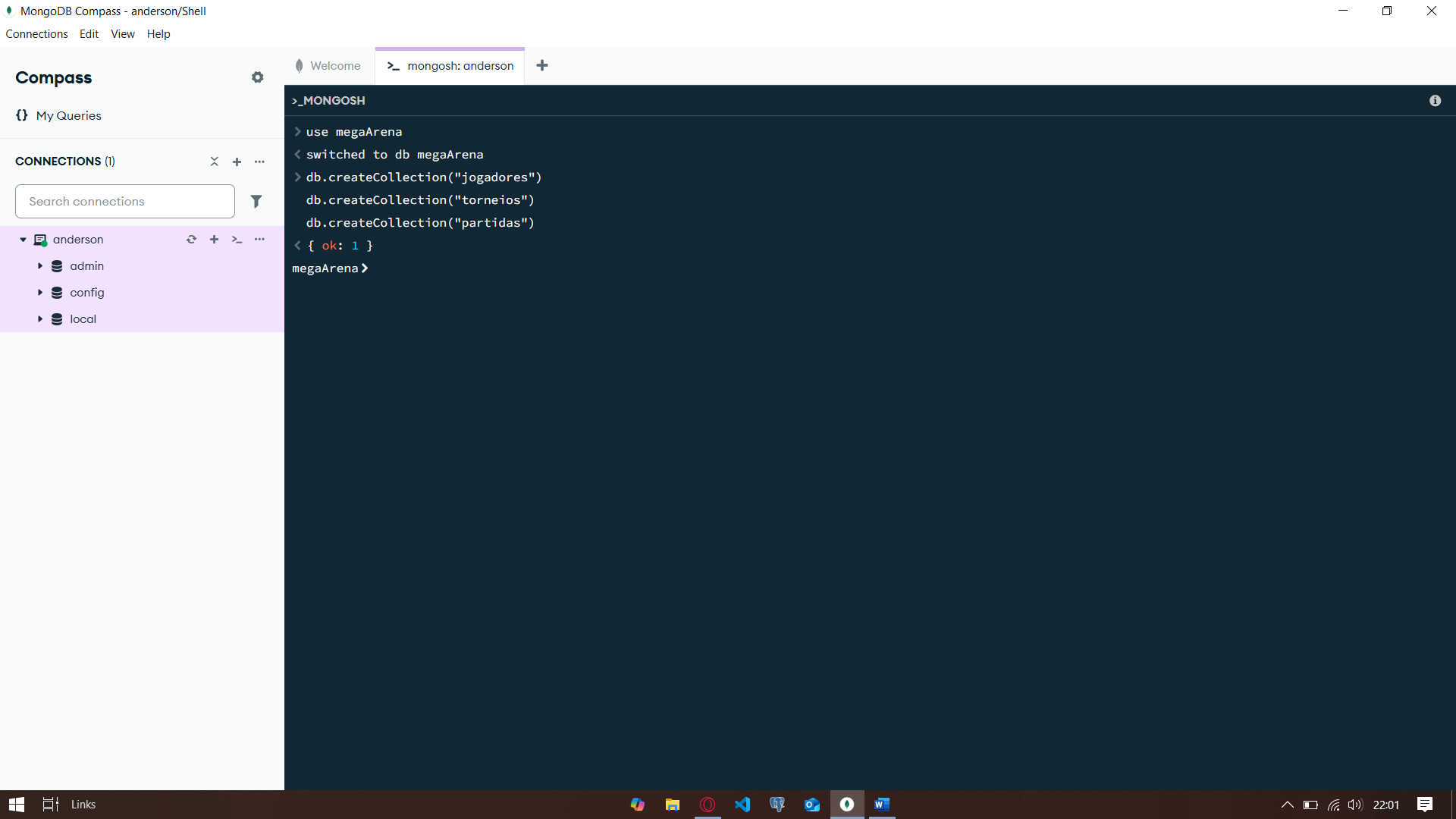
Questão 1 – Criação do Banco Crie o banco de dados megaArena com as coleções jogadores, torneios e partidas.  
  


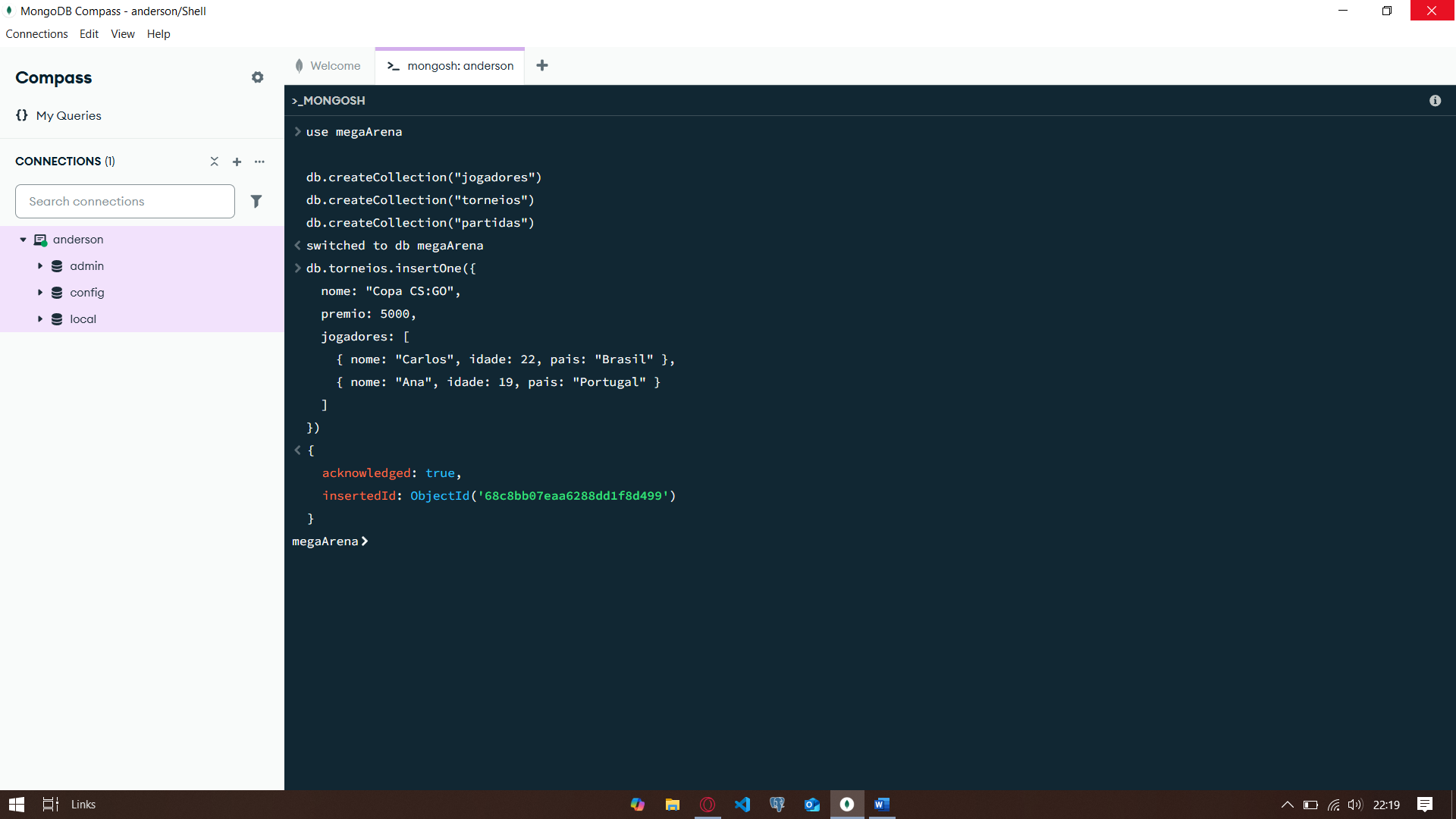
use megaArena

db.createCollection("jogadores")

db.createCollection("torneios")

db.createCollection("partidas")

Questão 2 – Embedding Modele uma versão alternativa onde os jogadores inscritos em um torneio fiquem incorporados diretamente dentro do documento torneio.



db.torneios.insertOne({

nome: "Copa CS:GO",

premio: 5000,

jogadores: [

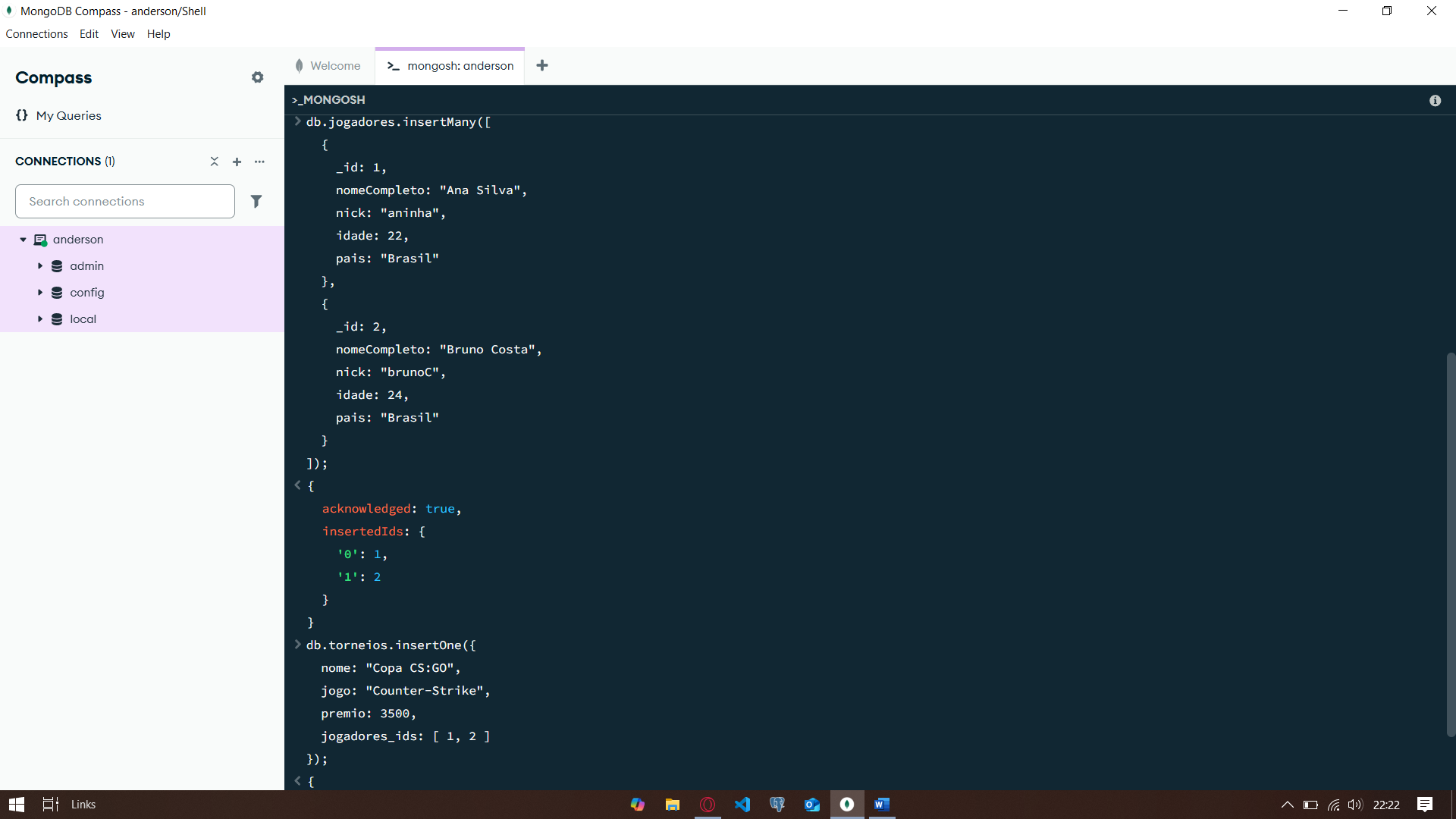
{ nome: "Carlos", idade: 22, pais: "Brasil" },

{ nome: "Ana", idade: 19, pais: "Portugal" }

]

})

Questão 3 – Referencing Modele uma versão alternativa onde cada torneio guarda apenas os IDs dos jogadores inscritos, mantendo-os em coleção separada.



db.jogadores.insertMany([

{

\_id: 1,

nomeCompleto: "Ana Silva",

nick: "aninha",

idade: 22,

pais: "Brasil"

},

{

\_id: 2,

nomeCompleto: "Bruno Costa",

nick: "brunoC",

idade: 24,

pais: "Brasil"

}

]);

db.torneios.insertOne({

nome: "Copa CS:GO",

jogo: "Counter-Strike",

premio: 3500,

jogadores\_ids: [ 1, 2 ]

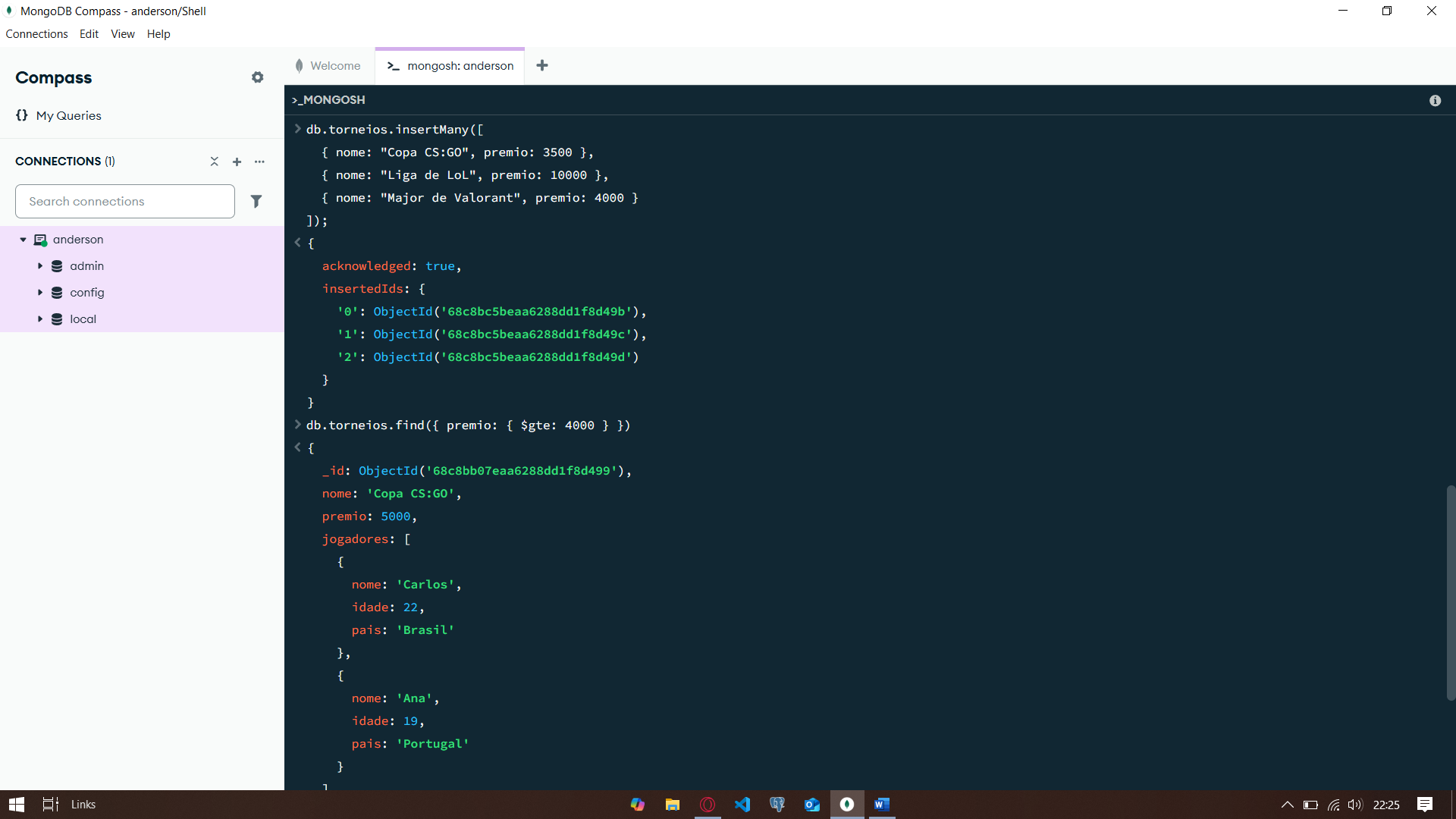
});

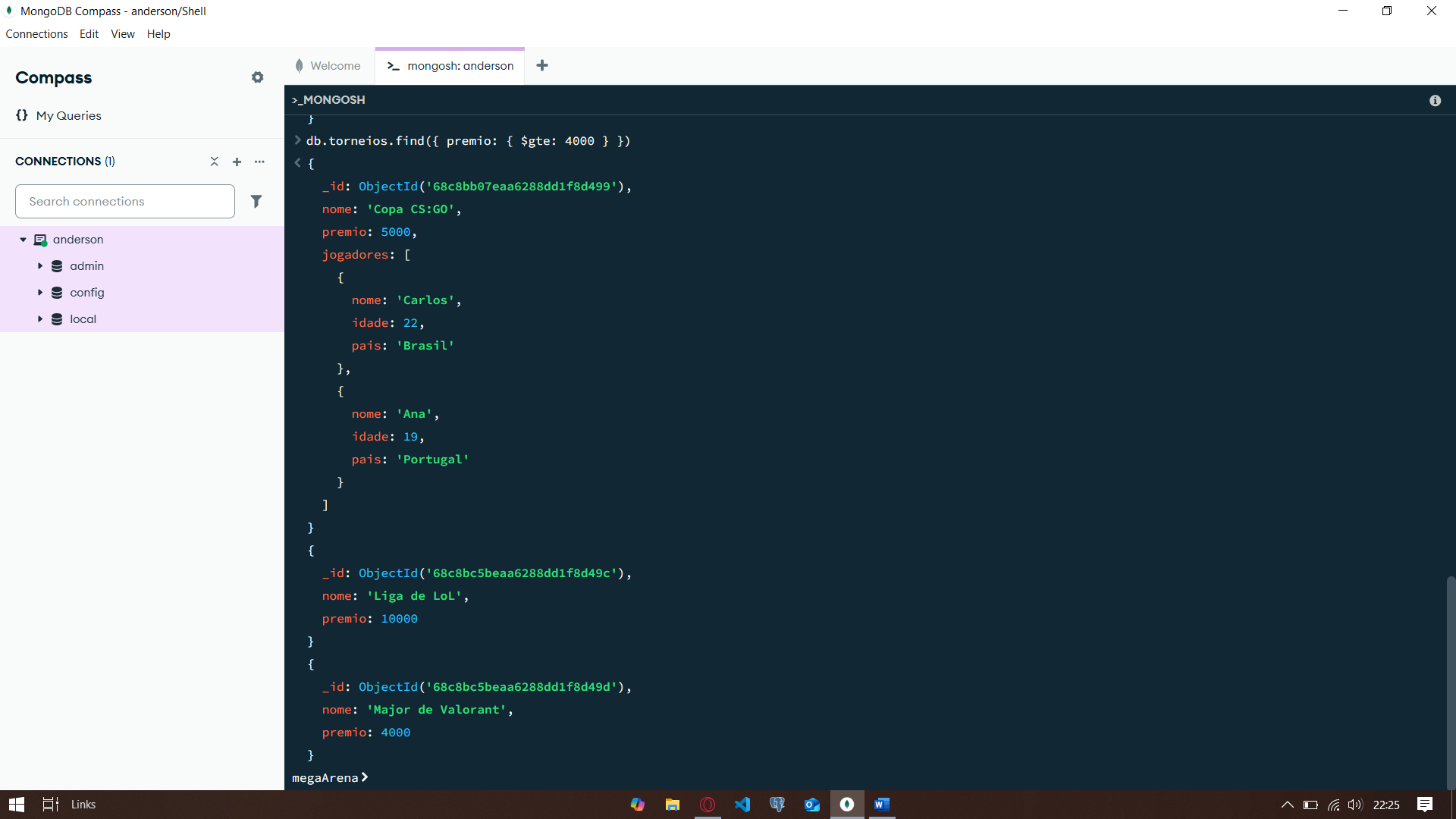
Questão 4 – Justificativa de Modelagem Explique em qual situação é melhor usar embedding e em qual é melhor usar referencing no cenário da MegaArena.

* **Embedding (Incorporação):**
  + **O quê é?** Colocar tudo junto em um único documento. O torneio e a lista completa de jogadores ficam no mesmo lugar.
  + **Quando usar?** Para torneios pequenos e únicos. É mais rápido de ler, pois você pega tudo de uma vez.
* **Referencing (Referência):**
  + **O quê é?** Manter as coisas separadas. Os jogadores ficam em uma coleção, e o torneio apenas anota o "ID" de quem está participando.
  + **Quando usar?** Para torneios grandes ou quando os mesmos jogadores participam de vários eventos. É melhor para manter os dados corretos, pois se o nick de um jogador mudar, você só altera em um lugar.

**Em resumo:**

* **Juntar tudo (Embedding)** = Mais rápido para ler.
* **Manter separado (Referencing)** = Mais fácil de atualizar e mais organizado.

Questão 5 – Consulta Liste todos os torneios cujo prêmio seja maior ou igual a 4000.  
  




db.torneios.insertMany([

{ nome: "Copa CS:GO", premio: 3500 },

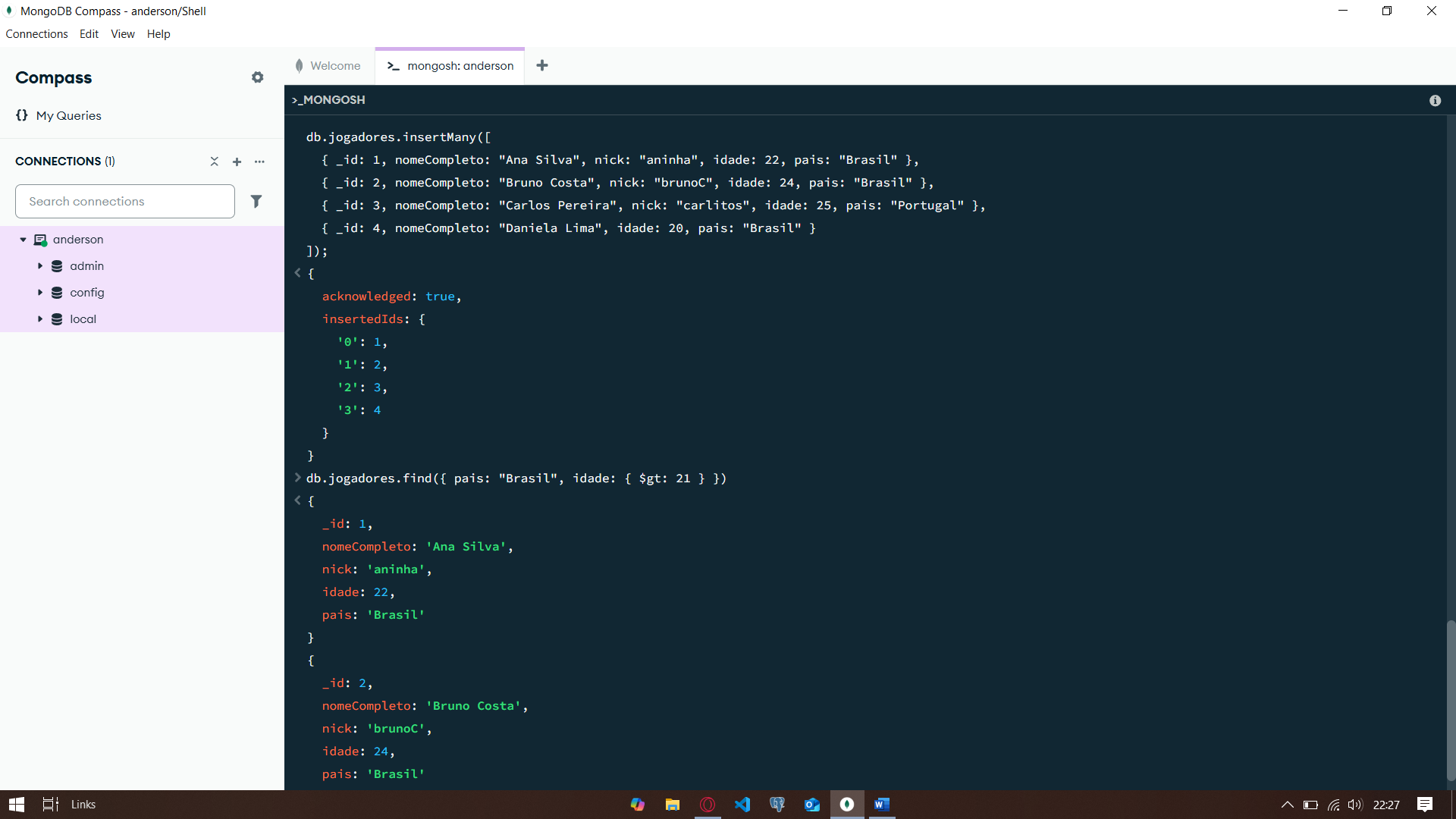
{ nome: "Liga de LoL", premio: 10000 },

{ nome: "Major de Valorant", premio: 4000 }

]);

db.torneios.find({ premio: { $gte: 4000 } })

Questão 6 – Consulta Avançada com $and Liste os jogadores do Brasil com idade maior que 21 anos



db.jogadores.deleteMany({});

db.jogadores.insertMany([

{ \_id: 1, nomeCompleto: "Ana Silva", nick: "aninha", idade: 22, pais: "Brasil" },

{ \_id: 2, nomeCompleto: "Bruno Costa", nick: "brunoC", idade: 24, pais: "Brasil" },

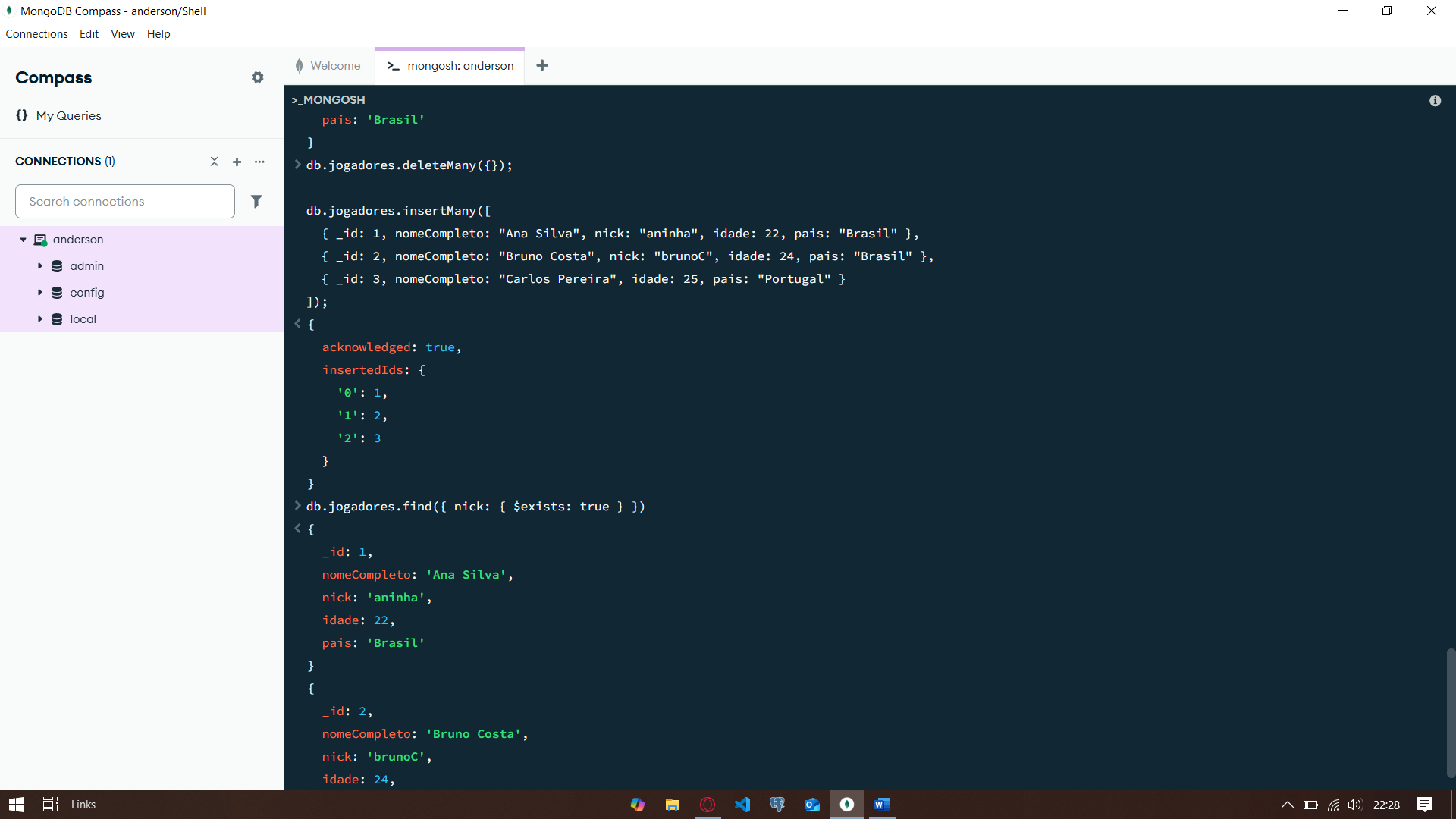
{ \_id: 3, nomeCompleto: "Carlos Pereira", nick: "carlitos", idade: 25, pais: "Portugal" },

{ \_id: 4, nomeCompleto: "Daniela Lima", idade: 20, pais: "Brasil" }

]);

db.jogadores.find({ pais: "Brasil", idade: { $gt: 21 } })

Questão 7 – Operador $exists Liste todos os jogadores que já possuem o campo nick.



db.jogadores.deleteMany({});

db.jogadores.insertMany([

{ \_id: 1, nomeCompleto: "Ana Silva", nick: "aninha", idade: 22, pais: "Brasil" },

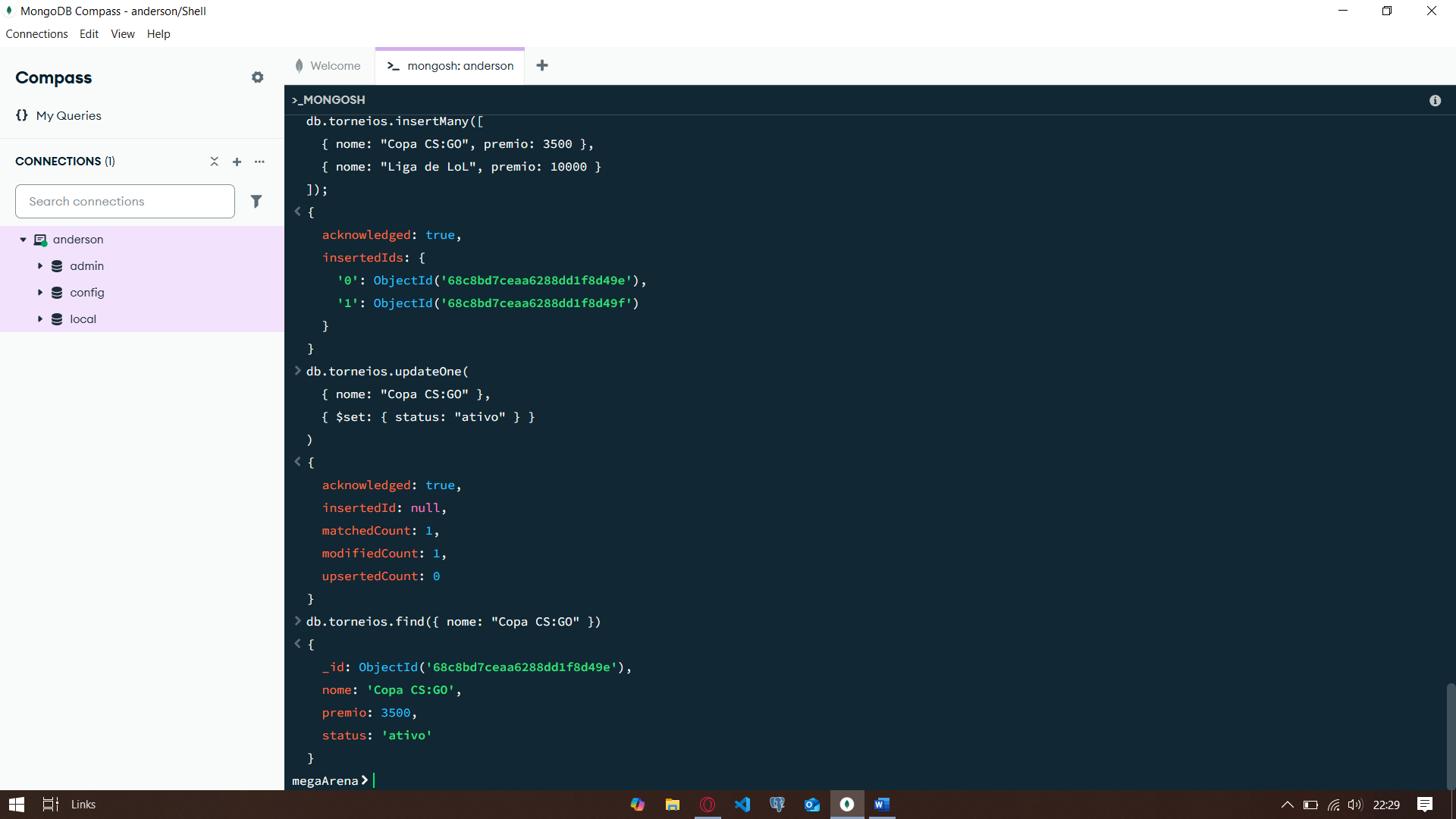
{ \_id: 2, nomeCompleto: "Bruno Costa", nick: "brunoC", idade: 24, pais: "Brasil" },

{ \_id: 3, nomeCompleto: "Carlos Pereira", idade: 25, pais: "Portugal" }

]);

db.jogadores.find({ nick: { $exists: true } })

Questão 8 – Atualização com $set Atualize o torneio “Copa CS:GO” adicionando o campo status: "ativo"



db.torneios.deleteMany({});

db.torneios.insertMany([

{ nome: "Copa CS:GO", premio: 3500 },

{ nome: "Liga de LoL", premio: 10000 }

]);

db.torneios.updateOne(

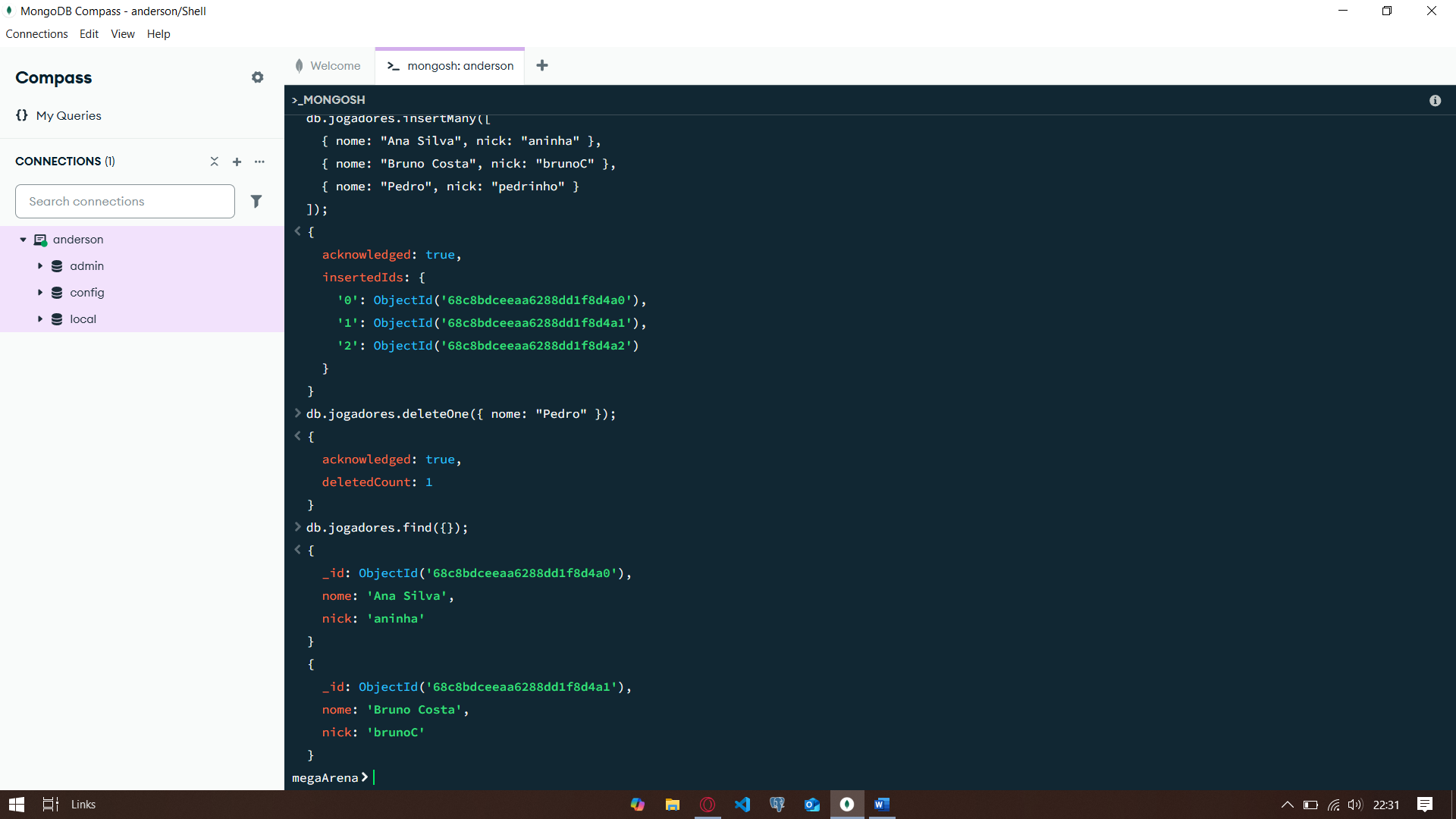
{ nome: "Copa CS:GO" },

{ $set: { status: "ativo" } }

)

db.torneios.find({ nome: "Copa CS:GO" })

Questão 9 – Exclusão Remova o jogador com nome Pedro



db.jogadores.deleteMany({});

db.jogadores.insertMany([

{ nome: "Ana Silva", nick: "aninha" },

{ nome: "Bruno Costa", nick: "brunoC" },

{ nome: "Pedro", nick: "pedrinho" }

]);

db.jogadores.deleteOne({ nome: "Pedro" });

db.jogadores.find({});

Questão 10 – Paginação Liste os torneios ordenados pela premiação (decrescente), exibindo apenas o segundo torneio mais bem pago.

db.torneios.deleteMany({});

db.torneios.insertMany([

{ nome: "Torneio Bronze", premio: 1000 },

{ nome: "Copa Prata", premio: 5000 },

{ nome: "Liga Ouro", premio: 10000 },

{ nome: "Major Diamante", premio: 8000 }

]);

db.torneios.find().sort({ premio: -1 }).skip(1).limit(1)

