



Întâlnirea 15

API testing

Sfaturi generale

- Să tratați cu **seriozitate** și **profesionalism** acest nou obiectiv.
- Cei care își ating obiectivele nu sunt întotdeauna cei mai smart, dar întotdeauna vor fi cei mai **muncitori**!
- Alocați-vă timp pentru studiu. Rutina dă **consistență**. Consistența dă **exelență**.
- Să faceți tot posibilul să participați la **toate** sesiunile live.
- Să vă lăsați **comentarii** explicative în cod. Notițe pentru voi din viitor.
- Recomand să vizualizați **înregistrarea**. Să vă notați aspectele importante + întrebări pentru trainer pentru ora următoare.
- Să vă faceți **temele** și unde nu reușiți singuri, să întrebați pe **grup**. Trainerul va **răspunde** și vor beneficia și ceilalți cursanți de răspuns.
- Puteți chiar să faceți un grup doar de studenți și să vă întâlniți o dată pe săptămână să discutați temele **împreună**. Fiecare va veni cu o perspectivă nouă și în final toți vor avea de câștigat.
- În timpul orelor, să aveți **curaj** să puneți **întrebări** când ceva nu e clar.

Reguli curs

- Va exista un sheet de prezență.
- În cadrul acestuia ne vom asuma și noțiunile învățate. Nu trecem mai departe până nu își asumă toți noile concepte.
- Temele se vor adăuga în Folderul grupei, veți face fiecare folder cu numele vostru. Veți primi feedback la aceste teme.
- Temele vor fi împărțite în 2 categorii.
 - Obligatorii (se pot face doar cu noțiunile învățate la clasă)
 - Opționale (acestea vor fi mai advanced și necesită poate extra research). Acest lucru îi va motiva și pe cei care au mai mult timp și le place să se aventureze prin task-uri mai dificile.
- Vă rog să mă întrerupeți oricând aveți întrebări. Doar așa îmi pot da seama unde trebuie să mai insist cu explicații/exemple.
- Vă rog să intrați cu 3 minute mai devreme în caz că apar probleme tehnice. Astfel putem profita la maxim de cele 2 ore alocate.
- Dacă nu puteți intra, sau dacă întârziați, anunțați trainerul pe grup.

Obiective principale

Până la final TOȚI* veți avea:

- Cunoștințe solide despre bazele programării în Python
- Cunoștințe mai avansate și extrem de utile despre programarea bazată pe obiecte.
- Capacitatea să identifice elemente și să scrie test scripts cu ajutorul Selenium
- Un Proiect final de testare automată a aplicațiilor web.
 - Acesta va folosi tendințele actuale: metodologia Behavior Driven Development și Page Object Model Design pattern.
 - Vă avea capacitatea să genereze rapoarte HTML ('living documentation')
 - Veți ști de la A la Z acest framework, astfel că veți avea capacitatea să continuați să îl dezvoltați post curs acest proiect (pentru orice website doriți).
- Noțiuni de bază despre API testing. (testarea backend - ce e în spate la un website).

* toți cei care sunt activi, implicați, își fac temele, dedică timp pentru studiu individual și pun întrebări trainerului vor atinge aceste obiective.

Obiective secundare

Nu fac parte din curricula cursului LIVE dar vă punem la dispoziție materiale extra că să aveți un avantaj la interviuri. Sfatul meu e să vă focusați pe ele doar după cursul live. Să nu fiți overwhelmed de new info.

- Cunoștințe ale bazelor de date relaționale - mySQL (Curs baze de date)
- Cunoștințe teoretice despre testarea manuală - acces la o platformă mobilă
- Capacitatea de a construi un mic brand personal (Curs Portofoliu Wordpress). Trebuie să ai:
 - Website propriu prin care angajatorul să te cunoască pe tine și munca ta
 - CV european în eng / sau canva.com
 - Profil LinkedIn
 - Github public (un loc în cloud unde se pune codul scris de tine)
 - Veți primi feedback dacă ne trimiteți un email cu ele la hello@itfactory.ro

Objective Întâlnire 15

- **Recapitulare Postman - CRUD**
- **API Automation Framework - starter kit**

Metode API

Acronimul CRUD vine de la create, read, update, delete - acțiunile principale când lucrăm cu date în software development

Cele mai folosite metode API sunt:

GET - cerem date de la server - în general 200

POST - trimitem date la server - în general 201

PATCH - updatăm date prin updatarea anumitor attribute ale obiectului (ex: din user doar prenumele) - 2xx

PUT - updatăm date prin suprascrierea întregului obiect (ex: tot userul - nume, prenume, adresa) - 2xx

DELETE - ștergem date - în general 204

Recap Postman:

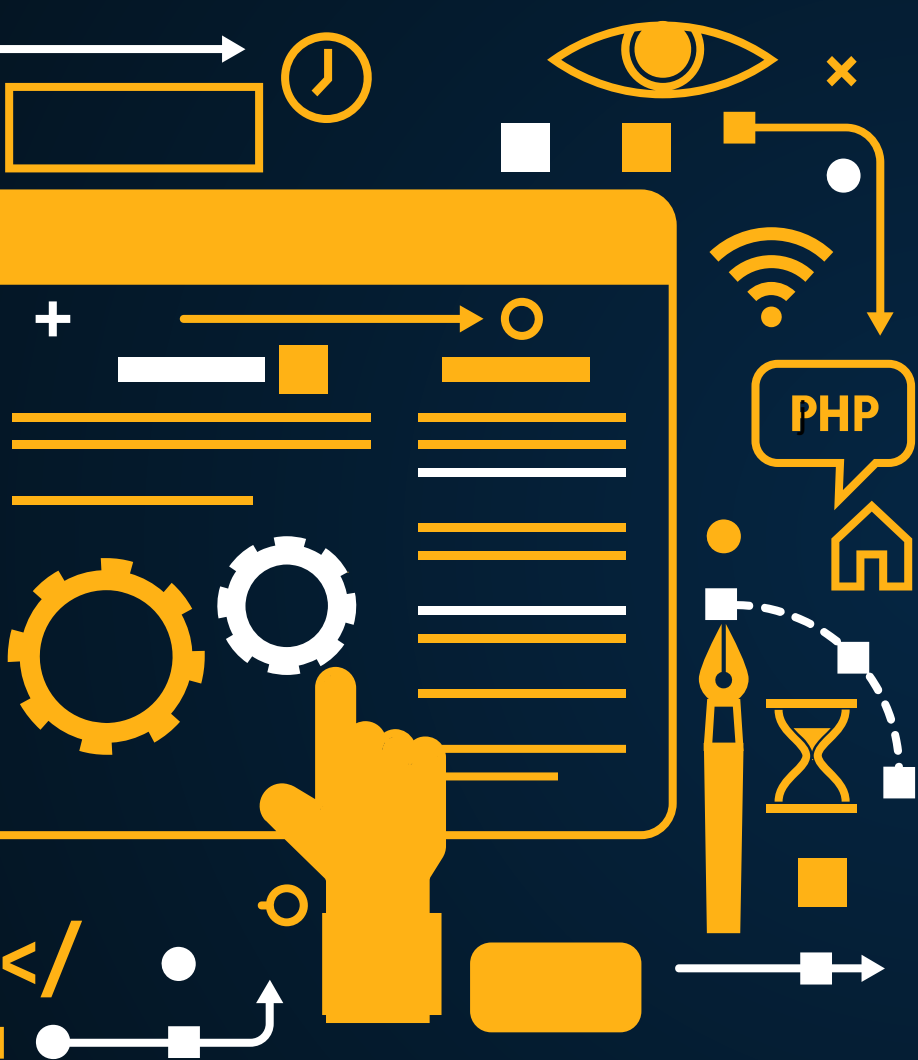
<https://github.com/vdespa/introduction-to-postman-course/blob/main/simple-books-api.md>

API Automation

- Instalăm pachetul care ne ajută să facem requesturi
- `pip install requests`
- Începem împreună testarea pentru Books

<https://github.com/vdespa/introduction-to-postman-course/blob/main/simple-books-api.md>

- Verificăm să mergă pentru toată lumea callul de status
 - Atât în postman
 - Cât și în cod
- Vă arăt eu codul dar discutăm împreună scenariile
- Vă rămâne la temă să implementați voi ulterior, urmărind video
- Nu vă dau fișierele intenționat ca să mă asigur că scrieți voi codul
 - Oportunitate să înțelegeți și mai bine, linie cu linie
- În final, îl veți urca în github că și repo de API testing



Întrebări de interviu:

- Ce librării putem folosi pentru API testing în Python?
- Ce este un token? Unde se pune, în header sau în body?
- Ce este un JSON?
- Care e diferența dintre url params și body params?

Întrebări & curiozități?