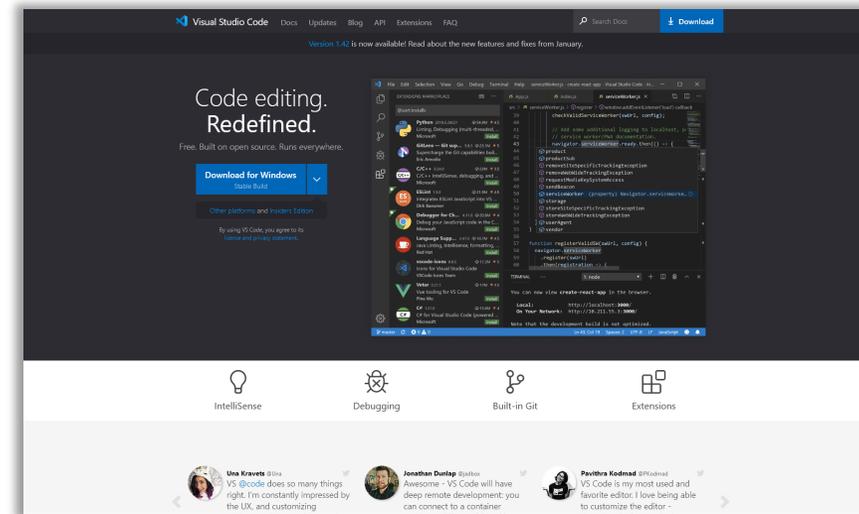


<https://www.oreilly.com/library/view/flask-web-development/9781491991725/>

Strumenti e ambienti di sviluppo



Python 3.10
<https://www.python.org>



Visual Studio Code
<https://code.visualstudio.com/>



Licenza

- These slides are distributed under a Creative Commons license “**Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)**”
- **You are free to:**
 - **Share** — copy and redistribute the material in any medium or format
 - **Adapt** — remix, transform, and build upon the material
 - The licensor cannot revoke these freedoms as long as you follow the license terms.
- **Under the following terms:**
 - **Attribution** — You must give [appropriate credit](#), provide a link to the license, and [indicate if changes were made](#). You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.
 - **NonCommercial** — You may not use the material for [commercial purposes](#).
 - **ShareAlike** — If you remix, transform, or build upon the material, you must distribute your contributions under the [same license](#) as the original.
 - **No additional restrictions** — You may not apply legal terms or [technological measures](#) that legally restrict others from doing anything the license permits.
- <https://creativecommons.org/licenses/by-nc-sa/4.0/>

