

Python OOP - Labo 7

Oefening "Static Site Generator"

- Een static site generator is een tool die een volledig statische HTML website maakt op basis van data en een reeks templates (een herbruikbaar gedeelte html dat kan gemixt worden met data)
- Het bestaat uit een script dat de taak om individuele HTML-pagina's te schrijven automatiseert. Bij elke wijziging aan de data laat je het script lopen en wordt de statische site opnieuw (en automatisch) gegenereerd. Deze wordt dan online gezet en is snel, veilig en goedkoop te hosten vergeleken met dynamische websites. Ze vormen dus een competitief alternatief voor content management systemen (CMS) zoals WordPress.
- Voorbeelden van SSGs zijn Eleventy, Hugo, Gatsby, Jekyll. **TAAK 1: DOE JE RESEARCH NAAR WAT SSG's ZIJN.**
- Je maakt zelf een heel eenvoudige SSG in Python bestaande uit een aantal pagina's. De auteur schrijft teksten en het script zet om naar een reële statische site.
- Je zorgt in je project voor een folder "pages". Daarin ga je een aantal statische pagina's toevoegen. Je schrijft deze pagina's in markdown en voegt er bovenaan zogenaamde front matter toe in yaml. **TAAK 2/3: ONDERZOEK HOE JE MARKDOWN SCHRIJFT EN HOE JE FRONT MATTER IN YAML TOEVOEGT AAN EEN BESTAND.** Je zal een Python-library gebruiken zowel voor het parsen van markdown als voor het parsen van de front matter.
- Je maakt ook een folder "templates" aan die één of meerdere html-bestanden bevat waarmee je de pagina's kan mixen. Voor het mixen zelf ga je gebruik maken van de Jinja templating engine: <https://jinja.palletsprojects.com/en/3.1.x/> en <https://github.com/pallets/jinja/>. Een overzichtelijke intro vind je hier: <https://realpython.com/primer-on-jinja-templating/> **TAAK 4: ONDERZOEK WAT JINJA IS EN DOET EN HOE JE ERMEE WERKT IN JE CODE.** Jinja = templates + tekst (markdown) + data (yaml)
- Door het runnen van je script (geef jouw SSG een eigen naam!) loop je door alle bestanden in de directories "pages" en zorg je ervoor dat de site gegenereerd wordt in een directory "docs". **TAAK 5: BEKIJK HIERVOOR DE SLIDES OVER WERKEN MET DIRECTORIES EN BESTANDEN**
- Je denkt na over de site navigatie. Elke gegenereerde pagina heeft een automatisch gegenereerd nav-element nodig waarmee naar de andere pagina's kan worden gesprongen.
- Je gebruikt git bij dit project en bestudeert Github pages. Lukt het je om bij elke push naar de git provider de nieuwe versie van je site automatisch online te laten verschijnen?
- Probeer zo verstandig mogelijk te ontwikkelen, met gebruik van functies in overzichtelijke modules. Werk bij voorkeur ook in een virtuele omgeving.
- Bonusvraag: Blogposts toevoegen aan je SSG, tags toevoegen aan de blogposts en een tagarchief-pagina toevoegen aan de site.