**Universiteti i Prishtinës “Hasan Prishtina”**

**Fakulteti Inxhinierisë Elektrike dhe Kompjuterike**



**Dokumentim teknik i projektit**

|  |  |  |
| --- | --- | --- |
| Punoi : | Gentrit Ademi | [gentrit.ademi2@student.uni-pr.edu](mailto:gentrit.ademi2@student.uni-pr.edu) |

Prishtinë, 2024

Përmbajtja

[**Abstrakti** 3](#_Toc168695268)

[**I.** **Hyrje** 4](#_Toc168695269)

[**II.** **Qëllimi i punimit** 5](#_Toc168695270)

[**III.** **Pjesa kryesore** 6](#_Toc168695271)

[Shpjegim i detajuar i procesit 6](#_Toc168695272)

[Shpjegim i detajuar i procesit 7](#_Toc168695273)

[Shpjegim i detajuar i pamjes 22](#_Toc168695274)

[Përfitimet dhe rëndësia e kësaj pamjeje 23](#_Toc168695275)

[**IV.** **Konkluzione** 24](#_Toc168695276)

[Referencat 25](#_Toc168695277)

# **Abstrakti**

Punimi "Big Data Homework \_ 2" është një udhëzues i përgatitjes së mjedisit dhe manipulimit të të dhënave në dy kontekste të ndryshme: manipulimi i të dhënave XML dhe manipulimi i të dhënave NoSQL. Tema e punimit është të ofrojë udhëzime dhe detyra për studentët në lidhje me përgatitjen e mjedisit dhe ekzekutimin e query-ve në dy mjedise të ndryshme të bazave të të dhënave: XML dhe NoSQL. Përvoja e përdorimit të mjediseve të ndryshme të bazave të të dhënave është e rëndësishme për studentët që të zhvillojnë njohuri të thella në këto fusha dhe për të pasur një përvojë praktike në përdorimin e tyre. Në rishikimin e literaturës për zgjidhjet e propozuara, autorët përdorin mjete të ndryshme si Saxon, Altova XML, MongoDB, Cassandra, etj., për të manipuluar dhe ekzekutuar query në të dhëna XML dhe NoSQL. Përdorimi i këtyre mjeteve është në përputhje me praktikat e mira të fushës dhe ofron një gamë të gjerë të mundësive për manipulimin dhe shqyrtimin e të dhënave në kontekstin e Big Data. Përdorimi i mjeteve të specifikuara dhe implementimi i query-ve të caktuara janë zgjidhjet më të mira për të përmbushur qëllimin e punimit. Kjo temë është e interesante sepse ofron studentëve një përvojë të strukturuar dhe praktike në manipulimin e të dhënave në mjedise të ndryshme të bazave të të dhënave, duke përfshirë të dhëna të strukturuara dhe të pandryshueshme si edhe të dhëna NoSQL. Kjo përvojë është e rëndësishme për zhvillimin e tyre të njohurive dhe aftësive në fushën e Big Data.

# **Hyrje**

Hyrja në punimin tonë ofron një përshkrim të arsyetimit pse është interesante të diskutohet për problemet e adresuara, duke përfshirë motivimin tonë personal, rëndësinë e fushës dhe të temës së zgjedhur, si dhe sfidat dhe kontributet potenciale në këtë fushë. Gjithashtu, ofron një përmbledhje të problematikës së përgjithshme që do të adresohet dhe udhëzime për zgjedhjen e saj. Për grupin tonë, diskutimi i temës së manipulimit të të dhënave të mëdha (Big Data) është interesant për shkak të rëndësisë së saj në kohën tonë të digitalizuar. Si pjesë e një projekti të zhvillimit të një platforme analitike, përdorimi i teknologjive për manipulimin e të dhënave është një aspekt kyç i punës sonë. Punimi në këtë fushë është e rëndësishme për zhvillimin dhe përparimin e teknologjive të të dhënave, të cilat kanë ndikim thelbësor në sektorin e teknologjisë dhe biznesit. Sfida kryesore në këtë temë është përdorimi i mjeteve dhe teknologjive të përshtatshme për manipulimin e të dhënave të mëdha, si dhe zbatimi i strategjive të efektshme për shqyrtimin dhe analizën e tyre. Çfarë është e re në këtë temë është avancimi i teknologjive dhe metodologjive për trajtimin dhe analizën e sasisë së madhe të të dhënave, duke përfshirë ndërhyrjen në fushën e NoSQL dhe manipulimin e të dhënave XML. Kontributi ynë në këtë fushë do të jetë në zgjidhjen e problemeve konkrete të manipulimit të të dhënave të mëdha duke përdorur teknologjitë dhe mjete të përshtatshme për këtë qëllim. Problemi i zgjedhur duhet të jetë një sfidë e vërtetë në fushën e manipulimit të të dhënave, duke pasur parasysh rëndësinë dhe kompleksitetin e tij, si dhe potencialin për kontribut të rëndësishëm në fushën e teknologjisë së të dhënave. Zgjedhja e problematikës duhet të bazohet në një analizë të kujdesshme të nevojave të sektorit dhe mundësive për inovacion dhe përparim.



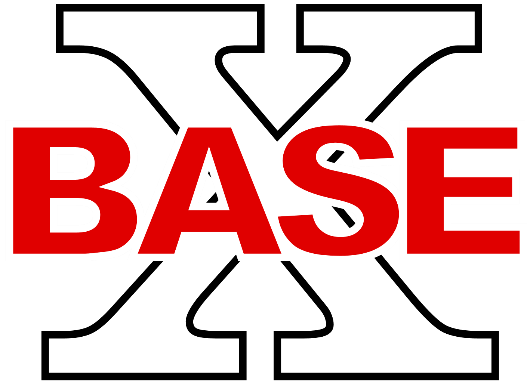


Figura 1: Ilustron mjedisin e përdorur gjatë punës sonë. Ne kemi përdorur MongoDB Compass si një mjedis redaktimi për MongoDB ((Mongodb Composer Editor - Google Search, 2024) për manipulimin dhe analizën e të dhënave në MongoDB. Në të njëjtën kohë, për të punuar me të dhëna XML dhe për të zhvilluar query-t për to, kemi përdorur BASEX ((Basex - Google Search, 2024), një mjedis i specializuar për të manipuluar dhe analizuar të dhënat XML. Kombinimi i këtyre mjediseve na ka lejuar të punojmë efektivisht me të dy llojet e të dhënave dhe të zhvillojmë query-t e nevojshme në kontekstin e punës sonë.

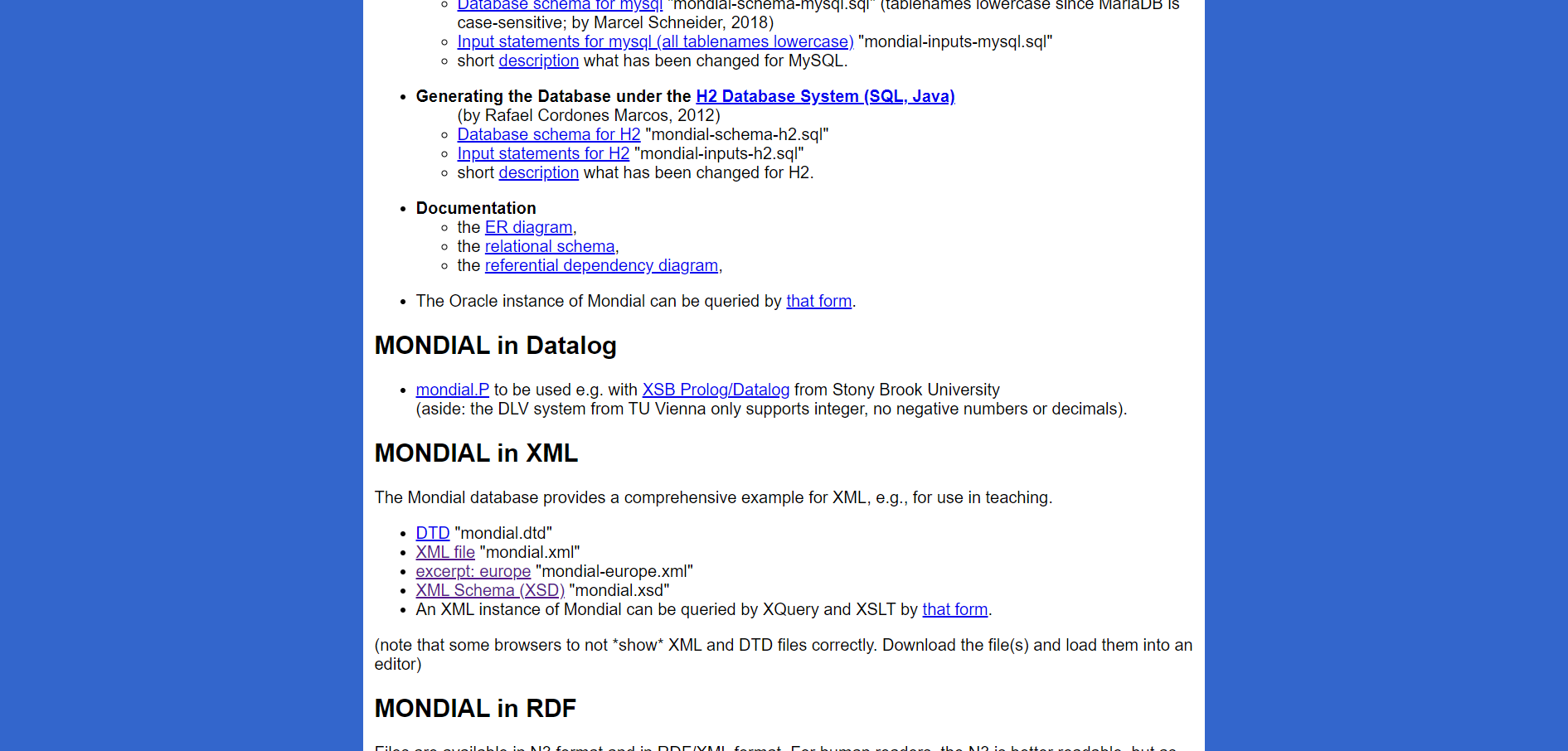
# **Qëllimi i punimit**

Qëllimi i punimit tonë është të përdorim mjete dhe teknologji të përshtatshme për të manipuluar dhe analizuar të dhënat në kontekstin e Big Data, duke përfshirë manipulimin e të dhënave XML dhe NoSQL. Arsyet për përzgjedhjen e mjeteve që përdorim janë të ndryshme. Për përpunimin e të dhënave XML, kemi zgjedhur BASEX për shkak të aftësisë së tij për të manipuluar dhe analizuar të dhënat XML në një mënyrë të efektshme dhe të specializuar. Një përparësi e këtij mjedisi është që ofron funksionalitete të zgjeruara dhe performancë të lartë për punën me të dhënat XML. Për punën me të dhënat NoSQL, kemi zgjedhur MongoDB Compass për shkak të lehtësisë së përdorimit dhe aftësisë së tij për të manipuluar të dhënat në MongoDB në një mënyrë intuitive dhe efektive. Një veçanti e përdorimit të këtyre mjeteve është se ato ofrojnë një gamë të gjerë të funksionaliteteve dhe mundësive për manipulimin dhe analizën e të dhënave në mjedisin e tyre të specializuar. Përparësitë e këtyre mjeteve janë aftësia e tyre për të përpunuar sasi të mëdha të të dhënave në mënyrë efektive dhe për të ofruar performancë të lartë në shqyrtimin dhe analizën e tyre. Megjithatë, mangësitë mund të përfshijnë një krësim të lartë të komplikacioneve teknike për konfigurimin dhe përdorimin e tyre në disa raste. Zgjidhjet tona më të mira që ofrojmë me përdorimin e këtyre mjeteve në projektin tonë janë aftësia për të manipuluar dhe analizuar të dhënat në mënyrë efektive dhe për të zhvilluar query-t e nevojshme në kontekstin e punës sonë. Metodat që kemi përdorur për të mbledhur informatat e duhura përfshijnë hulumtimin në dokumentacionin e mjeteve të përdorura, si dhe provën e tyre në kontekstin e punës sonë për të vlerësuar performancën dhe aftësitë e tyre për të përmbushur qëllimin tonë. Shënimet e mbledhura janë dokumentuar në mënyrë të rregullt për të ruajtur transparencën dhe për të lehtësuar përdorimin e tyre në fazën e zhvillimit të punës sonë.

# **Pjesa kryesore**

Në fillim të punës, kemi filluar duke zhvilluar një query XQuery specifik. Qëllimi i këtij query ishte të listonte kryeqytetet e vendeve që nuk janë anëtarë të NATO-s dhe në të cilat kalon të paktën dy lumenje. Kjo kërkesë specifike na udhëhoqi në hartimin e një strategjie të përshtatshme për të identifikuar dhe paraqitur kryeqytetet e vendeve të caktuara, duke përdorur aftësitë tona në shkrimin dhe shqyrtimin e të dhënave me XQuery.

Me poshte do te paraqesim konfigurimin e ambientit punues per xml.

  
  
**Figura 2**: Ilustron procesin e shkarkimit dhe vendosjes së skedarëve të nevojshëm për të punuar me të dhënat e Mondial-it në editorin BaseX.

### Shpjegim i detajuar i procesit

1. **Shkarkimi i skedarëve**:

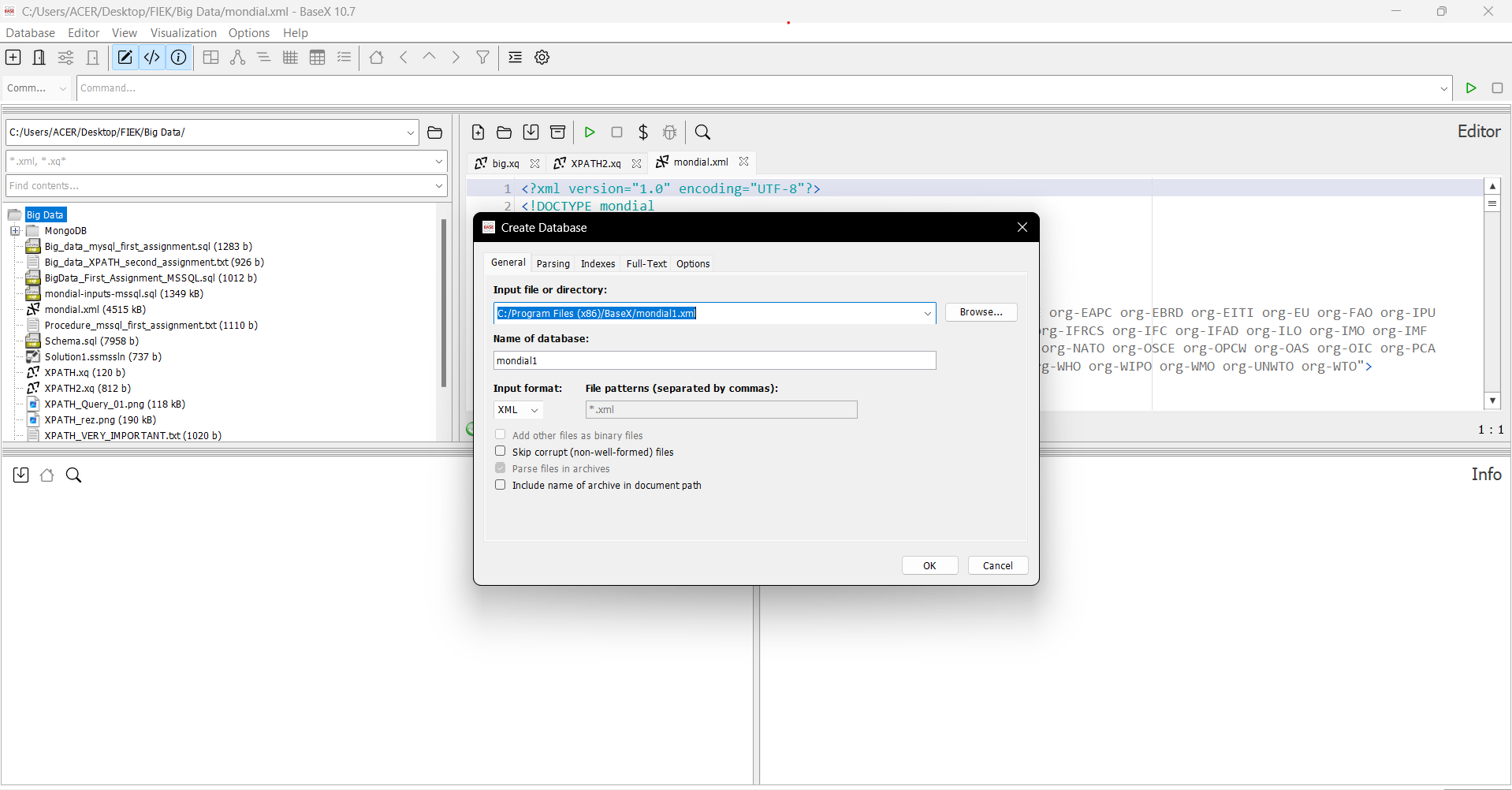
**mondial.dtd**: Ky është skedari DTD (Document Type Definition) që përshkruan strukturën dhe rregullat për të dhënat XML në skedarin mondial.xml. Ai përcakton elementet dhe atributet që mund të përdoren, si dhe hierarkinë dhe lidhjet midis tyre.

**mondial.xml**: Ky është skedari XML që përmban të dhënat reale të Mondial-it, të strukturuara sipas rregullave të përcaktuara në mondial.dtd. Ai përfshin informacione të ndryshme rreth vendeve, kryeqyteteve, lumenjve, organizatave ndërkombëtare, dhe shumë më tepër.

1. **Vendosja e skedarëve në folderin e BaseX**:

Pas shkarkimit, këta skedarë duhet të vendosen në një dosje specifike ku BaseX mund t'i qaset lehtësisht. Kjo zakonisht do të thotë vendosja e tyre në një dosje të dedikuar brenda strukturës së skedarëve të BaseX.

Kjo hap përgatit mjedisin e punës duke siguruar që editori BaseX ka qasje në të dhënat e nevojshme për të kryer analizat dhe manipulimet e kërkuara.



**Figura 3:** Ilustron procesin e krijimit të një baze të dhënash në BaseX, e cila është një hap themelor për të punuar me të dhënat XML në këtë editor të fuqishëm. Kjo procedurë përfshin disa hapa thelbësorë që sigurojnë që të dhënat të jenë të qasshme dhe të përgatitura për analizë dhe manipulim.

### Shpjegim i detajuar i procesit

1. **Hapja e BaseX**:

Filloni duke hapur aplikacionin BaseX. Kjo mund të bëhet përmes ikonës në desktop ose përmes menusë së programeve në sistemin tuaj operativ.

1. **Krijimi i një databaze të re**:

Në menunë e BaseX, zgjidhni opsionin për të krijuar një bazë të re të dhënash. Kjo zakonisht gjendet në menunë "Database" dhe pastaj "New".

Do të hapet një dritare e re ku duhet të vendosni një emër për bazën tuaj të dhënash dhe të specifikoni skedarin XML që dëshironi të përdorni.

1. **Specifikimi i skedarëve**:

Zgjidhni skedarin XML që keni vendosur më parë në dosjen e BaseX. Për këtë shembull, do të zgjidhni skedarin "mondial.xml".

Opsionalisht, mund të specifikoni edhe skedarin DTD "mondial.dtd" nëse dëshironi të përdorni rregullat e validimit të përcaktuara në këtë skedar.

1. **Importimi i të dhënave**:

Pasi të keni specifikuar skedarin XML, klikoni në butonin për krijimin e bazës së të dhënave. BaseX do të importojë të dhënat nga skedari XML dhe do të krijojë strukturën e nevojshme për t'i ruajtur ato.

Gjatë këtij procesi, BaseX do të analizojë dhe indeksë të dhënat për të siguruar qasje të shpejtë dhe efikase.

1. **Verifikimi i krijimit të databazës**:

Pasi të krijohet baza e të dhënave, do të mund të shihni të dhënat e importuara në panelin kryesor të BaseX. Kjo përfshin shfaqjen e strukturës së XML dhe mundësinë për të shfletuar elementet dhe atributet.

**xquery version "3.1";**

**let $mondial := db:open("mondial")**

**for $country in $mondial//country**

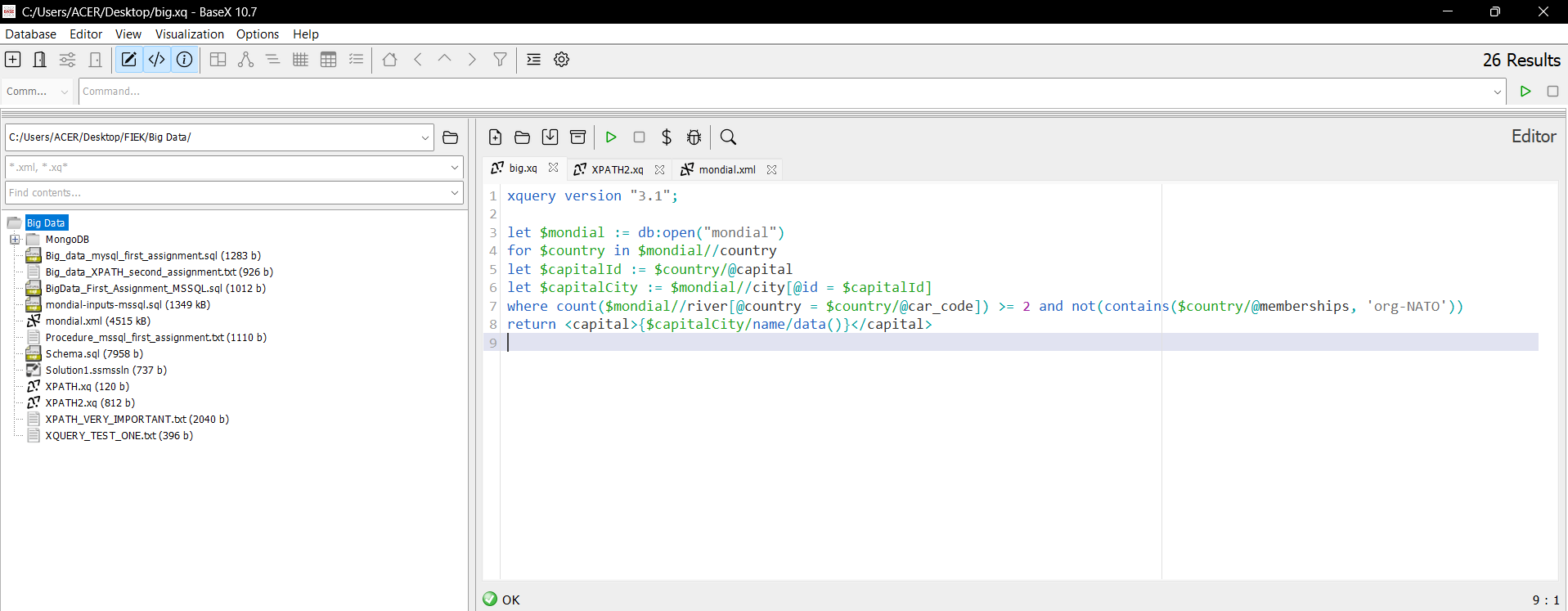
**let $capitalId := $country/@capital**

**let $capitalCity := $mondial//city[@id = $capitalId]**

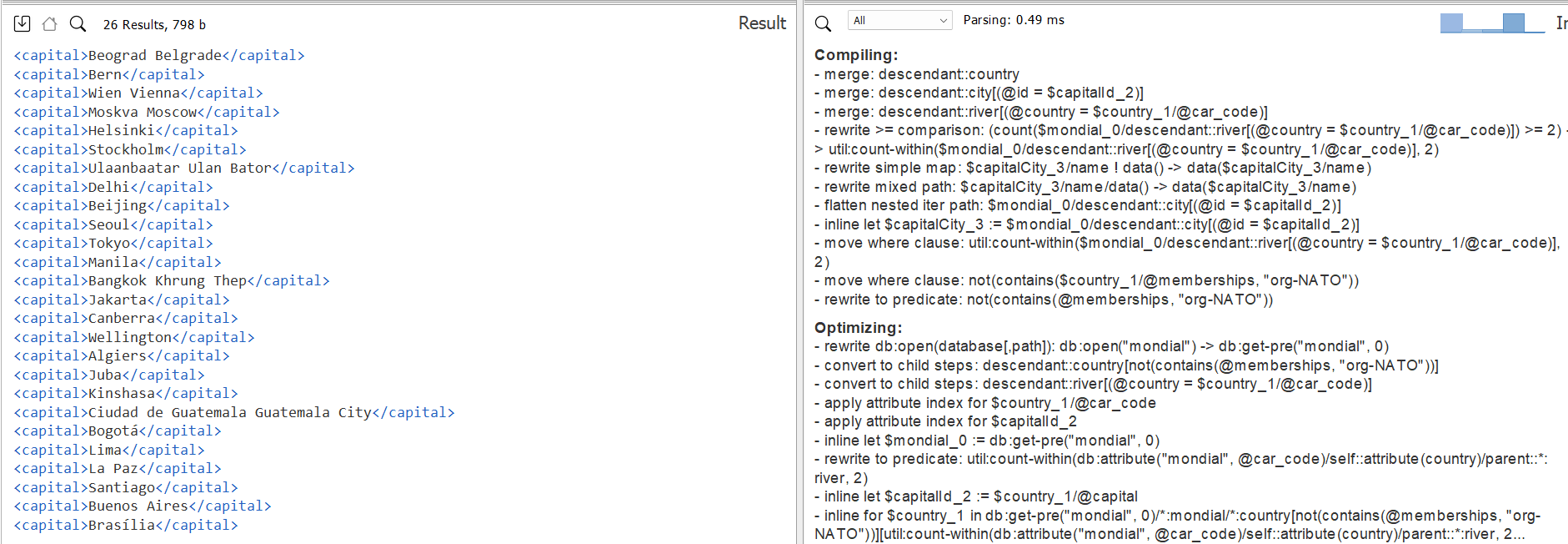
**where count($mondial//river[@country = $country/@car\_code]) >= 2 and not(contains($country/@memberships, 'org-NATO'))**

**return <capital>{$capitalCity/name/data()}</capital>**

Ky fragment i kodit XQuery ka një qëllim të qartë dhe specifik: të identifikojë kryeqytetet e vendeve që nuk janë anëtarë të NATO-s dhe në të cilat kalon të paktën dy lumenje. Për të filluar, kodit i është dhënë një variabël $mondial, e cila përfaqëson bazën e të dhënave "mondial". Pastaj, fillon një loop që për të gjitha vendet gjen një id të kryeqytetit dhe pastaj gjen informacionin për kryeqytetin në bazën e të dhënave. Ky loop filtratohet duke përdorur një kusht për të siguruar që vendi ka të paktën dy lumenje dhe nuk është anëtar i NATO-s. Kodi përdor funksionin count() për të numëruar lumenjet në vendin e caktuar, dhe funksionin not() për të kontrolluar nëse vendi nuk është anëtar i NATO-s. Në fund, kodit i kthehet një element XML i cili përmban emrin e kryeqytetit të vendit përkatës. Ky kod është e qartë dhe efektiv në arritjen e qëllimit të tij, duke përdorur shprehje të thjeshta dhe efikase në XQuery për të identifikuar dhe paraqitur kryeqytetet e vendeve të caktuara në mënyrë të përshtatshme.



**Figura 4:** Ofron një pamje të ambientit të punës sonë në BaseX editor, ku kemi shkruar dhe ekzekutuar kodin XQuery për të manipuluar bazën e të dhënave të Mondial-it. Kjo platformë na ofron një mënyrë të qartë dhe të përshtatshme për të zhvilluar dhe testuar query-t tona në mënyrë efikase. Nëpërmjet saj, mund të identifikojmë dhe të rregullojmë kodin tonë në mënyrë të shpejtë dhe të qëndrueshme.



**Figura 5:** Jep një pamje të rezultateve të kthyer pas ekzekutimit të XQuery-t në bazën e të dhënave të Mondial-it në BaseX. Kjo pamje na ofron një përmbledhje të rezultateve të përmbledhura dhe të paraqitura në një mënyrë të strukturuar dhe të lexueshme. Nëpërmjet kësaj pamjeje të rezultateve, jemi në gjendje të identifikojmë dhe të analizojmë të dhënat e kthyer në mënyrë të qartë dhe të efektshme.

Përveç XQuery-së, kemi përdorur edhe XPath për të realizuar një kërkesë të veçantë: të listohen të gjithë lumenjte që kalojnë nëpër vendet anëtare të NATO-s dhe BE-së, duke përjashtuar Norvegjinë.  
  
 xPath:

**doc("mondial")//river[name and (contains(@country, 'GR') or contains(@country, 'MNE') or contains(@country, 'F') or contains(@country, 'E') or contains(@country, 'CZ') or contains(@country, 'SK') or contains(@country, 'D') or contains(@country, 'H') or contains(@country, 'PL') or contains(@country, 'NL') or contains(@country, 'I') or contains(@country, 'B') or contains(@country, 'L') or contains(@country, 'S') or contains(@country, 'DK') or contains(@country, 'P') or contains(@country, 'BG') or contains(@country, 'RO') or contains(@country, 'LV') or contains(@country, 'LT') or contains(@country, 'EE') or contains(@country, 'HR') or contains(@country, 'EST') or contains(@country, 'SLO') or contains(@country, 'IS') or contains(@country, 'GB') or contains(@country, 'TR'))]/name**

Ky kod XPath ka një qëllim të qartë: të gjejë emrat e lumenjve të cilët kalojnë në vendet anëtare të NATO-s dhe BE-së, duke përfshirë emrat e këtyre vendeve. Për ta bërë këtë, kodi shikon për lumenjtë në dokumentin e të dhënave të Mondial-it (mondial.xml) dhe seleksionon ato që janë në vendet specifike, duke përdorur funksionin contains() për të verifikuar praninë e emrit të vendit në atributin @country. Përveç kësaj, për të siguruar që seleksioni bëhet vetëm për vendet anëtare të NATO-s dhe BE-së, përdoret një listë e vendeve të specifikuara në atributin @country. Për të bërë kodin më lexues dhe më të kuptueshëm, emrat e vendeve janë shkurtuar në formë të shkronjave të kodit, të cilat përfaqësojnë shtetet e caktuara.

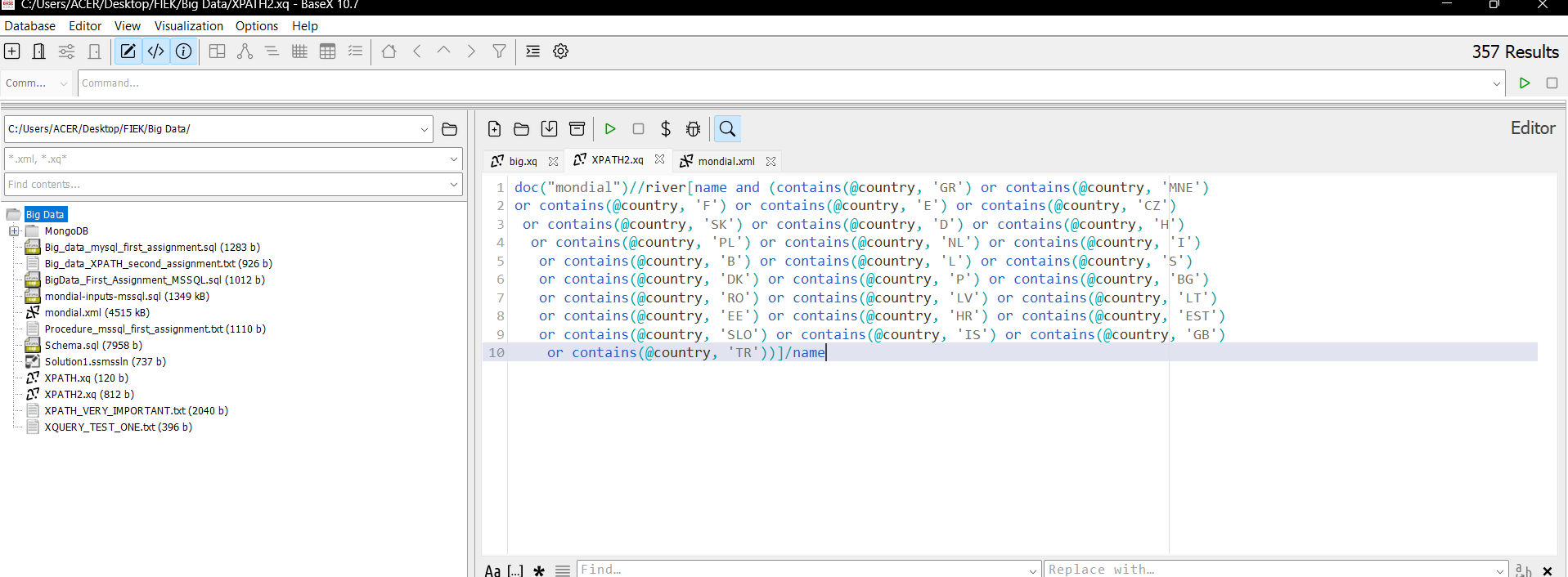
Për shembull:

'GR' për Greqinë

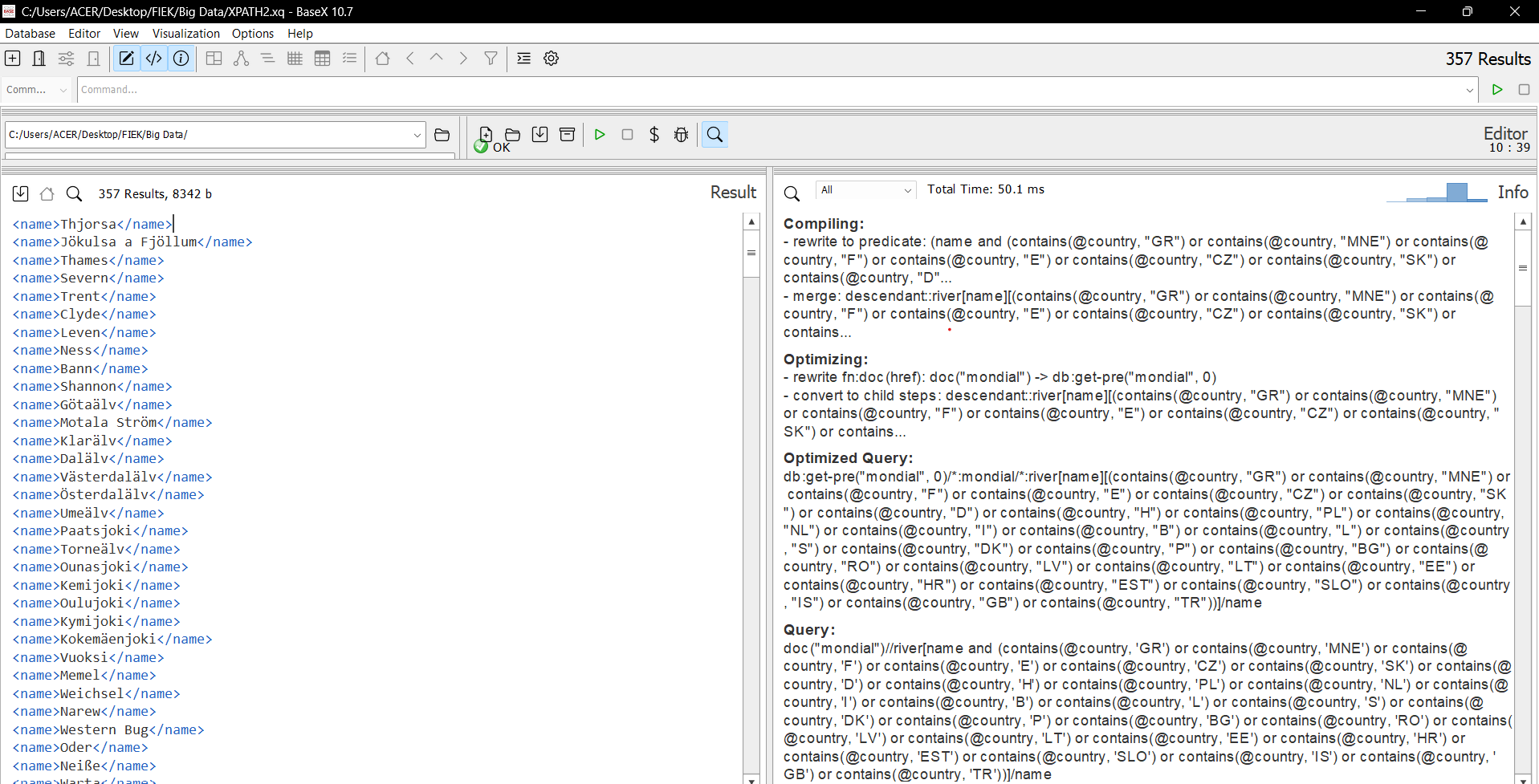
'MNE' për Malin e Zi

'F' për Francën

dhe kështu me radhë për vendet e tjera. Në fund, përzgjidhen emrat e lumenjve përkatës duke përdorur sintaksën e zakonshme për të selektuar emrin e lumenjit ("/name"). Kjo kërkesë XPath na jep një mënyrë të përshtatshme për të gjetur emrat e lumenjve të caktuar në kontekstin e punës sonë me të dhënat e mëdha të përdorura në projektin tonë.



**Figura 6:** Na ofron një pamje të ambientit të punës sonë në editorin BaseX, ku kemi shkruar dhe ekzekutuar një kërkesë XPath për të gjeneruar rezultate. Kjo pamje na lejon të shikojmë kodin tonë XPath dhe rezultatet e kthyer nga ekzekutimi i tij në bazën e të dhënave të Mondial-it.



**Figura 7:** Na jep një pamje të rezultateve të kthyer pas ekzekutimit të kërkesës sonë XPath në editorin BaseX. Kjo pamje përmban një përmbledhje të rezultateve të kthyer nga kodi XPath dhe mundëson një shqyrtim të qartë dhe të strukturuar të informacionit të gjetur në bazën e të dhënave të Mondial-it. Pamja e rezultateve na lejon të shohim emrat e lumenjve të kthyer në mënyrë të detajuar dhe të përshtatshme për përmbajtjen e kërkesës sonë. Kjo na jep një mundësi për të vlerësuar rezultatet e kthyer dhe për të siguruar që ato përputhen me pritshmëritë tona dhe me kërkesën specifike të kodit tonë XPath.

**doc("mondial")//country[contains(@memberships, "NATO") and contains(@memberships, "EU") and not(contains(@name, "Norway"))]/@car\_code**

Ky XPath ka një qëllim të qartë: të gjejë kodet e shteteve që janë anëtarë të NATO-s dhe BE-së, duke përjashtuar Norvegjinë. Për ta bërë këtë, kodi shikon për shtetet në dokumentin e të dhënave të Mondial-it (mondial.xml) dhe seleksionon ato që përmbajnë atributin e anëtarësimit "NATO" dhe "EU", por jo Norvegjinë. Kjo kërkesë seleksionon kodet e shteteve të cilat përmbajnë atributin e anëtarësimit "NATO" dhe "EU", duke përjashtuar Norvegjinë, dhe pastaj kthen këto kodi të shteteve si rezultat. Kjo është një mënyrë e shkëlqyeshme për të përmirësuar kërkesën e mëparshme dhe për të përmbushur kërkesat e veçanta të seleksionit në bazën e të dhënave të Mondial-it. Ky XPath na jep një mjet të fuqishëm për të identifikuar dhe për të analizuar shtetet e anëtara të NATO-s dhe BE-së, duke përjashtuar një shtet të caktuar, në këtë rast, Norvegjinë.

Në projektin tonë, përveç përdorimit të XQuery dhe XPath, kemi pasur edhe detyrën për të punuar me një bazë të dhënash NoSQL. Për këtë qëllim, kemi përdorur MongoDB dhe kemi zhvilluar një query specifike për të listuar emrat e kryeqyteteve të provincave të Kanadasë, duke përfshirë numrin e popullsisë dhe faktin që në to kalon të paktën një lum.

**Stage 1**

**Perzgjedhim te gjitha provincat ku country = kanada**

**{**

**Country: "CDN"**

**}**

**Stage 2**

**{**

**from: "located",**

**localField: "CapProv",**

**foreignField: "Province",**

**as: "Rivers"**

**}**

**Stage 3**

**{**

**Rivers: {**

**$ne: []**

**}**

**}**

**Stage 4**

**{**

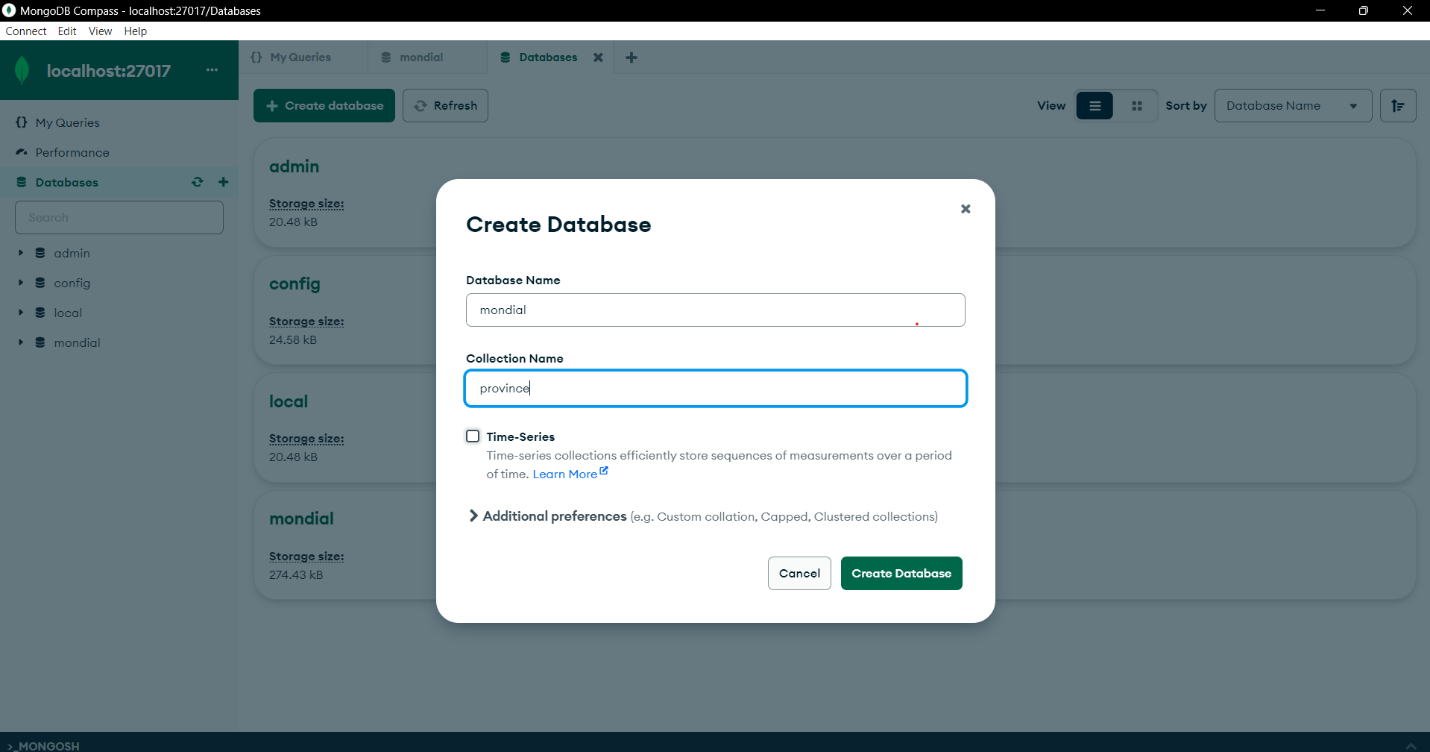
**Capital: "$Capital",**

**Population: "$Population",**

**Province: "$CapProv",**

**\_id: 0**

**}**

Më poshtë janë paraqitur disa hapa kryesorë për krijimin e databazës dhe importimin e të dhënave në mjedisin tonë të zhvillimit.

**Figura 8:** Ilustron procesin e krijimit të një baze të dhënash në MongoDB Compass, një ndërfaqe grafike për përdoruesit e MongoDB që e bën më të lehtë administrimin dhe analizimin e të dhënave në këtë sistem NoSQL. Procesi i krijimit të një baze të dhënash në MongoDB Compass përfshin disa hapa të thjeshtë, të cilët janë të rëndësishëm për të përgatitur mjedisin për manipulim dhe analizë të të dhënave.

#### Shpjegim i detajuar i procesit

1. **Hapja e MongoDB Compass**:

Filloni duke hapur aplikacionin MongoDB Compass. Kjo mund të bëhet përmes ikonës në desktop ose përmes menusë së programeve në sistemin tuaj operativ.

1. **Lidhja me serverin MongoDB**:

Në dritaren kryesore të MongoDB Compass, do të shihni një fushë për të futur informacionin e lidhjes me serverin MongoDB. Nëse jeni duke përdorur një server lokal, mund të përdorni "localhost" dhe portin standard (zakonisht 27017).

1. **Krijimi i një databaze të re**:

Pas lidhjes me serverin MongoDB, klikoni në butonin "Create Database" që ndodhet zakonisht në këndin e majtë të dritares kryesore.

Do të hapet një dritare e re ku duhet të vendosni një emër për bazën tuaj të dhënash dhe një emër për koleksionin fillestar. Koleksioni është ekuivalenti i një tabele në një databazë relacionale.

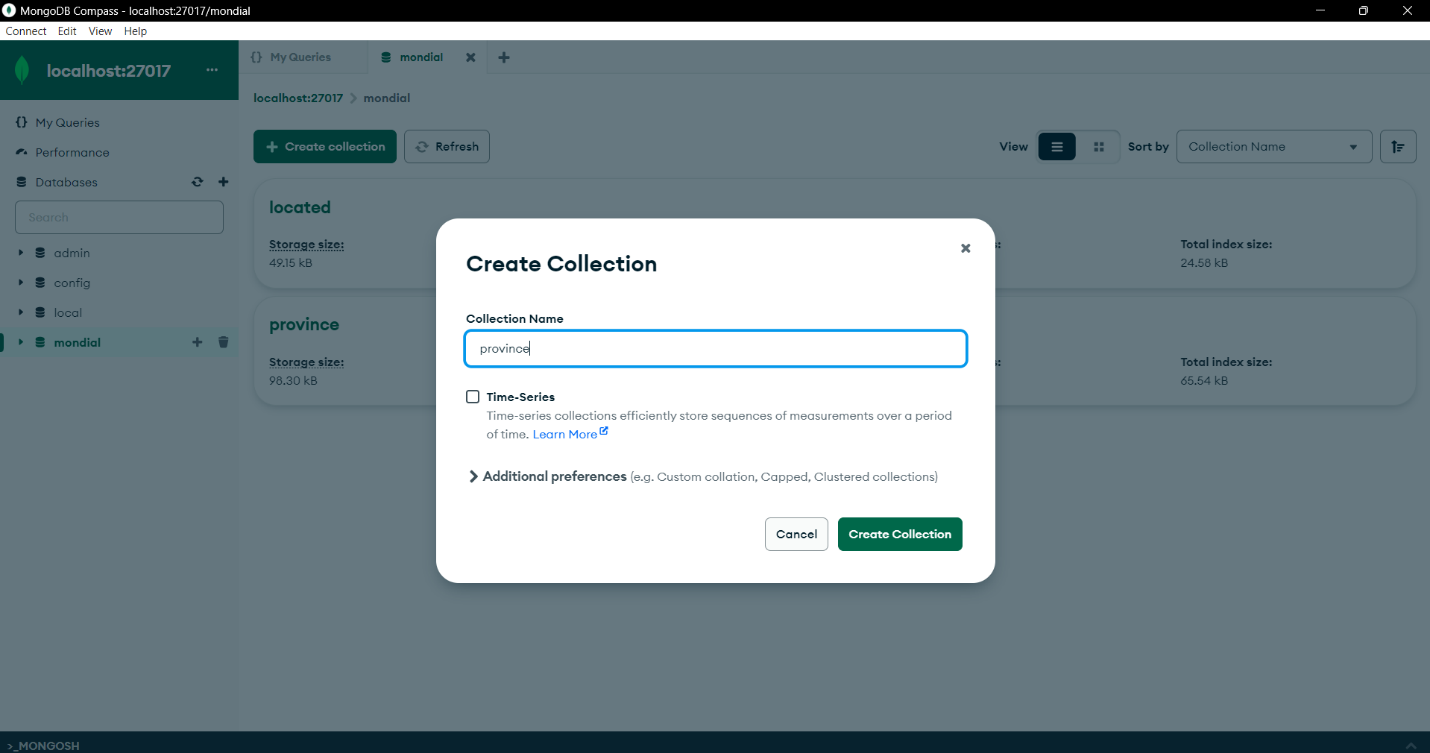
1. **Specifikimi i skemës fillestare**:

Në dritaren e krijimit të bazës së të dhënave, fusni emrin e dëshiruar për bazën e të dhënave dhe koleksionin fillestar. Për shembull, mund të vendosni emrin "CanadaProvinces" për bazën e të dhënave dhe "Provinces" për koleksionin fillestar.

Klikoni butonin "Create Database" për të përfunduar procesin e krijimit.

1. **Verifikimi i krijimit të databazës**:

Pasi të krijohet baza e të dhënave, do të mund të shihni bazën e re të dhënave dhe koleksionin e saj në panelin kryesor të MongoDB Compass. Kjo përfshin shfaqjen e strukturës dhe mundësinë për të shtuar, shfletuar dhe modifikuar dokumentet brenda koleksionit.



**Figura 9: I**lustron procesin e krijimit të një koleksioni të ri brenda një baze të dhënash në MongoDB Compass. Ky hap është thelbësor për organizimin dhe menaxhimin e të dhënave në MongoDB, ku të dhënat ruhen në formën e dokumenteve brenda koleksioneve.

#### Shpjegim i detajuar i procesit

1. **Hapja e MongoDB Compass**:

Filloni duke hapur aplikacionin MongoDB Compass dhe lidheni me serverin tuaj MongoDB.

1. **Navigimi te Baza e të Dhënave**:

Pasi të keni krijuar bazën e të dhënave (p.sh., "CanadaProvinces"), klikoni mbi emrin e saj në panelin e majtë për të hapur detajet e bazës së të dhënave.

1. **Krijimi i Koleksionit të Ri**:

Në dritaren kryesore të bazës së të dhënave, klikoni butonin "Create Collection".

Do të hapet një dritare e re ku duhet të vendosni një emër për koleksionin tuaj të ri. Për shembull, mund të vendosni emrin "Provinces".

1. **Specifikimi i Opsioneve të Koleksionit**:

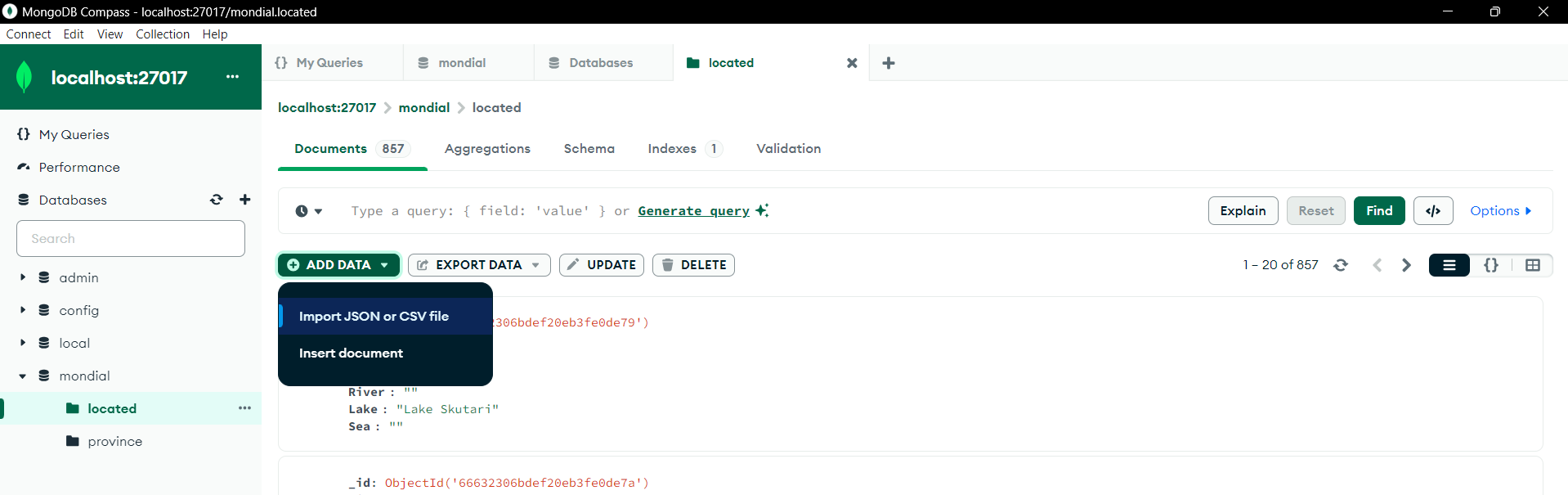
Opsionalisht, mund të specifikoni opsione të avancuara për koleksionin, siç janë indekset ose rregullat për validimin e dokumenteve.

Pasi të keni vendosur emrin dhe opsionet e dëshiruara, klikoni butonin "Create" për të përfunduar krijimin e koleksionit.

1. **Verifikimi i Koleksionit**:

Pasi të krijohet koleksioni, ai do të shfaqet në listën e koleksioneve brenda bazës së të dhënave në panelin e majtë të MongoDB Compass.

Klikoni mbi emrin e koleksionit të ri për të parë dhe menaxhuar dokumentet brenda tij.



**Figura 10:** Ilustron procesin e importimit të të dhënave në MongoDB Compass duke përdorur një skedar JSON. Ky hap është thelbësor për të sjellë të dhënat e jashtme në një koleksion të ri ose ekzistues në bazën tuaj të të dhënave.

#### Shpjegim i Detajuar i Procesit

1. **Hapja e MongoDB Compass**:

Filloni duke hapur aplikacionin MongoDB Compass dhe lidheni me serverin tuaj MongoDB.

1. **Navigimi te Koleksioni i Duhur**:

gjidhni bazën e të dhënave dhe koleksionin ku dëshironi të importoni të dhënat. Nëse koleksioni nuk ekziston, krijoni një të ri duke ndjekur hapat e përshkruar më parë.

1. **Fillimi i Procesit të Importimit**:

Klikoni në butonin "Import Data" që ndodhet në krye të dritares së koleksionit.

Do të hapet një dritare e re për të zgjedhur skedarin që dëshironi të importoni.

1. **Zgjedhja e Skedarit JSON**:

Klikoni në butonin "Select File" dhe gjeni skedarin JSON në kompjuterin tuaj që përmban të dhënat që dëshironi të importoni.

Pasi të keni zgjedhur skedarin, do të shihni opsionet për formatin e skedarit. Sigurohuni që formati i zgjedhur të jetë "JSON".

1. **Konfigurimi i Opsioneve të Importimit**:

Pasi të keni zgjedhur skedarin, mund të konfiguroni opsionet e importimit. Këto mund të përfshijnë:

* + - **JSON Mode**: Zgjedhja ndërmjet "Array of Documents" dhe "Each Line is a Document".
    - **Import Options**: Opsione të ndryshme siç janë "Ignore Errors" dhe "Stop on Errors".

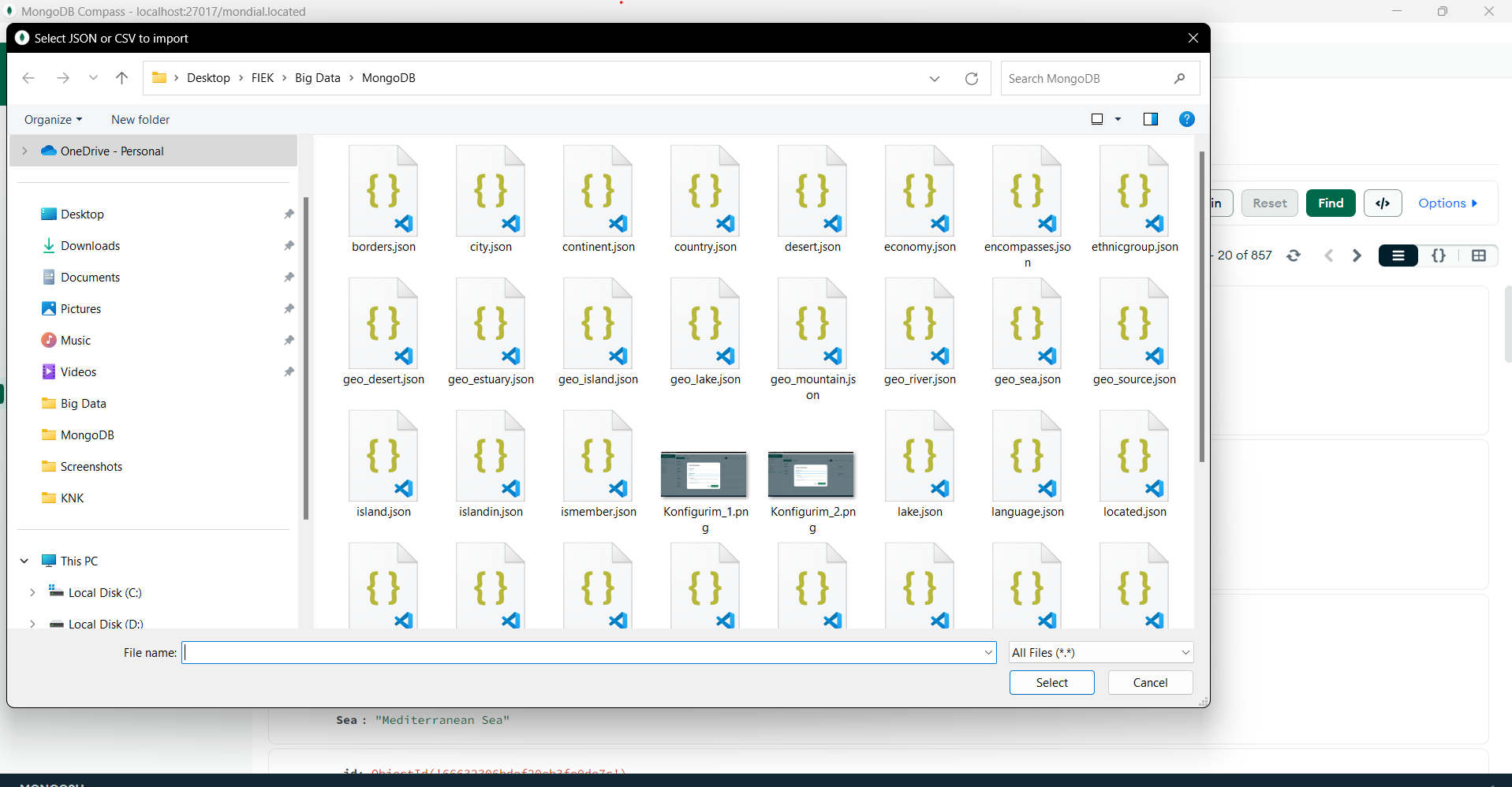
1. **Ekzekutimi i Importimit**:

Klikoni në butonin "Import" për të filluar procesin e importimit. MongoDB Compass do të lexojë skedarin JSON dhe do të importojë dokumentet në koleksionin e zgjedhur.

Një mesazh konfirmimi do të shfaqet pasi importimi të përfundojë me sukses, duke treguar numrin e dokumenteve të importuara.

1. **Verifikimi i të Dhënave të Importuara**:

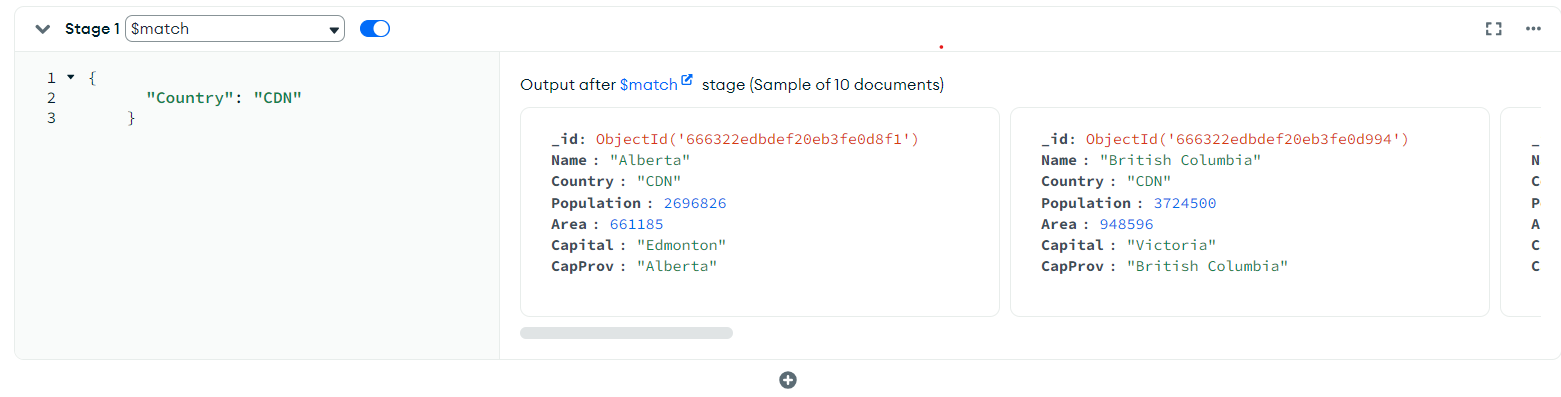
Pas përfundimit të importimit, mund të shfletoni dhe verifikoni të dhënat e importuara brenda koleksionit. Sigurohuni që të dhënat janë të sakta dhe të strukturuara siç pritej.



**Figura 11:** Ilustron procesin e zgjedhjes së skedarit të caktuar për importimin e të dhënave në MongoDB Compass. Ky hap është thelbësor për të siguruar që të dhënat e duhura nga burimet e jashtme të sillen në koleksionin e dëshiruar brenda bazës së të dhënave MongoDB.

Kjo pjesë e kodit MongoDB përfshin disa faza të një pipeline të agregimit për të përftuar informacionin e kërkuar. Le të komentojmë secilën fazë me më shumë detaje:   
***Stage 1: Përzgjedhim të gjitha provincat ku country = “kanada”***

**{ Country: "CDN" }**



**Figura 12:** Pamje nga Stage 1

Në këtë fazë, ne përzgjedhim të gjitha dokumentet nga koleksioni i provincave ku vendi është Kanada, duke përdorur kodin e vendit "CDN". Ky është një filtër që siguron që ne po punojmë vetëm me të dhënat relevante për Kanadanë.

***Stage 2: “lookup” për të bashkuar koleksionin e lumenjve***

**{ from: "located", localField: "CapProv", foreignField: "Province", as: "Rivers" }**

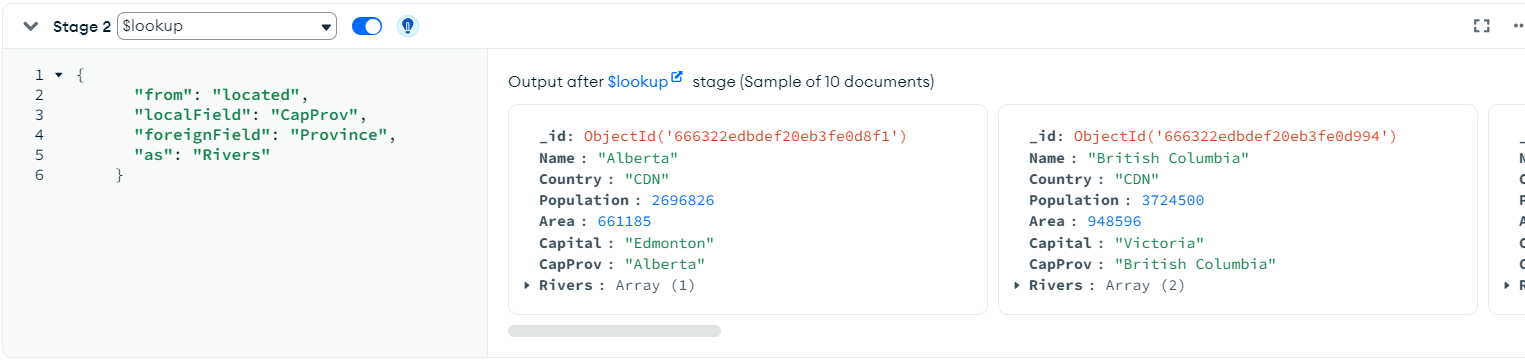


Figura 13 : Pamje nga Stage 2

Kjo fazë përdor operatorin **“$lookup”** për të bashkuar të dhënat nga koleksioni "located" (që përmban informacion për lumenjtë) me koleksionin aktual të provincave. Bashkimi bazohet në fushën **“CapProv”** nga dokumenti aktual të provincës dhe fushën **“Province”** nga koleksioni "located". Rezultati i bashkimit ruhet në një fushë të re të quajtur **“Rivers”.**

***Stage 3: Filtrimi i dokumenteve ku “Rivers” nuk është bosh***

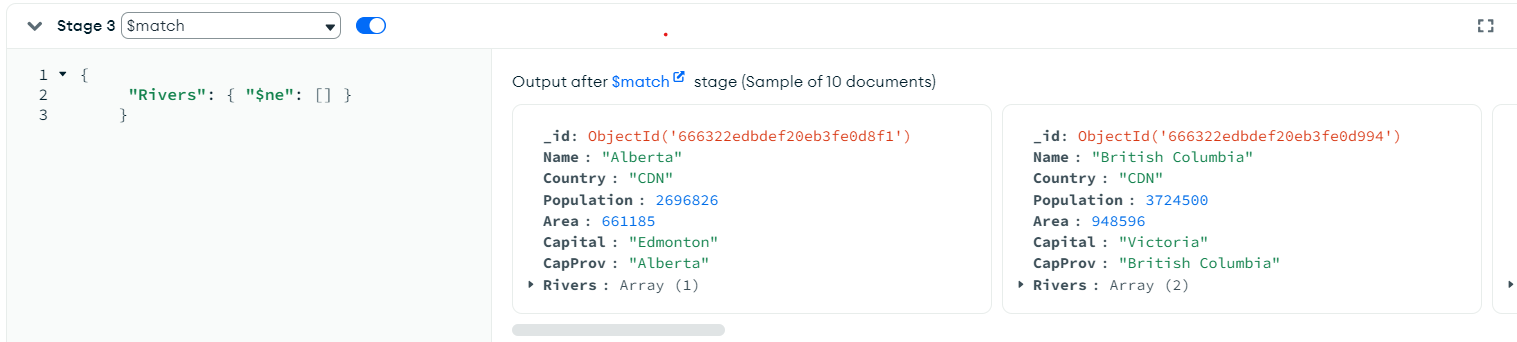
**{**

**Rivers: {**

**$ne: []**

**}**

**}**



**Figura 14:** Pamje nga Stage 3

Kjo fazë përdor operatorin **“$match”** për të filtruar dokumentet, duke siguruar që të përfshihen vetëm ato dokumente ku fusha **“Rivers”** nuk është bosh (pra, ekziston të paktën një lumë që kalon në provincën e përzgjedhur). Kjo fazë siguron që ne përfshijmë vetëm provincat që kanë lumenj.

***Stage 4: Projekti për të përzgjedhur fushat e nevojshme***

{

Capital: "$Capital",

Population: "$Population",

Province: "$CapProv",

\_id: 0

}



**Figura 15:** Pamja nga Stage 4

Në këtë fazë, ne përdorim operatorin **“$project”** për të përzgjedhur dhe formatuar fushat që duam të shfaqim në rezultatin përfundimtar. Këto fusha përfshijnë:

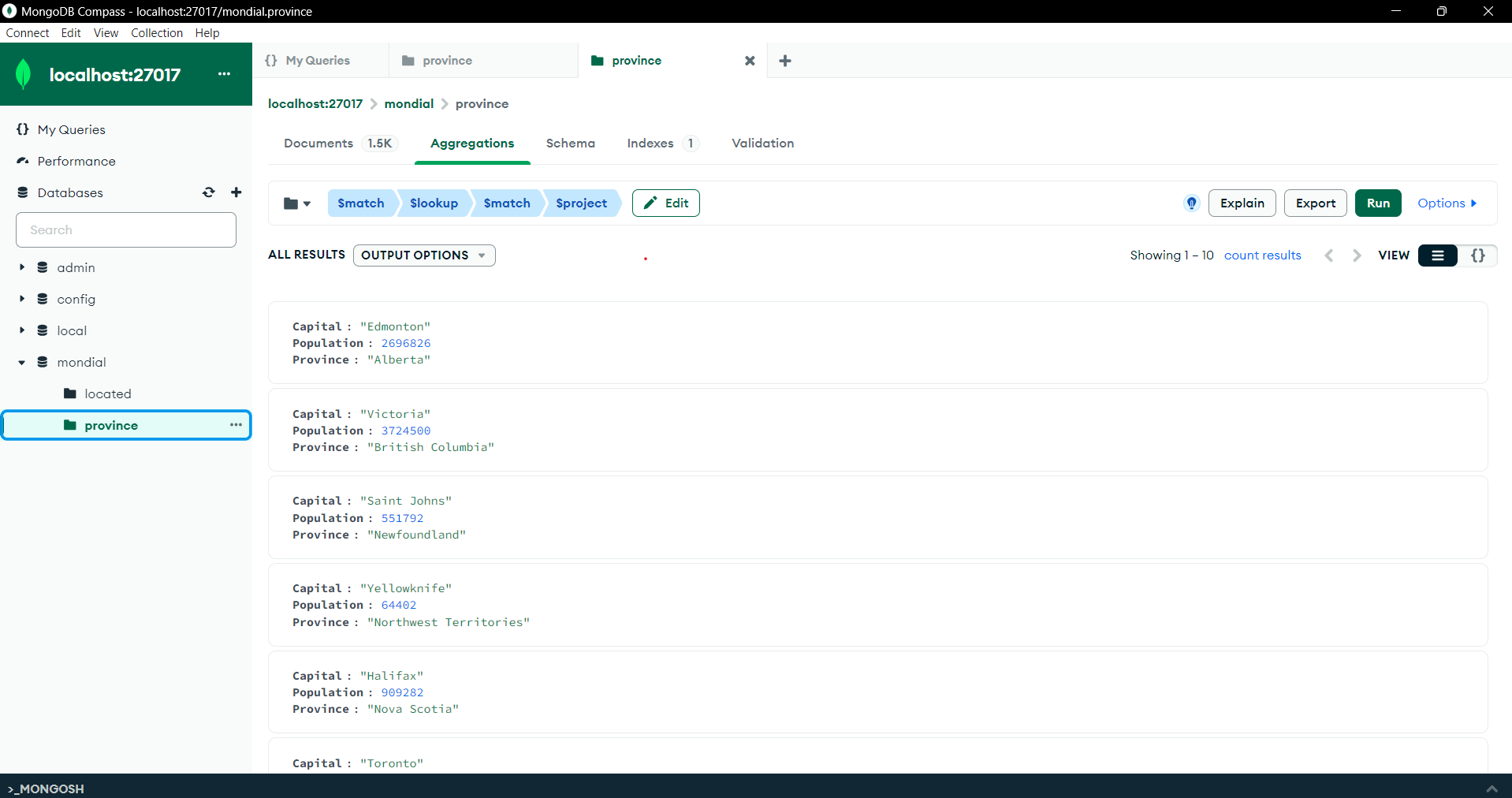
**“Capital”:** Emri i kryeqytetit.

**“Population”:** Numri i popullsisë së kryeqytetit.

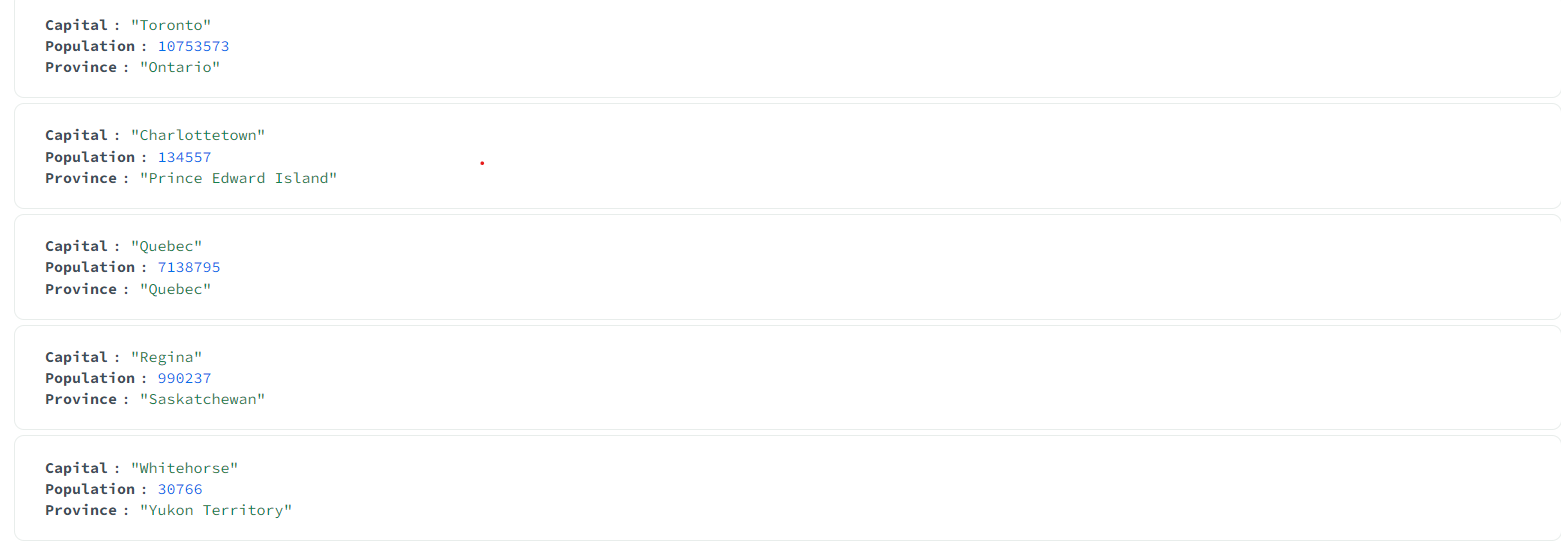
**“Province”:** Emri i provincës.

**“\_id”:** Përcaktohet si 0 për të përjashtuar fushën **“\_id”** nga rezultatet e përfundimtare.

Kjo fazë përfundimtare siguron që rezultati të përmbajë vetëm informacionin e kërkuar, të formatuar në mënyrë të qartë dhe të lexueshme. Në përmbledhje, këto faza së bashku krijojnë një pipeline të fuqishëm të agregimit që përzgjedh, bashkon, filtron dhe projekton të dhënat për të arritur rezultatin e kërkuar: një listë të emrave të kryeqyteteve të provincave të Kanadasë që përfshin numrin e popullsisë dhe faktin që në to kalon të paktën një lumë.



**Figura 16:** Ofron një pamje të rezultateve të përftuara pas ekzekutimit të pipeline-it të agregimit në editorin e MongoDB-së. Kjo pamje na lejon të shohim të dhënat e përzgjedhura dhe të formatuara sipas kërkesave të projektit tonë.



**Figura 17:** Vazhdimi i paraqitjës së rezultatëve

### Shpjegim i detajuar i pamjes

Pamja e rezultatit në editorin e MongoDB-së përmban të dhënat e përpunuara nga pipeline-i ynë i agregimit, të cilat përfshijnë:

1. **Capital**: Emri i kryeqytetit të secilës provincë të Kanadasë ku kalon të paktën një lumë.
2. **Population**: Numri i popullsisë për kryeqytetin përkatës.
3. **Province**: Emri i provincës përkatëse.

### Përfitimet dhe rëndësia e kësaj pamjeje

1. **Verifikimi i rezultateve**: Pamja e rezultateve na lejon të verifikojmë saktësinë dhe përputhshmërinë e të dhënave të përftuara me kërkesat e projektit. Ne mund të sigurohemi që të dhënat përmbajnë vetëm ato provinca të Kanadasë ku kalon të paktën një lumë dhe që përfshijnë numrin e popullsisë së kryeqytetit.
2. **Analiza e të dhënave**: Të dhënat e strukturuara në këtë mënyrë na mundësojnë të analizojmë më lehtë dhe të nxjerrim përfundime të vlefshme rreth kryeqyteteve dhe provincave të Kanadasë, duke parë drejtpërdrejt informacionin e rëndësishëm si popullsia dhe emrat e kryeqyteteve.
3. **Lehtësia e përdorimit**: Editorët si MongoDB Compass ofrojnë një ndërfaqe të thjeshtë dhe të përdorshme për të parë rezultatet e query-ve. Kjo na ndihmon të identifikojmë dhe korrigjojmë gabimet, nëse ka të tilla, dhe të përmirësojmë query-t tona në bazë të rezultateve të shfaqura.

Në përmbledhje, **Figura 17** paraqet rezultatet e një query-je komplekse të përpunuara në MongoDB, duke ofruar një pamje të qartë dhe të strukturuar të të dhënave që janë në përputhje me kërkesat e projektit tonë. Kjo pamje është një mjet i fuqishëm për verifikimin, analizën dhe interpretimin e të dhënave të marra.

# **Konkluzione**

Në konkluzion, puna jonë në manipulimin e të dhënave të mëdha ka përdorur teknologjitë e përshtatshme si BASEX për të punuar me të dhënat XML dhe MongoDB Compass për të manipuluar të dhënat NoSQL në MongoDB. Duke përdorur këto mjete, kemi arritur të zhvillojmë query-t e nevojshme dhe të manipulojmë të dhënat në një mënyrë efektive dhe të eficient. Rezultatet e arritura janë përpunimi i suksesshëm i të dhënave të mëdha në dy kontekste të ndryshme, duke përfshirë migrimin e të dhënave nga një sistem në tjetrin dhe zhvillimin e query-ve për të përmbushur qëllimet e punës sonë. Përmbledhur, kemi arritur qëllimin e punës sonë duke përdorur mjete të përshtatshme për manipulimin dhe analizën e të dhënave të mëdha në kontekstin e punës sonë. Future work për këtë projekt mund të përfshijë zhvillimin e strategjive për të përmirësuar performancën e query-ve në këto mjete, si dhe eksplorimin e teknologjive të reja që mund të ofrojnë avantazhe shtesë në manipulimin e të dhënave të mëdha. Një tjetër fokus i punës së ardhshme mund të jetë zhvillimi i aplikacioneve të përshtatshme për analizën dhe vizualizimin e të dhënave të përpunuara në këto mjete.

# Referencat

(Basex - Google Search, n.-b. (2024, June 07). *https://www.google.com/*. Gjetur në search?sca\_esv=4ce04de13f7e18f6&q=basex&uds=ADvngMjDct2CgUgIScTEbnyqnWsPre\_v78NnIVv634RP-tnFgtcVNRx5KPkixs-ettvBmhfmtoO0HaRdm8Ij7pjpoxlA0RbzRq5Ea9F3F4QGkJfK4g9JUIBm\_-BgRUXmL2FD\_krhtIjFbsGo6ftbxT92zxsrFnOi8f1RQiPfSKnzAOmQIhF3sJ8lhwifiiRO8y2sfGp-Y0uFjhCIDs9: https://www.google.com/search?sca\_esv=4ce04de13f7e18f6&q=basex&uds=ADvngMjDct2CgUgIScTEbnyqnWsPre\_v78NnIVv634RP-tnFgtcVNRx5KPkixs-ettvBmhfmtoO0HaRdm8Ij7pjpoxlA0RbzRq5Ea9F3F4QGkJfK4g9JUIBm\_-BgRUXmL2FD\_krhtIjFbsGo6ftbxT92zxsrFnOi8f1RQiPfSKnzAOmQIhF3sJ8lhwif

(Mongodb Composer Editor - Google Search, n. (2024, June 07). *https://www.google.com/search?q=mongodb+composer+editor*. Gjetur në &sca\_esv=4ce04de13f7e18f6&udm=2&prmd=visnbz&source=lnms&sa=X&ved=2ahUKEwi10deK6cmGAxUeVfEDHX9lI8oQ0pQJegQIAxAG&biw=1600&bih=754&dpr=1.6#vhid=nqGaHHSy5VWUWM&vssid=mosaic: https://www.google.com/search?q=mongodb+composer+editor&sca\_esv=4ce04de13f7e18f6&udm=2&prmd=visnbz&source=lnms&sa=X&ved=2ahUKEwi10deK6cmGAxUeVfEDHX9lI8oQ0pQJegQIAxAG&biw=1600&bih=754&dpr=1.6#vhid=nqGaHHSy5VWUWM&vssid=mosaic

database, T. M. (a.d.). *(n.d.). https://www.dbis.informatik.uni-goettingen.de/Mondial/*.