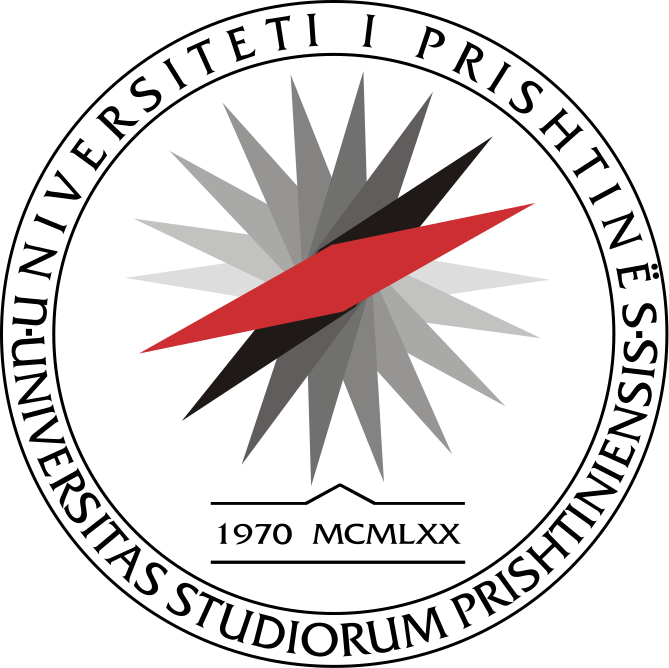
**UNIVERSITETI I PRISHTINES “HASAN PRISHTINA”**

**FAKULTETI I INXHINERISE ELEKTRIKE DHE KOMPJUTERIKE**

**DEPARTAMENTI: INXHINIERI KOMPJUTERIKE**



**TEMA: "Realizimi i nje vegle me Python per gjetjen dhe listimin e subdomeneve permes Brute force algoritmit"**

MENTORI: KANDIDATI:

Prof. AVNI REXHEPI KENAN M. PALLUSHA

# Permbajtja

#### Abstrakti

#### Abstract in English

#### Hyrja

#### Motivimi

#### Pershkrimi i problemit

#### Domenet dhe Subdomenet

#### Domenet e nivelit te larte (Top level domains)

#### Subdomenet dhe funksionaliteti i tyre

#### Gjuha programuese Python

#### Paketa PIP3 ne Python

#### Gjuha programuese HTML,

#### CSS

#### Algoritmet

#### Brute Force Algoritmi

#### Elaborimi Matematikor

#### Algoritmet e Brute Force dhe Greedy: Te përbashkëtat dhe dallimet

#### Vegla per gjetjen e subdomeneve përmes Brute Force algoritmit

#### Rezulltatet

Formati standard: A4, margjinat 2.5, teksti Times New Roman 12, rreshtimi ne te dy anet (Justify), distanca "single", etj...me afersisht 40-50 faqe.

Rradhitja:

- Abstrakti (shqip dhe anglisht),

- Hyrja (Hyrje, Motivimi, PërshkrimiIproblemit...),

- Kapitull i pergjithshem (ne rastin tend: algoritmet, vetite, perdorimi, karakteristika e tyre, etj...),

- Kapitull i pergjithshem (ne rastin tend: gjuhët/platformat, vetite, perdorimi, karakteristika e tyre, etj...),

- Aplikacioni/shembujt (detajet si dhe çkea eshte perdorur nga platforma/framework-at, rezultatet, krahasimet, etj).

- Perfundimet (shqip dhe anglisht)

- Referencat (Literatura, Lista e tabelave, lista e figurave, etj).

Në dokument inserton figura, tabela, pamje të ekranit (screen-shots), etj.

# Abstrakti

Në botën e sotme të udhëhequr nga teknologjia, shërbimet dhe shërbimet e rrjetit kanë bërë të pamundurën në fusha, nga bizneset deri te disa qeveritare. Pranishmëria e kësaj platforme që ka në web sjell edhe sfidat e globale dhe të sigurisë së tyre. Një aspekt kritik i sigurisë së njerëzve të rrjetit gjendet në identifikimin dhe numërimin e nën-domaineve, të cilat mund të jenë pikat e hyrjes potenciale për shkak të kibernetike. Procesi i gjetjes dhe listimit të nën-domaineve, i njohur si numërimi i nën-domaineve, një rol kyç ne humutimet si: Siguria dhe Vleresimi I mbrojtejes kibernetike, Testimi dhe Penetrimi (Hakeret- Etik), Menaxhimi dhe administrimi I Web-faqeve, Monitorimi dhe analiza e webfaqeve dhe e Domeneve e shume te tjera.

Vegeza eshte e e shkruar ne gjuhen programuese Python ne pjesen e integrimit te Brute force algoritmit dhe eshte e lidhur me nje Front-end te shkruar ne HTML, CSS dhe nje baze te dhenesh te shkruar ne gjuhen programuese PHP.

Mjeti i plotë konceptin e Brute Force, një teknikë e cila kërkon në mënyrë që të përcaktojë një listë emrash të mundshëm të nën-domaineve. Mjeti përdor një fjalor të gjerë të nën-domaineve. Duke përdorur multi- threading dhe programim asinkron, mjeti arrin një performancë të avancuar dhe gjetje më të shpejtë të nën-doma, duke e bërë atë të ndjehet duke u kujdesur, e domeve në shkallë të gjerë. Zhvillimi i mjetit për implementimin e librave dhe moduleve të Python, si DNS Python për zgjidhjen e DNS, Kërkesat për HTTP(S) dhe ThreadPoolExecutor procesimin paralel. Ndërfaqja e përdoruesve të mjetit është e projektuar të jetë e lehtë për t'u lidhur dhe e shfletueshme, duke u mundësuar përdoruesit të rregullojë parametrat e kërkimit, të kontrollojë konkurrencën dhe të eksportojë rezultatet për analizën e mëtejshme. Për të vlerësuar efikasitetin e mjeteve, testimin e gjerë në një grup të ndryshëm të domainesh, duke përfshirë faqet e internetit private dhe publike, domaine korporatash dhe portalet qeveritare. Metrikat e veprimeve që merren në konsideratë të procesit të zbulimit të nëndomaineve, saktësinë e rezultateve dhe kryerjen e burimeve. Përveç kësaj, një analizë krahasuese krahasuar me kundrejtëdrejtëve ekzistues për numërimin e nën-domaineve për të vlerësuar konkurrencën e mjetit në peisazhin aktual të miqve kibernetike.

Rezultatet e rezultateve se mjeti i Python shkëlqen në identifikimin e efikasitetit të nën-domaineve, duke kontribuar në përmirësimin e shërbimeve të internetit. fotografia e tij për të parë nën-domaine që mund të mos vërehet në të dhënat e rëndësisë së tij në mbrojtjen e organizatave të kundërta dhe metave potenciale. Përveç kësaj e hapur e mjetit shtesë, natyraliteti dhe përmirësimi i tyre mund të jenë nga komuniteti.

Në përfundim, ky abstrakt i të dhënave të bachelorit një kontribut në fushën e mjekësisë kibernetike duke pasur një mjet të ri dhe të fortë në Python për numërimin e nën-domaineve. Përqafimi i algoritmit Bruce Force dhe programimi në Python e bën një pasuri të sigurt për telefonin tuaj celular, duke promovuar një qasje proaktive në mjetet për mjetet e infrastrukturave të lidhura në web. Thjeshtësia, efikasiteti dhe natyra e hapur e mjetit e bën atë të ngjeshur dhe të duket për një pamje të gjerë të skenarëve të mjekëve, duke përmirësuar përmirësimin e mbrojtjes së digjitale në një teknologji që vazhdon të evoluojë.

# Abstract in English

In today's technology-driven world, network services and utilities have done the impossible in fields ranging from businesses to some government. The presence of this platform on the web also brings global and security challenges. A critical aspect of netizen security is found in identifying and enumerating sub-domains, which can be potential cyber entry points. The process of finding and listing sub-domains, known as sub-domain enumeration, a key role in areas such as: Security and Cyber Defense Evaluation, Penetration and Testing (Ethical Hackers), Website Management and Administration, Monitoring and analysis of websites and domains and many others.

Tool is written in the Python programming language in the Brute force algorithm integration part and is connected to a Front-end written in HTML, CSS and a database written in the PHP programming language.

The tool completes the concept of Brute Force, a technique which searches in order to determine a list of possible sub-domain names. The tool uses an extensive dictionary of sub-domains. By using multi-threading and asynchronous programming, the tool achieves advanced performance and faster sub-domes finding, making it feel like it is taking care of large-scale domains.

Tool development for implementing Python books and modules, such as DNS Python for DNS resolution, HTTP(S) Requests, and ThreadPoolExecutor parallel processing. The tool's user interface is designed to be easy to connect and navigable, enabling the user to adjust search parameters, check competition and export results for further analysis. To evaluate the effectiveness of the tools, extensive testing on a diverse set of domains, including private and public websites, corporate domains and government portals. Action metrics that take into account the subdomain discovery process, the accuracy of the results, and the resource performance. In addition, a comparative analysis against existing subdomain enumeration countermeasures to assess the tool's competitiveness in the current cybernetic landscape

The results show that the Python tool excels at identifying the efficiency of sub-domains, contributing to the improvement of Internet services. his picture to see sub-domains that may not be noticed in the data of his importance in protecting opposing organizations and potential flaws. In addition to this open of the additional tool, their naturalness and improvement can be from the community.

In conclusion, this bachelor data abstract a contribution to the field of cyber medicine by having a new and robust tool in Python for enumeration of sub-domains. Embracing the Bruce Force algorithm and programming in Python makes it a secure asset for your mobile phone, promoting a proactive approach to tools for web-connected infrastructure tools. The simplicity, efficiency and open nature of the tool makes it compact and visible for a broad view of clinician scenarios, improving digital protection in a technology that continues to evolve.

# Hyrja

Kjo teme e Bachelorit do te shtjelloj krijimin, dizajnimin dhe analizimin e nje vegeze te shkruar ne gjuhet programuese Python, HTML, CSS, dhe PHP ashtu qe perms nje Algoritmi I quajtur Brute force qe sherben per kerkim, gjetje dhe listim te subdomeneve apo nendomeneve ten je domeni me te madh. Alogritmi Brute force eshte nje algoritem I shume perdorur qe jep rezulltate te sigurta, rrjedhimisht ate qe e kerkojme mund ta gjejme perms implementimit te ketij Algoritmi. Si pike positive e ka kete aspect, kurse si pike me pak te favorshme e ka eficiencen gjate kerkimit. Brute Force Algoritmi perdor cdo mundesi te mundshme per te arritur ne rezultatin e kerkuar. Nuk ka ndonje selektim, grupim, apo logjike me specifike per t’I anashkaluar disa mundesi apo opcione ashtu qe te arrijme deri te rezulltati me shpejte pa I kaluar te gjitha mundesite. Kjo mbase e bene algoritmin te perdorshem po jo te favorshem ne aspect te vlersimit efficient te kohes dhe kalkulimit kompjuterik. Kodi I algoritmit eshte I shkruar ne gjuhen programuese Python dhe eshte i lidhur me nje Front-end te shkruar ne gjuhen programuese HTML dhe CSS. Poashtu per listimin e e subdomeneve ten je domeni te caktuar eshte e implementuar dhe nje backend apo database e shkruar ne gjuhen programuese PHP.

Listimi i subdomeneve ten je domeni perkates mund te sherbej ne disa sfera te ndryshme ne shkencat kompjuterike dhe ne siguri kibernetike. Listimet e tilla mund te shpien ne rezulltate shume te vyeshme sa I perket performances se Domenit apo Webfaqes, sigurise kibernetike apo edhe gjurmimit te te dhenave.

# Motivimi

Qellimi dhe motivimi për këtë teme, buron nga rëndësia dhe ndikimi në rritje i listimit të subdomaineve në fusha të ndryshme, duke përfshirë sigurinë kibernetike, zhvillimin e uebit dhe menaxhimin e rrjetit. Regjistrimi i subdomaineve luan një rol kryesor në identifikimin e dobësive të mundshme, optimizimin e infrastrukturës së uebit dhe rritjen e sigurisë së përgjithshme dixhitale. Duke kryer një studim të thelluar të teknikave të numërimit të nënfushave, tema synonon të kontriboj me njohuri të vlefshme që mund të përfitojnë organizatat dhe individët në mënyra të shumta:

**Rritja e sigurisë kibernetike:** Në një botë ku kërcënimet kibernetike po evoluojnë vazhdimisht, të kuptuarit e teknikave të numërimit të nëndomaineve mund t'i ndihmojë organizatat të identifikojnë pikat e mundshme hyrëse për sulmet kibernetike. Hulumtimi I tille mund të kontribuojë në përmirësimin e pozicionit të sigurisë së sistemeve dhe faqeve të internetit duke mundësuar zbulimin në kohë dhe zbutjen e dobësive të sigurisë.

**Menaxhimi efikas i Infrastrukturës së Uebit:** Menaxhimi efektiv i infrastrukturës së uebit është thelbësor për ofrimin e përvojave pa probleme në internet. Duke studiuar numërimin e nëndomaineve, ju mund të ofroni strategji për të përmirësuar dhe optimizuar burimet e uebit, duke çuar në përvoja më të mira të përdoruesit dhe shpërndarjen e burimeve. Në thelb, Tema ka potencialin për të lidhur njohuritë teorike me aplikime praktike, duke adresuar sfidat e botës reale dhe duke ofruar zgjidhje që mund të ndikojnë pozitivisht në industri të ndryshme.

**Fuqizimi i Hakimit Etik dhe Testimi i Penetrimit:** Hakerat etikë dhe testuesit e depërtimit (pentrimit) mbështeten në informacionin e saktë të nëndomainit për të simuluar sulmet e mundshme dhe për të vlerësuar dobësitë qe potencialisht mund te krijohen. Materiali hulutues I kesoj teme mund t'i pajis profesionistet me mjete dhe teknika më të mira për identifikimin e pikave të mundshme të hyrjes dhe ndihmën në masat proaktive të sigurisë.

**Mbështetja e marketingut dixhital dhe markimit:** Nëndomainët përdoren shpesh për fushata marketingu të synuara dhe iniciativat e markës. Hulumtimi juaj mund të ofrojë njohuri se si organizatat mund të përdorin në mënyrë strategjike nënfushat për të krijuar përvoja të dallueshme në internet që angazhojnë përdoruesit në mënyrë efektive.

**Përparimi i praktikave të zhvillimit të uebit:** Zhvilluesit mund të përfitojnë nga një kuptim i thellë i teknikave të numërimit të nëndomaineve për të siguruar aplikacione të fuqishme ueb. Teza juaj mund të hedhë dritë mbi rreziqet e mundshme që lidhen me nëndomenet dhe të ofrojë udhëzime për praktikat e sigurta të zhvillimit të uebit.

**Aktivizimi i analizës gjithëpërfshirëse të rrjetit:** Për administratorët dhe analistët e rrjetit, njohja e nëndimeneve të lidhura me domenin e një organizate mund të çojë në një analizë më të plotë të rrjetit dhe në përmirësimin e vendimmarrjes.

**Kontributi akademik**: Duke u thelluar në teknikat e numërimit të nënfushave, teza juaj kontribuon në bazën e njohurive të komunitetit akademik. Ai i shton të kuptuarit kolektiv të sigurisë kibernetike, zhvillimit të uebit dhe rrjetëzimit, duke ofruar një referencë për përpjekjet e ardhshme kërkimore.

# Pershkrimi i Problemit

Domenet luajnë rol te rëndësishëm ne strukturimin e uebfaqes prandaj, trajtimi i tyre eshte i nevojshëm per te arritur nje casje me te mire ne disa aspekte sikur siguria kibernetike dhe permisesimi i praktikes se përdorimit te tyre, te cilat i kemi cekur dhe me larte.

# Domenet dhe Subdomenet (nendomenet)

Domenat dhe nën-domenat janë koncepte të emërtimit në strukturën e sistemit dhe adresimit të internetit, të njohur si Sistemi i Emërimit të Domaineve (Domain Name Sytem - DNS). Ata marrin një rol kritik në identifikimin dhe gjetjen e burimeve në internet. Le të eksplorojmë se çfarë domenat dhe nën-domenat dhe si duhet:

Domenat: Një domen është një emër i lexueshëm nga njerëzit që përkon me një adresë IP ose një grup adresash IP. Përdoret për faqet e internetit, shërbimet dhe burimet e tjera në internet. Domenat janë hierarkike dhe organizohen nga e djathta në të majtë, të ndara (pikat). Çdo pjesë e domenit quhet një etiketë. Për shembull, në domenin "[www.shembull.com](http://www.shembull.com)", "com" është domeni i nivelit të lartë (TLD), "shembull" është domeni i nivelit të dytë (SLD). ), dhe "www" është një nën-domen. Domenat duhet te sigurohet qe emri i domenit te jete unik, pra te mos jete i përdorur nga asnjë përdorues tjeter.

Nën-domenat: Nën-domenat janë nën-ndarje të një domeni më të madh. Përdoren për të krijuar seksione ose degë shtesë brenda një domeni. Nën-domenat krijohen duke shtuar një etiketë në fund të domenit kryesor, të ndara me një pikë. Për shembull, në domenin "blog.shembull.com", "blog" është një nën-domen dhe "shembull.com". Në thelb, dhe nën-domenet kanë një mënyrë të strukturuar dhe intuitive për të lëvizur në internet, duke u bërë më të lehtë për të gjetur përdoruesit dhe që janë në burime në internet. Ata në organizim, markë dhe funksionimin e faqeve të internetit dhe shërbimeve në linjë.

Nën-domenet duhet të përdoret shpesh për të dhe kategoritë e përmbajtjes ose shërbimeve brenda një faqe interneti. Ata mund të punojnë drejt serverëve të ndryshëm, direktivë ose të veçanta, ndërsa janë ende pjesë e të njëjtit domen. Nën-domenat të shumëanshëm dhe të mundshëm janë për qëllime të ndryshme.

Domenet mund te jene te këtyre shembujve:

Struktura Organizative Kompanitë mund të përdoren nën-domena për departamentet ose divizionet e ndryshme. Për shembull, "shitje.shembull.com" dhe "mbështetje.shembull.com".

Shërbime: Nën-domenat mund të jenë për të hyrë në shërbimin e "email.shembull.com" për email ose "dyqan.shembull.com" për një dyqan në internet.

Gjuhë ose Rajon - Faqet e internetit të përdorin nën-domena për të dhënat e rajonit në gjuhë të ndryshme ose të ndryshme të përcaktuara, si "sq.shembull.com" për përmbajtjen në shqip.

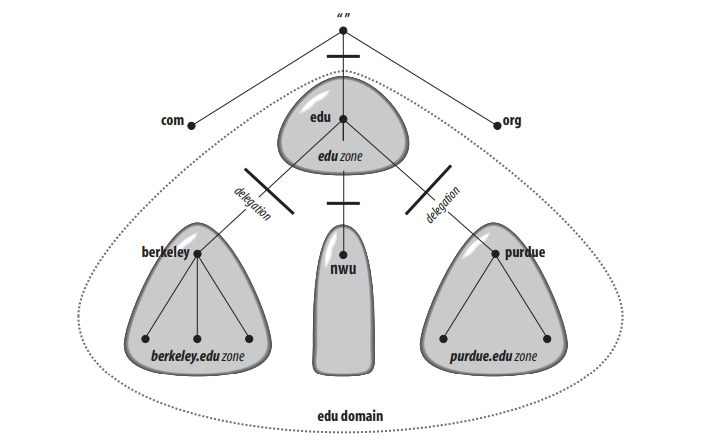
Kampanja e Marketingut- Nën-domenat mund të përdoren për fushata marketingu ose promovimi, si "verë.shitje.shembull.com".

Domain Name Server (DNS) - Sistemi i emrave të domenit është në thelb një bazë të dhënash e informacionit të hostit. Baza e të dhënave e shpërndarë e DNS indeksohet sipas emrave të domeneve. Çdo emër domain është në thelb vetëm një shteg në një pemë të madhe të përmbysur, e quajtur hapësira e emrave të domenit. Të struktura hierarkike e pemës, e paraqitur në figurën 1, është e ngjashme me strukturën e Sistemi i skedarëve Unix. Pema ka një rrënjë të vetme në krye.\* Në sistemin e skedarëve Unix, kjo është quhet adresa (direktoriumi) rrënjësore dhe përfaqësohet nga një prerje (/). DNS thjesht e quan atë "the rrënjë.” Ashtu si një sistem skedari, pema e DNS mund të degëzojë çdo numër mënyrash në çdo ndërthurje pika e veprimit, ose nyja. Thellësia e pemës është e kufizuar në 127 nivele (kufi që nuk jeni gjasa për të arritur).



FIG.1 Struktura e DNS namespace

Shembull: Një zonë përmban të gjithë emrat e domenit që përmban domeni me të njëjtin emër domeni, me përjashtim të emrave të domeneve në nënfushat e deleguara. Për shembull, niveli i lartë domeni ca (për Kanadanë) ka nënfusha të quajtur ab.ca, on.ca dhe qc.ca, për provincat Alberta, Ontario dhe Quebec. Autoriteti për ab.ca, on.ca dhe qc.ca domenet mund t'u delegohen serverëve të emrave në çdo krahinë. Domeni ca përmban të gjitha të dhënat në ca plus të gjitha të dhënat në ab.ca, on.ca dhe qc.ca. Megjithatë, zona rreth përmban vetëm të dhënat në ca (shih Figurën 2-10), e cila ndoshta është kryesisht tregues nënfushat e deleguara. ab.ca, on.ca dhe qc.ca janë zona të ndara nga ca zonë.



Ndarja e Domenit te . EDU ne subdomenet e tij.

Llojet e serverëve të emrave

Specifikimet e DNS përcaktojnë dy lloje të serverëve të emrave: Primary Master dhe Secondary Master.

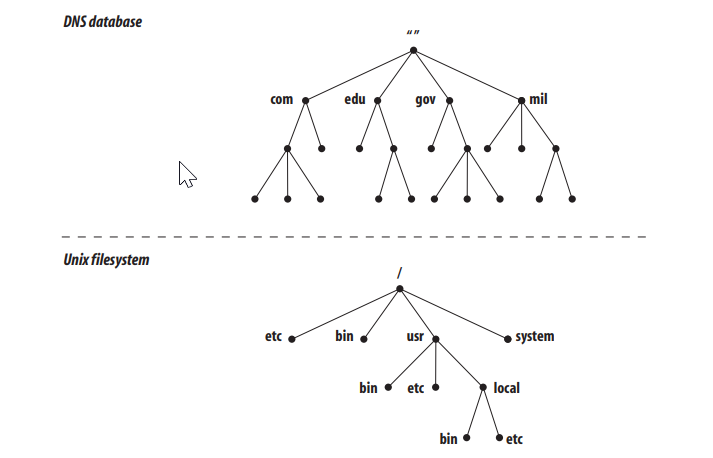


Fig.3. Dallimi ne mes te Domain Name Server dhe Unix Sistemit.

Çdo domain ka një emër unik ne baze te cilit identifikohet, si çdo directory. Emri i domenit të një domeni identifikon pozicionin e tij në bazën e të dhënave, ashtu si emri absolut i rrugës së një directory specifikon vendin e tij në sistemin e filesystem. Në DNS, emri i domenit është sekuenca e etiketave nga nyja në rrënjën e domenit deri te rrënja e të gjithë pemës, me pika (.) duke ndarë etiketat.

Në filesistemin e Unix, emri absolut i rrugës së një drejtorie është lista e emrave të afërm të lexuar nga rrënja në fletë (drejtimi i kundërt nga DNS, si treguar në figurën 3), duke përdorur një të pjerrët për të ndarë emrat.

## Domenet e nivelit të lartë (Top Level Domain )

Domenet origjinale të nivelit të lartë ndanë organizatën e hapësirës së emrave të domenit të Internetit

aleat në shtatë fusha:

***.com, .edu, .gov, .mil, .org, .int, .net***

Organizatat tregtare:

Hewlett-Packard (hp.com),

Sun Microsystems (sun.com)

IBM (ibm.com).

Organizatat arsimore:

Universiteti i Prishtines “Hasan Prishtina” (uni-pr.edu),

Oxford University (oxford.edu).

Organizatat qeveritare, si NASA (nasa.gov) dhe National Science,

SHBA - state.gov/

Pse ta përdorim Domain Name Server?

* Perdorshmeri e gjere ne mbare boten virtuale.
* Nese përdorim TCP/IP – based internet
* Nese përdorim Local Area Network apo Site Network qe janë te kyqyr ne WAN apo MAN.

# Subdomenet dhe funksionaliteti i tyre

Nëndomenet janë një metode dhe praktike e shkelqyshme për të organizuar dhe strukturuar përmbajtjen, shërbimet ose funksionalitetin e një faqe interneti brenda hierarkisë më të madhe të emrave të domenit (DNS). Ato shërbejnë për qëllime të ndryshme dhe ofrojnë përfitime që mund të përmirësojnë përdorshmërinë, menaxhimin dhe funksionalitetin e një faqe interneti. Këtu janë disa arsye për të përdorur nëndomain dhe funksionet e tyre kryesore:

**Organizimi dhe hierarkia:** Nëndomenet ofrojnë një mënyrë të qartë dhe logjike për të organizuar seksione ose aspekte të ndryshme të një faqe interneti. Për shembull, një blog mund të strehohet në një nëndomain "blog.example.com", ndërsa një dyqan i tregtisë elektronike mund të jetë në një nëndomain "shop.example.com". Ky organizim hierarkik mund ta bëjë më të lehtë për përdoruesit të gjejnë përmbajtje specifike.

**Ndarja e përmbajtjes:** Nëndomenet ju lejojnë të mbani lloje të ndryshme të përmbajtjes të ndara nga njëri-tjetri. Kjo mund të jetë e dobishme për të shmangur konfliktet midis pjesëve të ndryshme të faqes suaj të internetit ose për izolimin e disa shërbimeve për arsye sigurie.

**Përmbajtje shumëgjuhëshe ose rajonale:** Nëse faqja juaj e internetit u përshtatet gjuhëve ose rajoneve të ndryshme, mund të përdorni nënfushat për të pritur përmbajtje specifike për secilën gjuhë ose rajon. Për shembull, "en.example.com" mund të jetë për përmbajtje në anglisht, ndërsa "es.example.com" mund të jetë për përmbajtje spanjolle.

**Segmentimi i funksionalitetit:** Nëndomenet mund të përdoren për të ndarë funksione të ndryshme të një faqe interneti, të tilla si API-të e pritjes ("api.example.com"), asetet statike ("assets.example.com") ose llogaritë e përdoruesve ("account.example". com").

**Testimi dhe zhvillimi:** Nëndomenet mund të përdoren për testimin e veçorive, modeleve ose zhvillimeve të reja pa ndikuar në faqen kryesore të internetit. Kjo lejon një mjedis të veçantë për të rregulluar ndryshimet përpara se t'i vendosë ato në domenin kryesor.

**Optimizimi i performancës:** Duke shpërndarë përmbajtje nëpër nëndomanë të ndryshëm, mund të përfitoni nga ngarkimi paralel, i cili mund të përmirësojë performancën e faqes në internet. Për shembull, imazhet mund të strehohen në një nëndomain të dedikuar për përmbajtjen statike, duke çuar në kohë më të shpejta të ngarkimit.

**Shërbimet e palëve të treta:** Nëndomenet mund të përdoren për të integruar shërbime ose platforma të palëve të treta. Për shembull, një nëndomain "login.example.com" mund të trajtojë vërtetimin përmes një shërbimi të palës së tretë.

# Gjuha Programuese Python

Python është një gjuhë programimi e nivelit të lartë, e gjithanshme dhe e përdorur gjerësisht e njohur për thjeshtësinë dhe lexueshmërinë e saj. Karakteristikat dhe karakteristikat kryesore:

**Lexueshmëria:** Sintaksa e Python është krijuar për të qenë e qartë dhe e lehtë për t'u lexuar, gjë që e bën atë një zgjedhje të shkëlqyer si për fillestarët ashtu edhe për programuesit me përvojë.

**Shkathtësia:** Python mund të përdoret për lloje të ndryshme programimi, duke përfshirë zhvillimin e uebit, analizën e të dhënave, llogaritjen shkencore, mësimin e makinerive, inteligjencën artificiale, automatizimin dhe më shumë.

**Librari te gjerë:** Python vjen me një koleksion të pasur modulesh dhe Librarish të integruara, gjë që redukton ndjeshëm nevojën për të shkruar kodin nga e para për detyrat e zakonshme.

**Cross-Platform:** Python është i disponueshëm në platforma të ndryshme, duke përfshirë Windows, macOS dhe shpërndarje të ndryshme Linux.

## Paketa PIP3 ne Python 3

Menaxhimi i paketave: pip3 përdoret për të menaxhuar paketat në Python, duke e bërë të lehtë instalimin dhe përditësimin e Librarive, mjeteve dhe kornizave të ndryshme.

Versioni Python: pip3 është krijuar posaçërisht për të punuar me Python 3. Nëse jeni duke përdorur Python 2, mjeti ekuivalent quhet pip.

Komanda ne Python per qëllime te ndryshme PIP3:

**pip3 install numpy**

**pip3 install package\_name**: Instalon një paketë.

**pip3 uninstall package\_name:** Çinstalon një paketë.

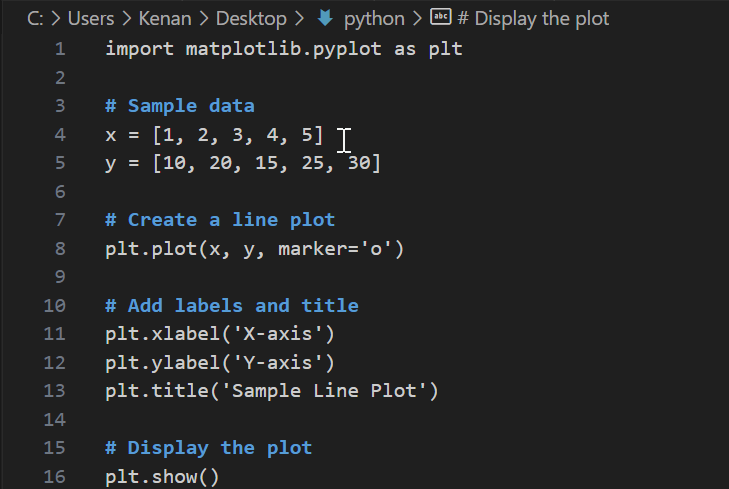
**Pip3 freeze:** Liston paketat e instaluara së bashku me versionet e tyre.

**pip3 search query:** Kërkon për paketa në PyPI.

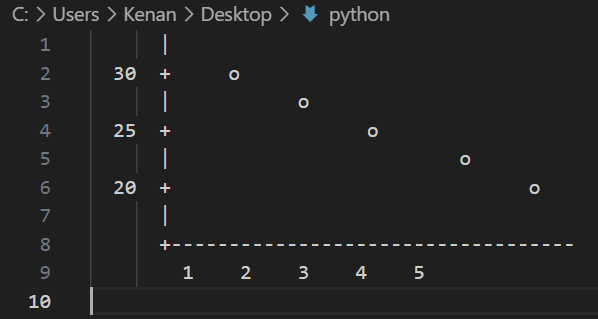
**pip3 show package\_name**: Ofron informacion për një paketë specifike të instaluar.

Shembull: Perdorimi i librarisë PIP3 me paketen **matplotlib**

**Se pari e instalojmë paketën duke përdorur komandën pip3 instal matplotlib, dhe pastaj e shkruajmë kodin:**



Kodi do te ekzekutohet si ne vijim:



# Gjuha Programuese HTML dhe CSS:

HTML është gjuha standarde e shënjimit e përdorur për të krijuar dhe strukturuar përmbajtjen në ueb. Ai siguron bazën për ndërtimin e faqeve në internet dhe përcakton strukturën dhe paraqitjen e përmbajtjes. HTML përdor etiketat për të përcaktuar lloje të ndryshme elementesh brenda një faqe interneti, të tilla si titujt, paragrafët, listat, imazhet, lidhjet, formularët dhe më shumë.

Pikat kryesore rreth HTML:

**Struktura** - HTML organizon përmbajtjen duke përdorur një strukturë hierarkike të elementeve, duke përfshirë hapjen dhe mbylljen e etiketave.

**Elementet** - Elementet HTML përbëhen nga etiketa dhe përmbajtje. Etiketat përcaktojnë llojin e elementit, dhe përmbajtja është të dhënat brenda elementit.

**Atributet** - Elementet mund të kenë atribute që ofrojnë informacion shtesë rreth elementit, si p.sh. përcaktimi i burimeve të imazhit ose përcaktimi i destinacioneve të lidhjeve.

**HTML semantike** - HTML5 prezantoi elemente semantike (p.sh., `<header>`, `<nav>`, `<main>`, `<footer>`) që tregojnë kuptimin dhe rolin e përmbajtjes që ata përmbajnë.

**Hyperlinks** - HTML mundëson krijimin e hiperlidhjeve për të lidhur faqet e internetit dhe burimet së bashku.

**Format** - Format HTML lejojnë hyrjen e përdoruesit dhe dërgimin e të dhënave në serverë. Elementet e formularit përfshijnë fushat e tekstit, butonat e radios, kutitë e kontrollit dhe butonat.

## CSS (Cascading Style Sheets)

CSS është një gjuhë e fletës së stilit që përdoret për të kontrolluar pamjen vizuale dhe paraqitjen e elementeve HTML në një faqe interneti. Ai ndan përmbajtjen (HTML) nga prezantimi (CSS), duke i lejuar zhvilluesit të aplikojnë stile të qëndrueshme nëpër faqe dhe pajisje të shumta ueb. Së bashku, HTML dhe CSS formojnë shtyllën kurrizore të zhvillimit të uebit, me HTML që përcakton strukturën dhe përmbajtjen e faqeve të internetit dhe CSS që kontrollon prezantimin dhe paraqitjen e tyre vizuale. Ata punojnë së bashku për të krijuar faqe interneti tërheqëse dhe të dizajnuara mirë.Nder karakteristikat kryesore te CSS janë te renditura poshtë:

**Selektoret apo Zgjeruesit -** Përzgjedhësit CSS synojnë elementet HTML për të aplikuar stile. Përzgjedhësit mund të synojnë elementë sipas emrit të etiketës, klasës, ID-së ose atributeve të tjera.

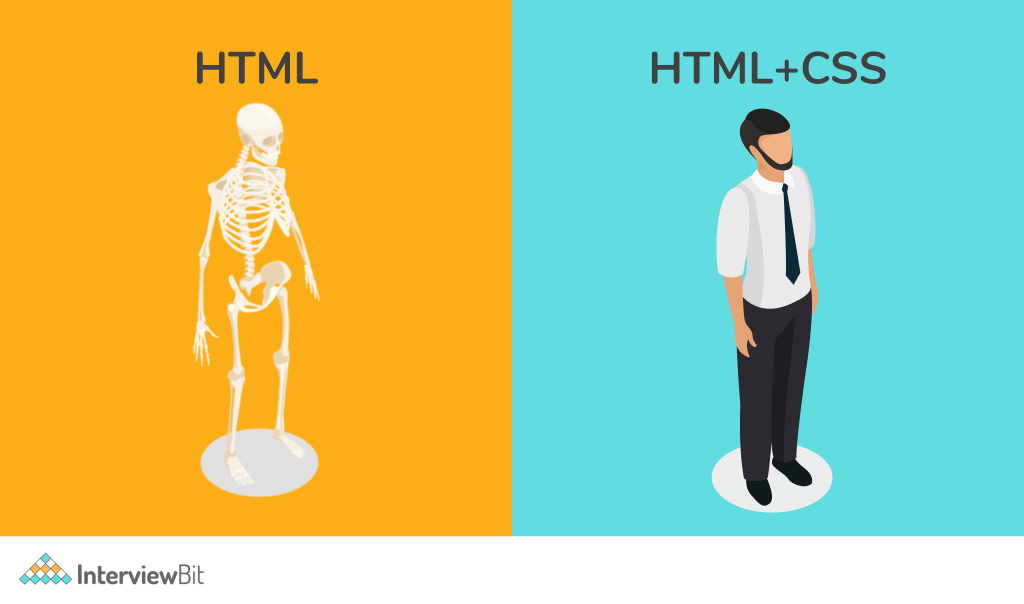
**Vetitë (Properties) dhe vlerat** - Vetitë CSS përcaktojnë aspektet e një elementi që do të stilohet, si ngjyra, madhësia e shkronjave, diferenca, mbushja dhe më shumë. Vlerat vendosin stilin specifik për secilën pronë.

**Cascading** - Natyra "kaskaduese" e CSS i referohet mënyrës sesi stilet aplikohen në një hierarki, me stile nga burime të ndryshme (përdorues, autor, parazgjedhje) që ndërveprojnë bazuar në specifikat dhe trashëgiminë.

**Zgjedhuesit dhe trashëgimia** - Vetitë CSS mund të trashëgohen nga elementët prindër tek fëmijët e tyre. Disa prona trashëgohen si parazgjedhje, ndërsa të tjerat jo.

**Dizajni i përgjegjshëm** - CSS i lejon zhvilluesit të krijojnë paraqitje të përgjegjshme që përshtaten me madhësi dhe pajisje të ndryshme të ekranit, duke përmirësuar përvojën e përdoruesit.

**Kërkesat e medias** - CSS3 prezantoi pyetjet e medias, duke u mundësuar zhvilluesve të aplikojnë stile bazuar në kushte specifike, të tilla si madhësia e ekranit, orientimi ose aftësitë e pajisjes.



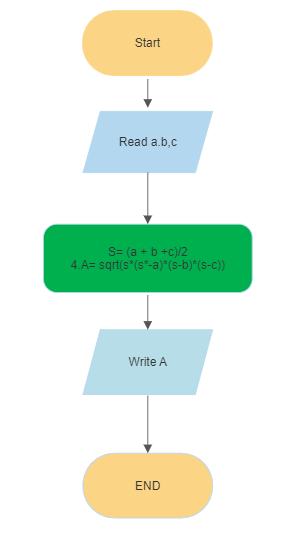
# Algoritmet

Kur shkruajmë një program kompjuterik, në përgjithësi po zbatojmë një metodë që është shpikur më parë për të zgjidhur një problem. Kjo metodë është shpesh pavarësisht nga gjuha e caktuar e programimit që përdoret - ka të ngjarë të jetë po aq i përshtatshëm për shumë kompjuterë dhe shumë gjuhë programimi. Është metodë, në vend të vetë programit kompjuterik, që specifikon hapat që ne mundemi marrë për të zgjidhur problemin. Termi algoritëm përdoret në shkencën kompjuterike për të përshkruar një metodë e fundme, përcaktuese dhe efektive e zgjidhjes së problemeve, e përshtatshme për zbatim si një program kompjuterik. Algoritmet janë lëndë e shkencës kompjuterike: ato janë objektet qendrore të studimit në terren.Ne mund të përcaktojmë një algoritëm duke përshkruar një procedurë për zgjidhjen e një problemi në a gjuhë natyrore, ose duke shkruar një program kompjuterik që zbaton procedurën.

Termi "algoritëm" rrjedh nga emri i matematikanit dhe studiuesit persian Muhamed ibn Musa al-Khwarizmi, i cili jetoi gjatë shekullit të 9-të. Al-Khwarizmi dha një kontribut të rëndësishëm në matematikë dhe prezantoi konceptin e zgjidhjes së ekuacioneve lineare dhe kuadratike, të cilat hodhën bazat për algjebër dhe të menduarit algoritmik.

Algoritmet mund te paraqiten edhe ne forme te flowchart apo diagrameve, kjo metode eshte e përdorur dhe thjeshton tejmase zbërthimin e algoritmit. Prandaj per ta kuptuar me mire do te marrim nje shembull me Algorimin e përshkruam ne forme te thjeshte, diagramit dhe kodit.

Algorithm ne flowchart diagram



1. Start
2. Read a,b,c
3. S= (a + b +c)/2
4. A= sqrt(s\*(s\*-a)\*(s-b)\*(s-c))
5. Print or Disply A
6. Stop

Kodi:

#include<iostream>

#include <cmath>

Unsing namespace std;

Int main ()

{

Int a,b,c;

Double s, A;

Count << “Enter values of a, b, c” <<endl;

cin >> a>>b>>c;

s= (a+b+c)/2;

A= sqrt (s\*(s-a)\*(s-b)\*(s-c));

Cout << “Area of triangle is:”<<A;

Return 0;

}

# Brute Force Algoritmet

Algoritmi Brute Force: Shpjeguar, Rastet e Përdorimit dhe Shpjegimi Matematikor

Algoritmi i Brute Force është një qasje e drejtpërdrejtë dhe shteruese për zgjidhjen e problemeve që përfshin përpjekjen e të gjitha zgjidhjeve të mundshme për të gjetur atë që plotëson një kriter të caktuar. Megjithëse nuk është metoda më efikase, ajo garanton një zgjidhje të saktë duke eksploruar sistematikisht çdo opsion të mundshëm.

Përshkrimi algoritmik:

Algoritmi i forcës brutale përfshin përsëritjen e të gjitha zgjidhjeve të mundshme dhe testimin e secilës për të përcaktuar nëse e zgjidh problemin. Është njësoj si të kërkosh për një gjilpërë në një kashtë duke ekzaminuar çdo pjesë të barit.

Raste te perdorimit:

Algoritmet e forcës brutale përdoren zakonisht në skenarë ku metodat e tjera më të optimizuara janë jopraktike ose të panevojshme. Këtu janë disa raste përdorimi:

**Thyerja e fjalëkalimit:** Forca brutale mund të përdoret për të thyer fjalëkalimet duke provuar të gjitha kombinimet e mundshme derisa të gjendet e duhura.

**Kërkim shterues**: Kur kërkoni për një artikull ose zgjidhje specifike me një hapësirë të kufizuar kërkimi, si p.sh. zgjidhja e një enigme ose gjetja e një sekuence lëvizjesh në një lojë.

**Thyerja e enkriptimit**: Në rastet kur metodat e kriptimit kanë dobësi, forca brutale mund të përdoret për të provuar të gjithë çelësat e mundshëm të deshifrimit.

**Probleme kombinuese:** Disa probleme në matematikë dhe shkenca kompjuterike, siç është problemi i shitësit udhëtues, mund të zgjidhen duke përdorur forcën brutale për madhësi të vogla të dhënash.

# **Elaborimi matematikor:**

Le të marrim një problem matematikor për të ilustruar qasjen e forcës brutale: gjetja e vlerës minimale të një funksioni.

Problem: Gjeni vlerën minimale të funksionit f(x) = x2 - 4x + 4.

Qasja e Brute Force algoritmit:

Zgjidhni një domen për x, p.sh., -10 deri në 10.

Llogaritni f(x) për çdo x në interval.

Mbani gjurmët e f(x) më të vogël të hasur.

F(x) më e vogël i përgjigjet vlerës minimale të funksionit.

Shpjegimi matematikor:

Në këtë rast, algoritmi i forcës brutale vlerëson funksionin për çdo vlerë të mundshme të x në diapazonin e dhënë. Duke provuar sistematikisht të gjitha vlerat, siguron që asnjë minimum potencial të mos humbasë. Megjithatë, kjo qasje mund të marrë kohë dhe mund te thuhet se eshte joefikase për funksione më komplekse ose diapazon dhe domen më të madh.

Avantazhet dhe disavantazhet:

Përparësitë:

Garanton një zgjidhje të saktë per kërkesën konkrrete.

E thjeshtë dhe e lehtë për t'u zbatuar.

I përshtatshëm për madhësi të vogla të hyrjes ose skenarë ku optimizimi nuk është prioritet.

Disavantazhet:

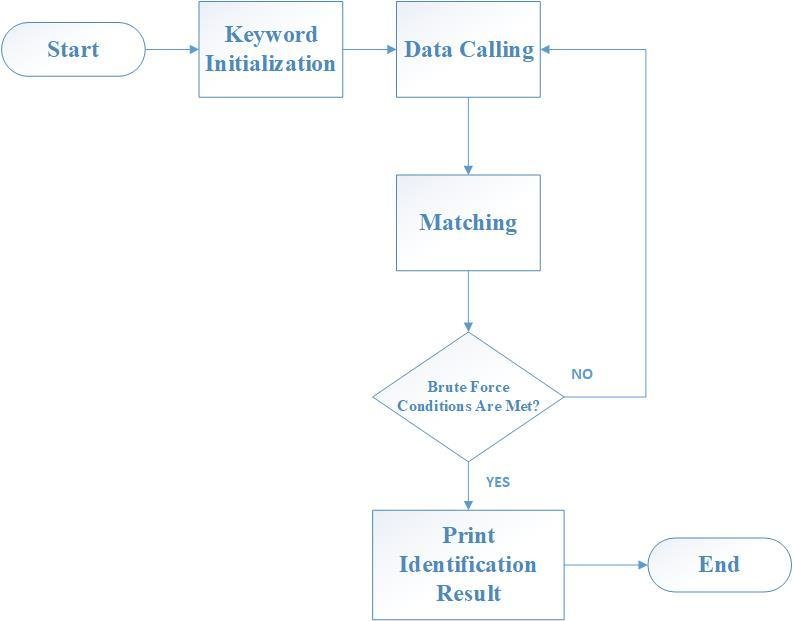
I paefektshëm për hapësira të mëdha kërkimi(Domene te mëdha te kërkimit).

Konsumon kohë për probleme komplekse(Nuk i eleminon mundësitë sipas nje logjike te caktuar por i provon te gjitha mundësitë duke kërkuar rrezulltatin e sakte gjithmonë).

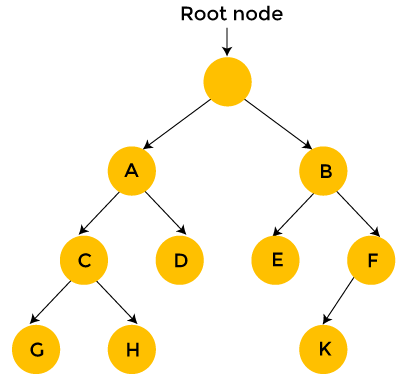
Jo i përshtatshëm për probleme me një numër të madh zgjidhjesh të mundshme.

konkluzioni:

Algoritmi i forcës brutale është një metodë e drejtpërdrejtë dhe e lehte poashtu edhe besueshme për zgjidhjen e problemeve duke provuar në mënyrë shteruese të gjitha zgjidhjet e mundshme. Ndërsa mund të mos jetë qasja më efikase në shumë raste, ajo shërben si një bazë për të kuptuar algoritme më të avancuara dhe optimizimet e tyre.



Flowchart diagrami i paraqitur me larte përshkruan ne menyre te thjeshte funksionimin e Brute force algoritmit. Si fillim inicializohen variablat dhe pastaj thirren te dhënat e nevojshme apo mund ta quajmë dhe si domeni i veprimit te algorimit. Nese ka rezultat te kërkuar sipas kushtit te dhene te kërkuara nga ana e programit atëherë algoritmi vjen me nje rezultat dhe programi merr fund, por nëse nuk gjindet rezultat me herën e pare, algoritmi do te hyj ne loop apo ne unaze qe do te riekzekutoj kushtin duke marre te dhëna nga domeni deri ne momentin qe do te arrij ne rezultat. Pasi qe Brute Force algoritmi eshte i bazuar ne rezultat dhe gjithmonë na jep nje rezultat te sakte pasiqe i provon te gjihta mundësitë ,por qe ne aspekt te vlersimit te kohës brute force algoritmi mbetet i mangët ne krahasim me algoritmet tjera.

Shembull: Supozoni se e kemi problemin në formën e pemës së treguar si më poshtë.

Kërkimi përmes Brute Force algoritmit do te merr në konsideratë çdo gjendje të një peme, dhe gjendja përfaqësohet në formën e një nyje. Për sa i përket pozicionit fillestar, ne kemi dy zgjedhje, d.m.th., një shtet dhe një shtet B. Ne mund të gjenerojmë ose gjendjen A ose gjendjen B. Në rastin e gjendjes B, ne kemi dy gjendje, d.m.th., gjendjen E dhe F. Në rastin e kërkimit me Brute force algoritmit, çdo shtet konsiderohet një nga një. Siç mund të vërejmë në pemën e mësipërme se kërkimi i forcës brutale kërkon 12 hapa për të gjetur zgjidhjen.

Nga ana tjetër, backtracking, i cili përdor kërkimin Depth-First, merr parasysh gjendjet e mëposhtme vetëm kur gjenja ofron një zgjidhje të mundshme. Merrni parasysh pemën e mësipërme, filloni nga nyja rrënjë, pastaj kaloni te nyja A dhe më pas nyja C. Nëse nyja C nuk ofron zgjidhjen e realizueshme, atëherë nuk ka kuptim të merren parasysh gjendjet G dhe H. Ne kthehemi nga nyja C në nyja A. Më pas kalojmë nga nyja A në nyjen D. Meqenëse nyja D nuk ofron zgjidhjen e mundshme, ne e hedhim poshtë këtë gjendje dhe kthehemi nga nyja D në nyjen A. Kalojmë në nyjen B, pastaj kalojmë nga nyja B në nyjen E. Kalojmë nga nyja E në nyjen K; Meqenëse **K është një zgjidhje**, kështu që duhen 10 hapa për të gjetur zgjidhjen. Në këtë mënyrë, ne eliminojmë një numër më të madh gjendjesh në një përsëritje të vetme. Prandaj, mund të themi se kthimi prapa është më i shpejtë dhe më efikas se qasja e forcës brutale.

# Algoritmet e Brute Force dhe Greedy: të përbashkëtat dhe dallimet

Algoritmet Brute force dhe Greedy janë dy qasje të dallueshme për zgjidhjen e problemeve në shkencën kompjuterike. Ndërsa ata kanë disa të përbashkëta, ato gjithashtu ndryshojnë në strategjitë dhe karakteristikat e tyre. Le të shqyrtojmë aspektet dhe dallimet e tyre të përbashkëta.

Aspekte të përbashkëta:

**Kërkimi shterues**: Të dy algoritmet e babëzitur dhe të forcës brutale përfshijnë eksplorimin e të gjitha zgjidhjeve ose opsioneve të mundshme brenda një hapësire të caktuar kërkimi.

**Qëllimet e optimizmit**: Të dyja qasjet synojnë të gjejnë një zgjidhje optimale, pavarësisht nëse është zgjidhja më e mirë e mundshme ose një zgjidhje që plotëson kritere specifike.

**Procesi përsëritës**: Të dyja qasjet zakonisht përfshijnë një proces përsëritës ku hapat ndërmerren në mënyrë sekuencave dhe çdo hap kontribuon në përsosjen e zgjidhjes.

**Dallimet:**

**Brute Force:** Algoritmet e forcës brutale eksplorojnë sistematikisht të gjitha zgjidhjet e mundshme, të cilat mund të jenë joefikase për madhësi më të mëdha të problemeve për shkak të natyrës së tyre shteruese.

**Greedy**: Algoritmet e babëzitur bëjnë zgjedhje lokale optimale në çdo hap, duke shpresuar se këto zgjedhje do të çojnë në një zgjidhje optimale globalisht. Ata marrin vendimin më të mirë në çdo hap pa marrë parasysh pasojat e ardhshme.

**Kompleksiteti:**

Brute Force shpesh kanë kompleksitet të lartë kohor dhe hapësinor, veçanërisht për problemet me hapësira të mëdha kërkimi.

Greedy algoritmet pritet të kenë kompleksitet më të ulët kohor sepse nuk marrin parasysh të gjitha mundësitë. Megjithatë, ata nuk mund të gjejnë gjithmonë zgjidhjen më të mirë.

Optimaliteti:

Brute Force: garantojnë gjetjen e zgjidhjes optimale sepse eksplorojnë të gjitha mundësitë. Ato janë të besueshme, por mund të jenë jopraktike për probleme të mëdha.

Greedy: e gjejnë gjithmonë zgjidhjen optimale globalisht, sepse ato fokusohen në optimizimin lokal në çdo hap. Ato mund të çojnë në zgjidhje jo optimale në disa raste.

Aplikimet e dy algoritmeve janë te ngjajshme por sa i përket Brute Force algoritmit përdoret shpesh kur hapësira e kërkimit është e vogël ose kur optimizimi nuk është kritik. Është i përshtatshëm për probleme ku numri i mundësive është i menaxhueshëm.

Greedy - përdoren kur një problem mund të ndahet në nënprobleme më të vogla dhe optimizimi lokal kontribuon në optimizimin global. Ato janë të zakonshme në problemet e optimizimit.

Marrja e Vendimeve:

Brute Force: Algoritmet e forcës brutale ndjekin një proces sistematik të vlerësimit të të gjitha mundësive, pa marrë parasysh nëse një zgjedhje e veçantë është më e mirë se të tjerat.

Greedy: Algoritmet e babëzitur marrin vendime bazuar në përfitimin e menjëhershëm, pa marrë parasysh ndikimin e mundshëm afatgjatë në zgjidhje.

Shembull:

Për një shembull konkret, merrni parasysh "Problemin e kthimit te kusurit me sasi sa me te paket te kartëmonedhave apo monedhave metalike" ku duhet të jepni një sasi të caktuar ndryshimi duke përdorur sa më pak monedha të mundshme. Një qasje e Brute Force do të përfshinte përpjekjen e të gjitha kombinimeve të monedhave për të gjetur më të mirën. Casja tjetër e algoritmit Greedy mund të përfshijë gjithmonë zgjedhjen e prerjes më të madhe të monedhës në dispozicion në çdo hap. Ndërsa qasja greedy algoritmit mund të funksionojë në disa raste, ajo mund të çojë në zgjidhje jo optimale, ndërsa qasja e forcës brutale garanton zgjidhjen më të mirë. Të dy algoritmet synojnë të gjejnë zgjidhje optimale, ato ndryshojnë në strategjitë, efikasitetin dhe optimalitetin e tyre. Algoritmet e pangopur bëjnë zgjedhje lokale optimale, ndërsa algoritmet e forcës brutale kërkojnë në mënyrë shteruese të gjitha mundësitë. Zgjedhja midis të dyjave varet nga karakteristikat e problemit dhe kërkesat e optimizmit.

# Vegla per gjetjen e subdomeneve përmes Brute Foce algoritmit

Vegla per gjetjen e subdomeneve përmes perdorimt te algoritmit Brute Force eshte e shkruar ne disa gjuhe programuese. Algoritmi i Brute Force algoritmit eshte i shkruar ne gjuhen programuese Python duke i përdorur dhe libraritë shoqëruese relevante.Pjesa e kodit qe përshkruan algoritmin eshte e lidhur me nje front end te shkruar ne gjuhen programues te HTML5 dhe CSS. Permes Front end mundësohet qasje me e mire vizuele ndaj programit dhe do te paraqiten rezultatet e gjeneruara përmes searchit. Te dhënat ngarkohen përmes butonit “Ngarko” me qe rast aty do te ngarkohet nje liste e siguruar nga interneti dhe kjo liste ngarkohet ne back end (PHP database). Pasi qe te ngarkohet lista, shtypim butonin search dhe pas nje kohe te caktuar programi do te gjeneroj nje përgjigje apo ndryshe do ti listoj te disa subdomene te nje domeni përcaktues ashtu qe te rezultatet do te shafqen ne front end dhe poashtu ne back-end PHP database.

Sa i përket rezultateve do ti mund te krijojmë edhe një mendim te përgjithshëm rreth domenit duke pare fushëveprimin dhe ndarjen e domenit ne nën domene. Kjo është e nevojshme ne shume fusha te studimit te cilat i kemi potencuar me pare.

Per te startuar programin per ekzekutim duhet qe te instalohet programi XAMPP.

XAMPP është një paketë softuerësh falas, me burim të hapur që ofron një mjedis zhvillimi lokal për aplikacionet në ueb. XAMPP perdoret per Web Local development dhe per testime te platformave qe përdorin databaze dhe vegza tjera qe i përkrah XAMPP.

. Akronimi "XAMPP" qëndron për:

X: I referohet ndonjë prej sistemeve operative (Windows, Linux, macOS).

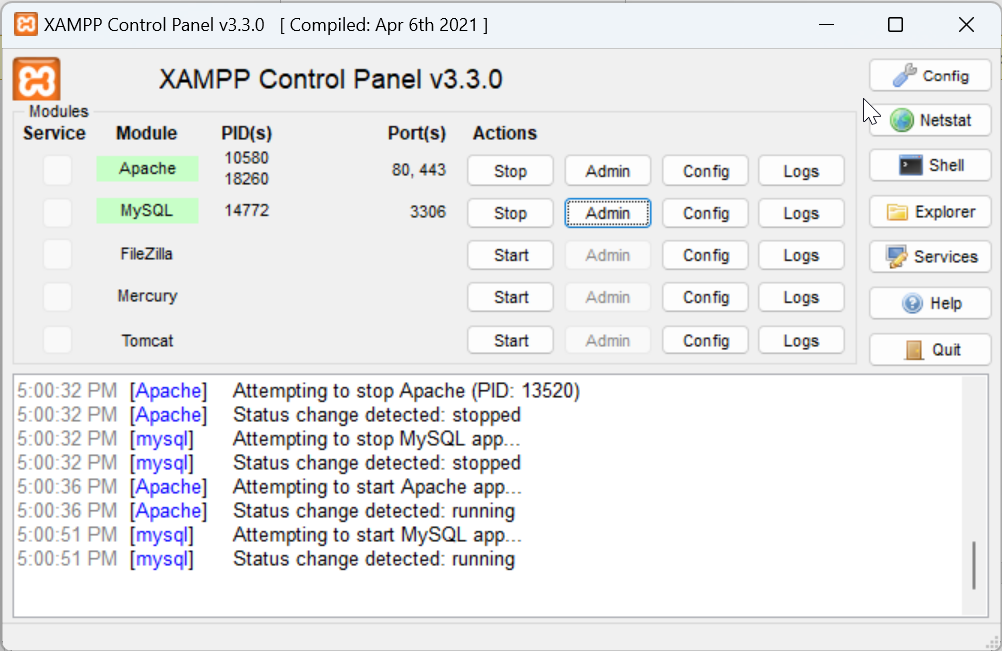
Përgjigje: Serveri HTTP Apache

M: MySQL ose MariaDB (sistemet e menaxhimit të bazës së të dhënave)

P: PHP

P: Perl

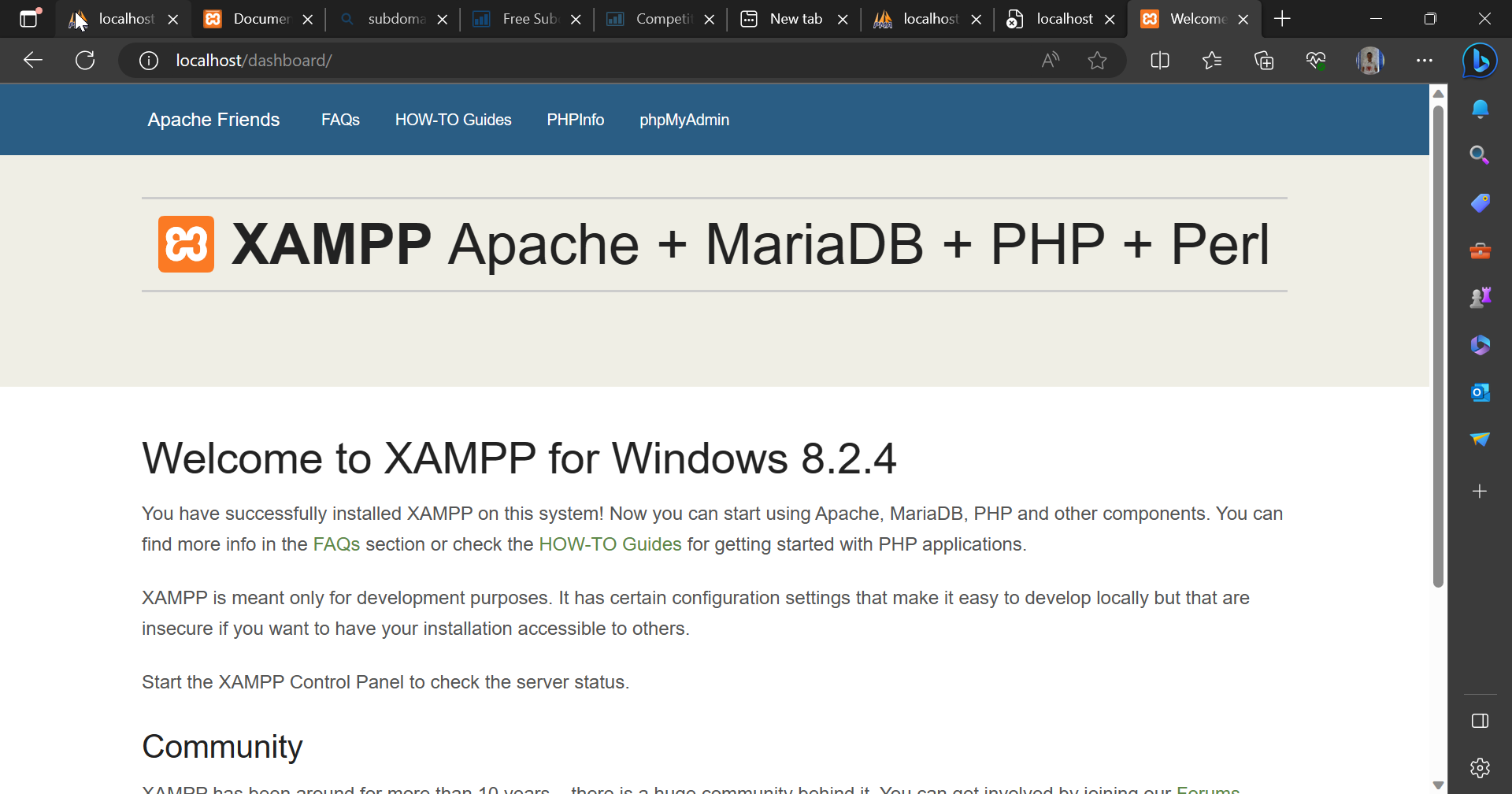
XAMPP i bashkon këta komponentë së bashku, duke e bërë të lehtë për zhvilluesit të konfigurojnë një mjedis server lokal për testimin dhe zhvillimin e aplikacioneve në ueb pa pasur nevojë të instalojnë secilin komponent veç e veç. XAMPP ne vete përmban Apache HTTP Server, MySQL/ MairaDB, PHP, PHP myAdmin.



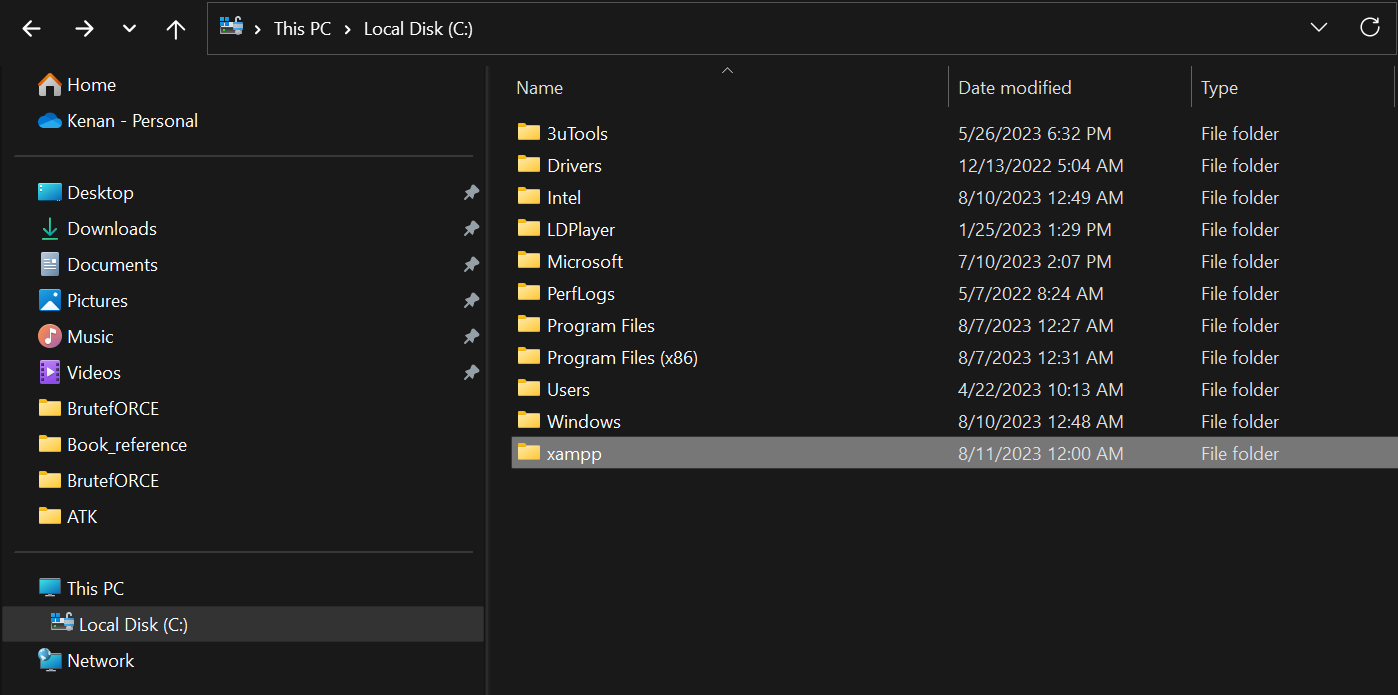


Pikesepari duhet te startohen APACHE dhe pastaj duhet te startohet edhe MySQL.

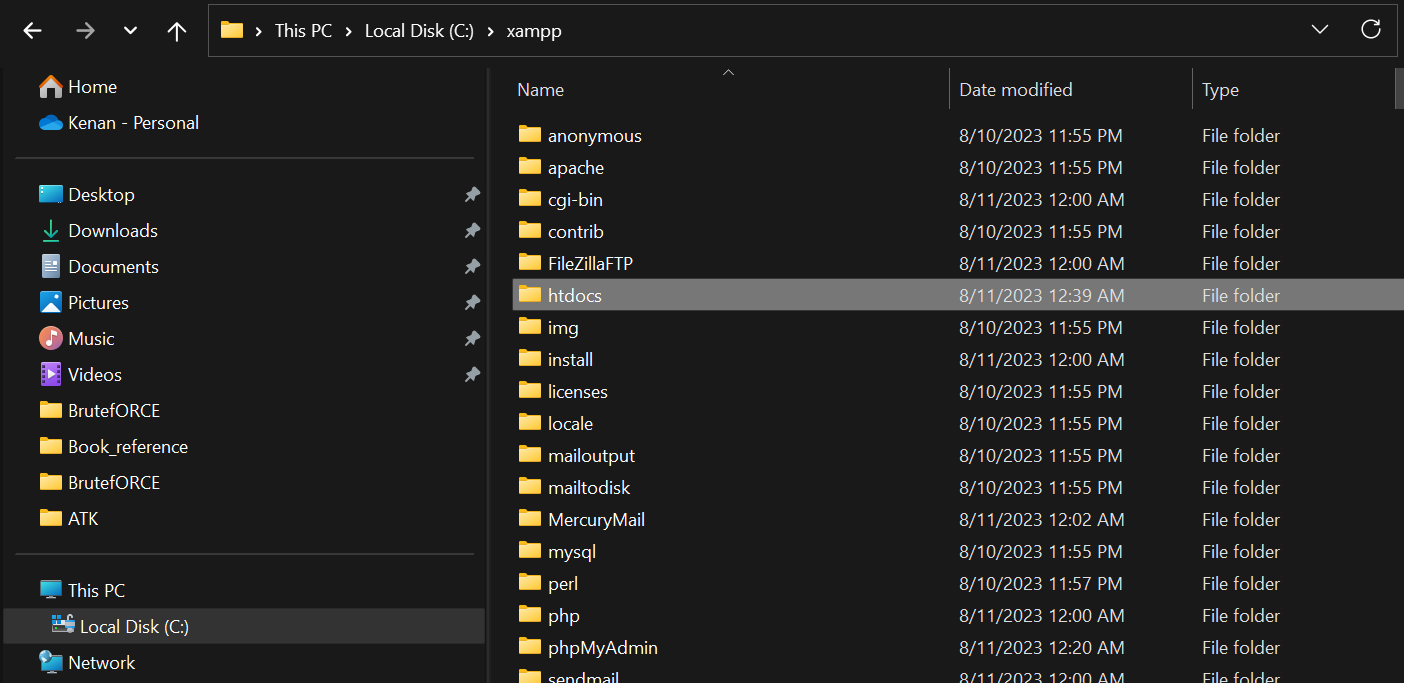
Pastaj duhet te startohet nje Browser mund te perdoret Google Chrome apo Microsoft Edge, dhe ne te kërkohet ne pjsen e search bar “localhost”, ne rast se eshte cdo gje ne rregull dhe programi eshte i konfiguruar ashtu sic duhet, ne browser do te shfaqet, si me poshtë:



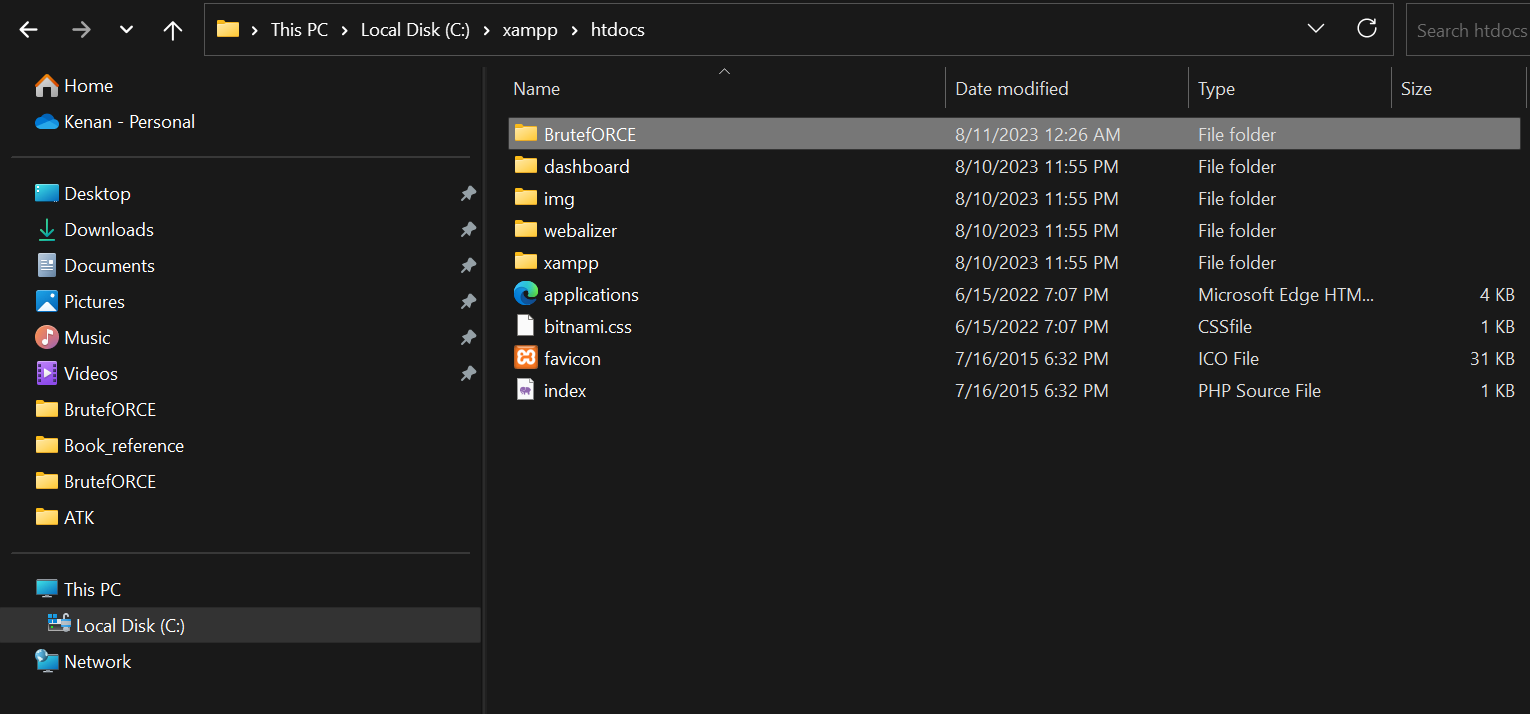
Hapi i mëtutjeshëm eshte qe ta vendosim filet e programit ne folderin burimor te XAMPP ashtu qe te thirret programi dhe te vazhdohet me inicializim te mëtutjeshëm. Prandaj i ndjekim hapat te përshkruar si me poshtë:



Drejtohuni ne C apo ne local diskun ne te cilin e keni te instaluar programin XAMPP. Pastaj duhet te zgjidhet folderi me emrin “xampp”.



Selektoni folderin “htdoocs”, dhe vendosni te fajllat e programit te bashkuara ne nje folder te vetëm.



Duhet te kopjohet emri i folderit te projektit, qe ne kete rast eshte “BrutefORCE”. Pastaj vazhdojmë me search bar te browserit. “localhost/BrutefORCE” dhe do te gjenerohet fornt end i vegezes i lidhur me databaze.

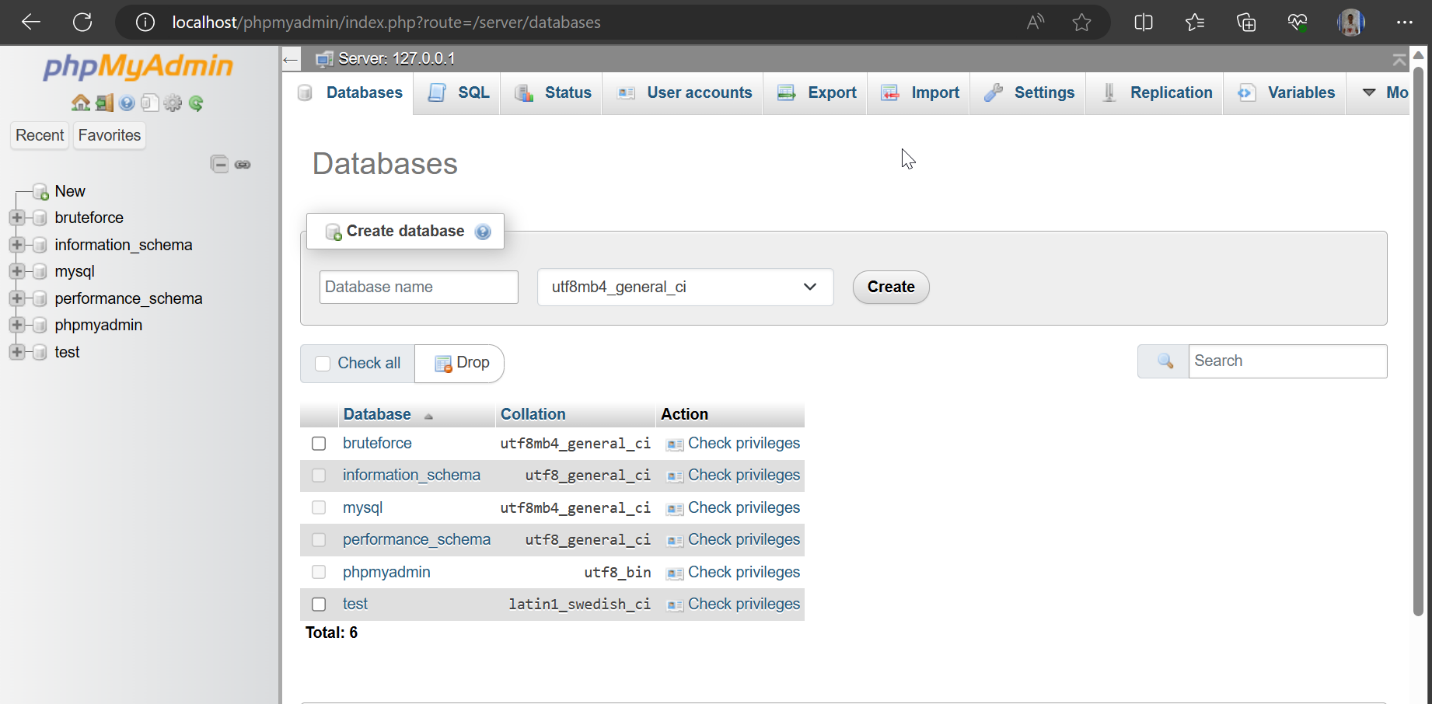


Interface per përdorimin e vegezes eshte paraqitur ne pamjen me sipër. Ne momentin kur shtypet butoni “Upload” uploadohet file me lista te subdomeneve, rrjedhimisht ngarkohet ne databaze. Ne pjesën e search mund ti kërkojmë subdomenet te cilat janë pjese e listës se ngarkuar ne databaze.

 Search bar per Kerim te subdomenave përkatës.

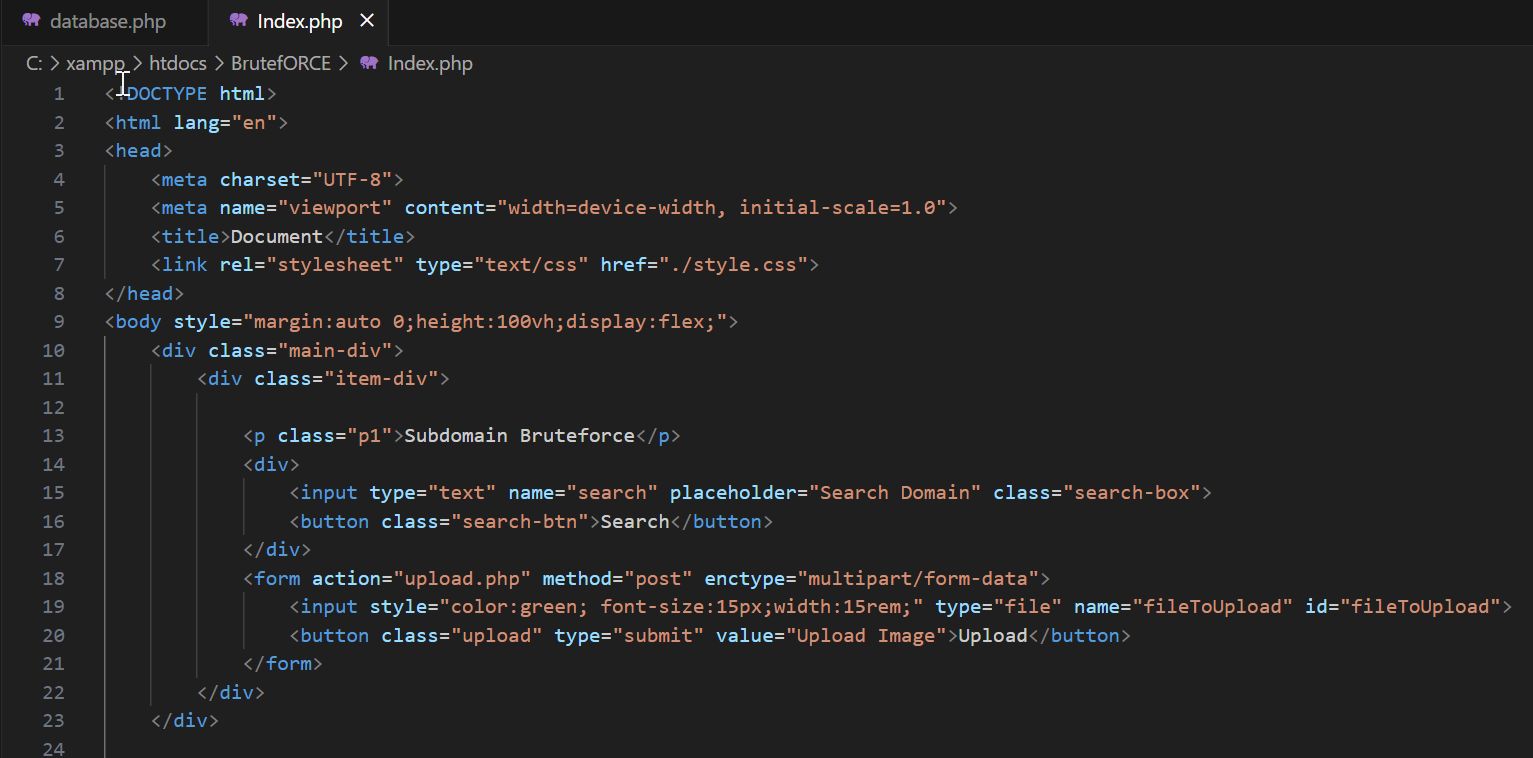
 Butoni per ngarkim te fajllit, qe do te perdoret per analizim.

 Ekzekutimi i komandës per ngarkim te fajllit.

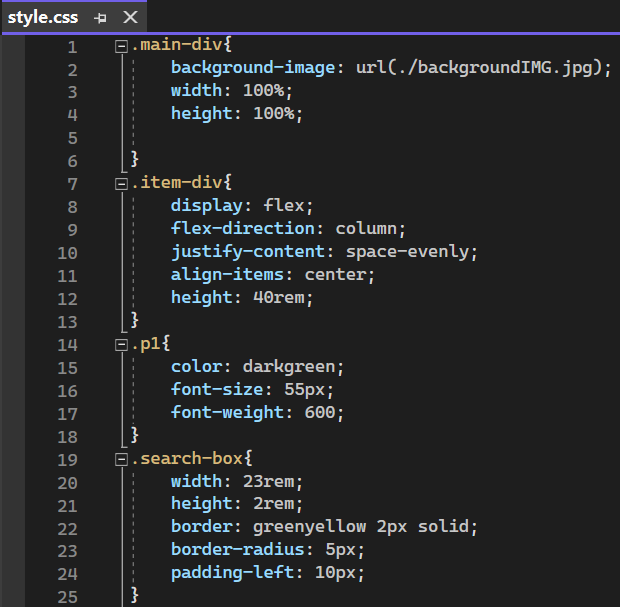


Pamje nga PHPMyAdmin.

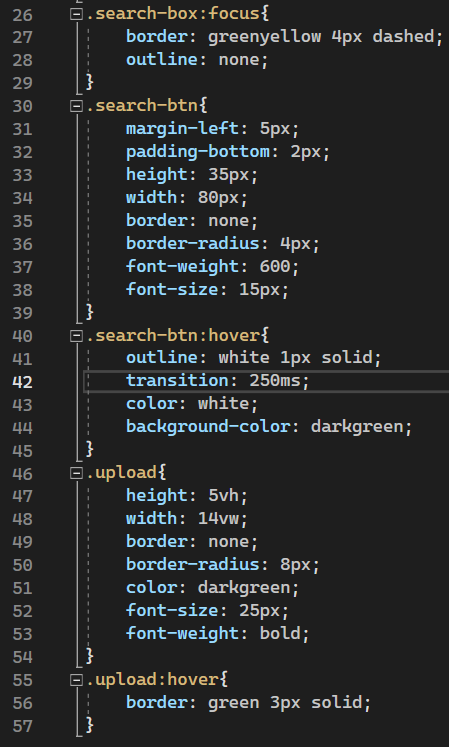
Pjesa e kodit e shkruar ne gjuhen programuese HTML 5

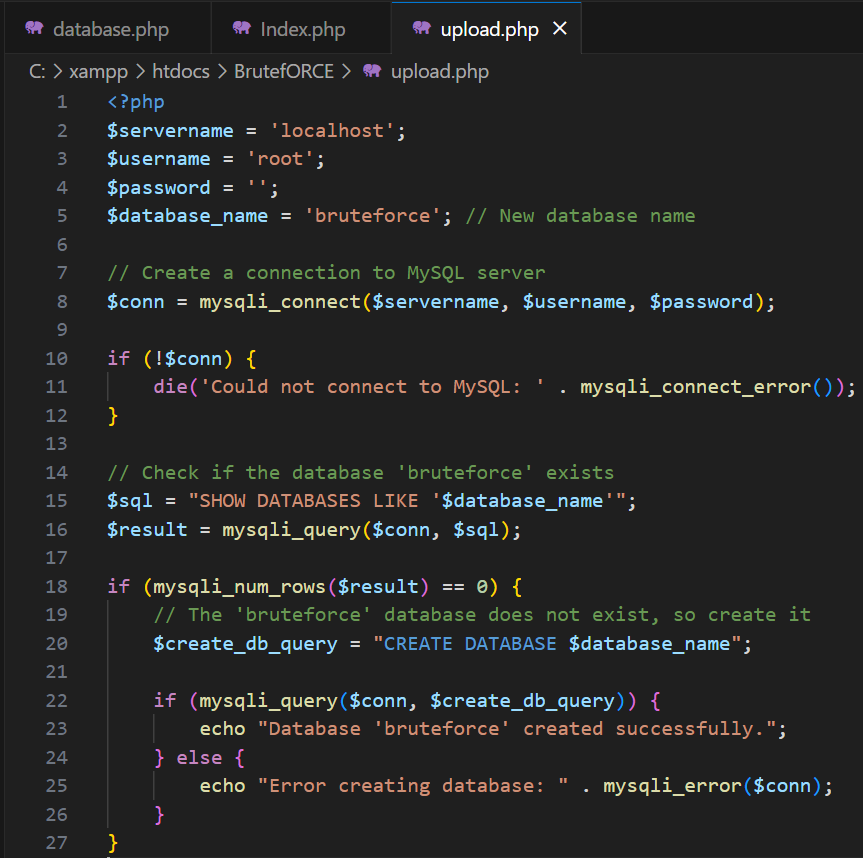


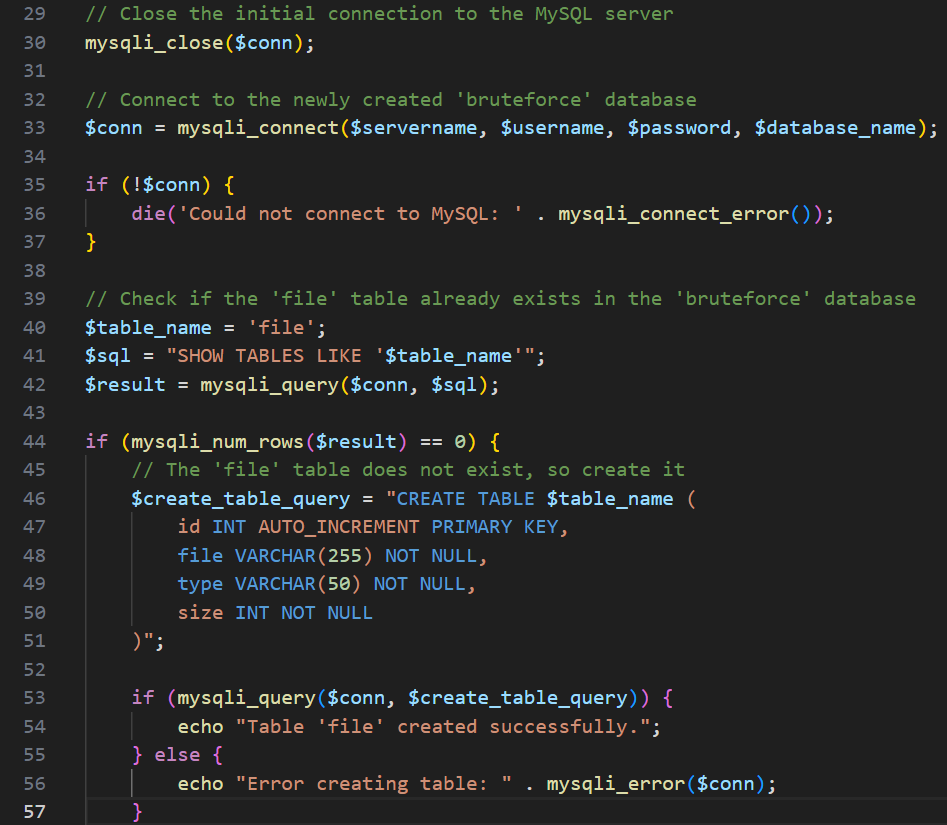
das

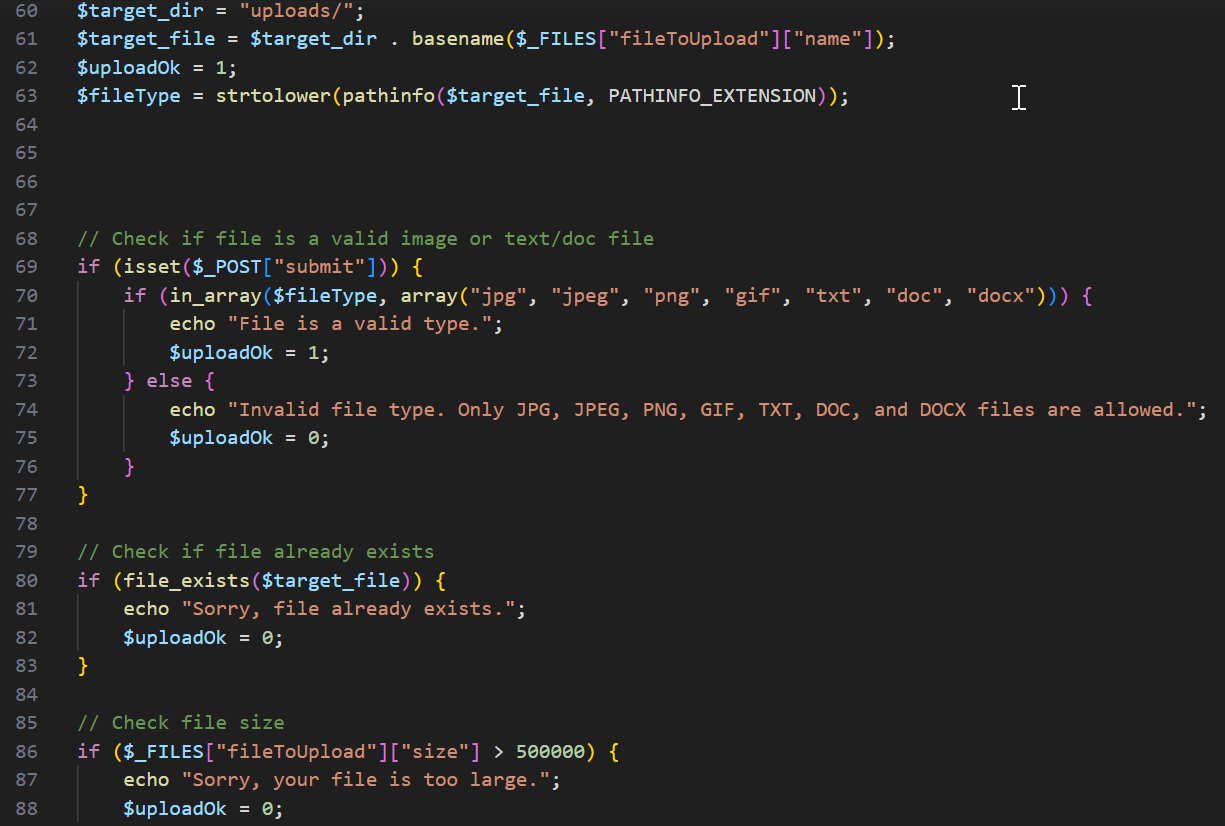


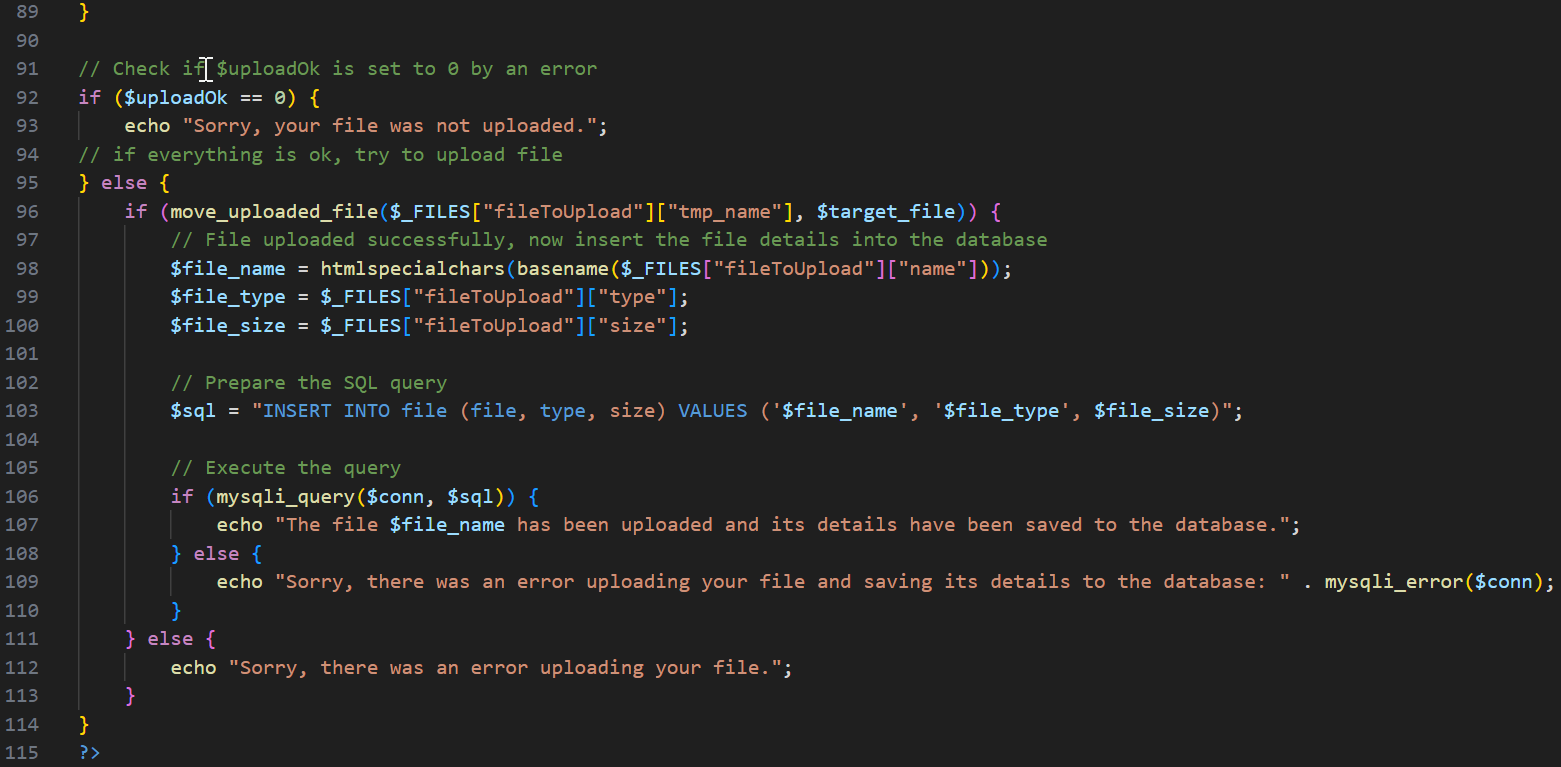
Kodi dhe parametrat ne CSS.











# Referencat