

Seminární práce z předmětu NoSQL databáze

7 Dotazy

Tato kapitola slouží pro popis použitých dotazů nad kolekcemi.

7.1 Healthcare

Tato kapitola slouží k popisu dotazů použitých nad kolekcí healthcare

7.1.1 Práce s daty – Insert, Update, Delete

Vložení nového pacienta do databáze:

```
db.healthcare.insertOne({"Name": "Jan Novak", "Age": 45, "Gender":  
"Male", "Blood Type": "O+", "Medical Condition": "Hypertension", "Date of  
Admission": new ISODate("2024-03-01T00:00:00Z"), "Doctor": "Dr. Pepa",  
"Hospital": "Motol", "Insurance Provider": "VZP", "Billing Amount": new  
Double(15000), "Room Number": 300, "Admission Type": "Emergency",  
"Discharge Date": new ISODate("2024-03-10T00:00:00Z"), "Medication":  
"Ibalgin", "Test Results": "Normal"})
```

Návratová hodnota:

```
{  
  acknowledged: true,  
  insertedId: ObjectId('67eae447093771487e6b1413')  
}
```

Aktualizace diagnózy pacienta:

```
db.healthcare.updateOne({"Name": "Jan Novak"}, {$set: { "Medical  
Condition": "Diabetes"}}
```

Návratová hodnota:

```
{  
  acknowledged: true,  
  insertedId: null,  
  matchedCount: 1,  
  modifiedCount: 0,  
  upsertedCount: 0  
}
```

Smazání pacienta, který byl propuštěn před rokem:

```
db.healthcare.deleteOne({"Discharge Date": {$lt: new ISODate("2024-03-  
31T00:00:00Z")}})
```

Návratová hodnota:

```
{ acknowledged: true, deletedCount: 1 }
```

Vložení pacienta, který nemá kompletní informace – test validačního schéma:

```
db.healthcare.insertOne({"Name": "Jan Novak", "Age": 45, "Gender":  
"Male", "Blood Type": "O+", "Medical Condition": "Hypertension", "Date of  
Admission": new ISODate("2024-03-01T00:00:00Z"), "Doctor": "Dr. Pepa",  
"Hospital": "Motol", "Insurance Provider": null, "Billing Amount": new  
Double(15000), "Room Number": 300, "Admission Type": "Emergency",  
"Discharge Date": new ISODate("2024-03-10T00:00:00Z"), "Medication":  
"Ibalgin", "Test Results": "Normal"})
```

Návratová hodnota:

Seminární práce z předmětu NoSQL databáze

```
Uncaught:
MongoServerError: Document failed validation
Additional information: {
  failingDocumentId: ObjectId('67eae6ba093771487e6b1415'),
  details: {
    operatorName: '$jsonSchema',
    schemaRulesNotSatisfied: [
      {
        operatorName: 'properties',
        propertiesNotSatisfied: [
          {
            propertyName: 'Insurance Provider',
            description: 'Insurance provider covering the patient,
required.',
            details: [
              {
                operatorName: 'bsonType',
                specifiedAs: { bsonType: 'string' },
                reason: 'type did not match',
                consideredValue: null,
                consideredType: 'null'
              }
            ]
          }
        ]
      }
    ]
  }
}
```

Aktualizace pacienta, který není v DB – příkaz v pořádku, ale nic není aktualizováno:

```
db.healthcare.updateOne({"Name": "Jan Novak"}, {$set: {"Blood
Type": "A"}})
```

Návratová hodnota:

```
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 0,
  modifiedCount: 0,
  upsertedCount: 0
}
```

Hromadná aktualizace pacientů hospitalizovaných déle než 30 dnů:

```
db.healthcare.updateMany({$expr: {$gte: [{ $subtract: ["$Discharge Date",
"$Date of Admission"] }, 1000 * 60 * 60 * 24 * 30]}}, {$set: {"Billing
Amount": new Double(500000)}})
```

Návratová hodnota:

```
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1874,
  modifiedCount: 1874,
  upsertedCount: 0
}
```

Seminární práce z předmětu NoSQL databáze

7.1.2 Agregací funkce

Počet pacientů podle krevní skupiny:

```
db.healthcare.aggregate([{$group: {_id: "$Blood Type", count: {$sum: 1}}}]])
```

Návratová hodnota:

```
[
  { _id: 'O-', count: 6877 },
  { _id: 'B-', count: 6944 },
  { _id: 'B+', count: 6945 },
  { _id: 'AB-', count: 6945 },
  { _id: 'AB+', count: 6947 },
  { _id: 'O+', count: 6917 },
  { _id: 'A-', count: 6968 },
  { _id: 'A+', count: 6956 }
]
```

Průměrný věk pacientů v nemocnici:

```
db.healthcare.aggregate([{$group: {_id: null, avgAge: {$avg: "$Age"}}}])
```

Návratová hodnota:

```
[ { _id: null, avgAge: 51.539018721058035 } ]
```

Nejvyšší účty za hospitalizaci:

```
db.healthcare.aggregate([{$sort: {"Billing Amount": -1}}, {$limit: 3 }])
```

Návratová hodnota:

```
[
  {
    _id: ObjectId('67eadd86946b92272a296dbf'),
    Name: 'tOdd CARrILlO',
    Age: 51,
    Gender: 'Female',
    'Blood Type': 'A+',
    'Medical Condition': 'Hypertension',
    'Date of Admission': ISODate('2023-09-08T00:00:00.000Z'),
    Doctor: 'Kathleen Griffin',
    Hospital: 'Griffin Group',
    'Insurance Provider': 'Blue Cross',
    'Billing Amount': 52764.2767364692,
    'Room Number': 209,
    'Admission Type': 'Elective',
    'Discharge Date': ISODate('2023-10-04T00:00:00.000Z'),
    Medication: 'Ibuprofen',
    'Test Results': 'Normal'
  },
  {
    _id: ObjectId('67eadd86946b92272a29a830'),
    Name: 'kARen klInE',
    Age: 74,
    Gender: 'Female',
    'Blood Type': 'AB+',
    'Medical Condition': 'Cancer',
    'Date of Admission': ISODate('2021-06-19T00:00:00.000Z'),
    Doctor: 'Dr. Joseph Gordon',
    Hospital: 'Hernandez-Morton',
    'Insurance Provider': 'UnitedHealthcare',
  }
]
```

Seminární práce z předmětu NoSQL databáze

```
'Billing Amount': 52373.0323742418,
'Room Number': 181,
'Admission Type': 'Urgent',
'Discharge Date': ISODate('2021-07-03T00:00:00.000Z'),
Medication: 'Penicillin',
'Test Results': 'Inconclusive'
},
{
  _id: ObjectId('67eadd86946b92272a2966e2'),
  Name: 'kARen klInE',
  Age: 79,
  Gender: 'Female',
  'Blood Type': 'AB+',
  'Medical Condition': 'Cancer',
  'Date of Admission': ISODate('2021-06-19T00:00:00.000Z'),
  Doctor: 'Dr. Joseph Gordon',
  Hospital: 'Hernandez-Morton',
  'Insurance Provider': 'UnitedHealthcare',
  'Billing Amount': 52373.0323742418,
  'Room Number': 181,
  'Admission Type': 'Urgent',
  'Discharge Date': ISODate('2021-07-03T00:00:00.000Z'),
  Medication: 'Penicillin',
  'Test Results': 'Inconclusive'
}
]
```

Počet hospitalizací podle typu přijetí:

```
db.healthcare.aggregate([{$group: {_id: "$Admission Type", count: {$sum: 1}}}}])
```

Návratová hodnota:

```
[
  { _id: 'Emergency', count: 18268 },
  { _id: 'Elective', count: 18655 },
  { _id: 'Urgent', count: 18576 }
]
```

Celkové náklady na hospitalizaci dle nemocnice:

```
db.healthcare.aggregate([{$group: {_id: "$Hospital",
averageBillingAmount: {$avg: "$Billing Amount"}}}}])
```

Návratová hodnota:

```
[
  { _id: 'and Wolf Sons', averageBillingAmount: 31070.013970411303 },
  { _id: 'Sanchez-Morris', averageBillingAmount: 28317.7126961858 },
  { _id: 'Mack Sons and', averageBillingAmount: 47524.3223647636 },
  {
    _id: 'Grant, and Wood Larsen',
    averageBillingAmount: 27319.4804305011
  },
  {
    _id: 'Miller Montgomery Allen, and',
    averageBillingAmount: 8519.0043811014
  },
  { _id: 'Patrick-Vaughn', averageBillingAmount: 36487.948616563 },
  { _id: 'Everett PLC', averageBillingAmount: 20900.4113271649 },
  {
    _id: 'and Lewis Cohen, Buck',

```

Seminární práce z předmětu NoSQL databáze

```
averageBillingAmount: 34955.0803788092
},
{
  _id: 'Finley and Vargas, Gutierrez',
  averageBillingAmount: 42098.4661381742
},
{ _id: 'Martin-Rosario', averageBillingAmount: 34903.32532242053 },
{ _id: 'LLC Keith', averageBillingAmount: 6026.32743841605 },
{
  _id: 'and Herring Clark, Dorsey',
  averageBillingAmount: 47780.3629878701
},
{ _id: 'Rojas-Allen', averageBillingAmount: 48768.6881735947 },
{
  _id: 'and Roberts Martinez, Ramirez',
  averageBillingAmount: 9038.4581657275
},
{ _id: 'Group Phillips', averageBillingAmount: 31201.306454375124 },
{ _id: 'Carroll-Crawford', averageBillingAmount: 45929.0208207266 },
{ _id: 'Group Bryant', averageBillingAmount: 32530.91904622947 },
{ _id: 'Wheeler-Owens', averageBillingAmount: 32571.6555413477 },
{ _id: 'Novak-Gutierrez', averageBillingAmount: 30796.3062789342 },
{ _id: 'Finley-Snyder', averageBillingAmount: 7826.5511363074 }
]
```

Počet unikátních diagnóz mezi pacienty:

```
db.healthcare.aggregate([{$group: {_id: "$Medical Condition"}}, {$count:
"Unique Diagnoses"}])
```

Návratová hodnota:

```
[ { 'Unique Diagnoses': 6 } ]
```

7.2 Pokémon

Tato kapitola slouží k popisu dotazů použitých nad kolekcí pokemon.

7.2.1 Práce s indexy a optimalizace

Nalezení Pokémona před přidáním indexu:

```
db.pokemon.find({"Name": "Pikachu"}).explain("executionStats");
```

Doba hledání:

```
executionTimeMillis: 7
```

Přidání indexu:

```
db.pokemon.createIndex({ "Type 1": 1, Attack: -1 })
```

Doba hledání:

```
executionTimeMillis: 3
```

Vytvoření složeného indexu:

```
db.pokemon.createIndex({ "Type 1": 1, Attack: -1 })
```

Návratová hodnota:

```
Type 1_1_Attack_-1
```

Testování dotazu s indexem a bez indexu:

Seminární práce z předmětu NoSQL databáze

```
db.pokemon.find({ Name: "Pikachu" }).explain("executionStats")
```

Návratová hodnota:

```
{
  Explain...
}
```

Použití funkce hint k vynucení indexu:

```
db.pokemon.createIndex({"Type 1": 1 })
db.pokemon.find({ "Type 1": "Electric" }).hint({ "Type 1": 1 })
```

Návratová hodnota:

```
[...
  {
    _id: ObjectId('67ec03cd62a2ade3c8a20a29'),
    '#': 310,
    Name: 'ManectricMega Manectric',
    'Type 1': 'Electric',
    'Type 2': '',
    Total: 575,
    HP: 70,
    Attack: 75,
    Defense: 80,
    Sp: { 'Atk': 135, 'Def': 80 },
    Speed: 135,
    Generation: 3,
    Legendary: 'False'
  }
]
```

Odstranění indexu:

```
db.pokemon.dropIndex("Name_1")
```

Návratová hodnota:

```
{
  nIndexesWas: 4,
  ok: 1,
  raw: {
    'shard-primary/shard-primary:27017,shard-secondary:27017,shard-tertiary:27017': { nIndexesWas: 4, ok: 1 }
  },
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1743528103, i: 1 }),
    signature: {
      hash: Binary.createFromBase64('p00sNjpCpgEY5155t+F2CyE/rWg=', 0),
      keyId: Long('7488364313384583179')
    }
  },
  operationTime: Timestamp({ t: 1743528103, i: 1 })
}
```

Agregační funkce s indexem:

```
db.pokemon.aggregate([ { $match: { "Type 1": "Fire" } }, { $group: { _id: "$Type 1", avgAttack: { $avg: "$Attack" } } } ])
```

Návratová hodnota:

```
[ { _id: 'Fire', avgAttack: 84.76923076923077 } ]
```

Seminární práce z předmětu NoSQL databáze

7.2.2 Replikace, sharding a odolnost proti výpadkům

Kontrola replikace:

```
sh.status()
```

Návratová hodnota:

```
{
  ...
  {
    database: {
      _id: 'nosql_project',
      primary: 'shard-primary',
      version: {
        uuid: UUID('1cd84c31-9596-4b21-8b1c-927a683c177a'),
        timestamp: Timestamp({ t: 1743520711, i: 2 }),
        lastMod: 1
      }
    },
    collections: {}
  }
}
```

Vytvoření hashed indexu – pro distribuci napříč shardy:

```
db.pokemon.createIndex({ "Type 1": "hashed" })
```

Návratová hodnota:

Type 1_hashed

Vložení dat do sharded kolekce:

```
sh.shardCollection("nosql_project.pokemon", { "Type 1": "hashed" })
```

Návratová hodnota:

```
{
  collectionsharded: 'nosql_project.pokemon',
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1743529847, i: 6 }),
    signature: {
      hash: Binary.createFromBase64('QeGlyh3QpSJQ0KdoJJyToywq470=', 0),
      keyId: Long('7488364313384583179')
    }
  },
  operationTime: Timestamp({ t: 1743529847, i: 5 })
}
```

Kontrola funkčnosti:

```
sh.status()
```

Návratová hodnota:

```
{
  ...
  {
    database: {
      _id: 'nosql_project',
      primary: 'shard-primary',
      version: {
```

Seminární práce z předmětu NoSQL databáze

```
    uuid: UUID('1cd84c31-9596-4b21-8b1c-927a683c177a'),
    timestamp: Timestamp({ t: 1743520711, i: 2 }),
    lastMod: 1
  }
},
collections: {
  'nosql_project.pokemon': {
    shardKey: { 'Type 1': 'hashed' },
    unique: false,
    balancing: true,
    chunkMetadata: [ { shard: 'shard-primary', nChunks: 1 } ],
    chunks: [
      { min: { 'Type 1': MinKey() }, max: { 'Type 1': MaxKey() }, 'on
shard': 'shard-primary', 'last modified': Timestamp({ t: 1, i: 0 }) }
    ],
    tags: []
  }
}
}
```

Vyhledání dat napříč shardy:

```
db.pokemon.find({ "Type 1": "Water" })
```

Návratová hodnota:

```
[
  {
    _id: ObjectId('67ec03cd62a2ade3c8a20941'),
    '#': 99,
    Name: 'Kingler',
    'Type 1': 'Water',
    'Type 2': '',
    Total: 475,
    HP: 55,
    Attack: 130,
    Defense: 115,
    Sp: { 'Atk': 50, 'Def': 50 },
    Speed: 75,
    Generation: 1,
    Legendary: 'False'
  }
]
```

Simulace výpadku uzlu:

Nutné vypnout primární/nebo jiný uzel:

```
docker compose down shard-primary
```

Poté lze testovat dostupnost dat:

```
db.pokemon.find().limit(1)
```

Návratová hodnota:

```
[
  {
    _id: ObjectId('67ec03cd62a2ade3c8a208d7'),
    '#': 2,
    Name: 'Ivysaur',
    'Type 1': 'Grass',
    'Type 2': 'Poison',
  }
]
```


Seminární práce z předmětu NoSQL databáze

```
Total: 405,  
HP: 60,  
Attack: 62,  
Defense: 63,  
Sp: { 'Atk': 80, 'Def': 80 },  
Speed: 60,  
Generation: 1,  
Legendary: 'False'  
}  
]
```

Kontrola replikace a “vypadlého uzlu”:

```
rs.status()
```

Návratová hodnota:

```
name: 'shard-primary:27017',  
  health: 0,  
  state: 8,  
  stateStr: '(not reachable/healthy)'  
  
name: 'shard-secondary:27017',  
  health: 1,  
  state: 2,  
  stateStr: 'SECONDARY',  
  
name: 'shard-tertiary:27017',  
  health: 1,  
  state: 1,  
  stateStr: 'PRIMARY'
```

Obnova výpadku:

Zapnutí vypadlého uzlu:

```
docker compose up shard-primary
```

Rekonfigurace replica setu:

```
rs.status()
```

Návratová hodnota:

```
name: 'shard-primary:27017',  
  health: 1,  
  state: 2,  
  stateStr: 'SECONDARY'
```

7.3 Video Games Sales

Tato kapitola slouží k popisu dotazů použitých nad kolekcí games.

7.3.1 Vyhledávání a filtrování dat

Vyhledání všech her vydaných po roce 2010:

```
db.games.find({"Year": {$gt: 2010}})
```

Návratová hodnota:

```
[...  
  {
```

Seminární práce z předmětu NoSQL databáze

```
_id: ObjectId('67ed37daf8bcb9ec6a807d3d'),
Rank: 74,
Name: 'Animal Crossing: New Leaf',
Platform: '3DS',
Year: 2012,
Genre: 'Simulation',
Publisher: 'Nintendo',
NA_Sales: 2.01,
EU_Sales: 2.32,
JP_Sales: 4.36,
Other_Sales: 0.41,
Global_Sales: 9.09
}
]
```

Vyhledání her podle konkrétní platformy – PS4

```
db.games.find({"Platform": "PS4"})
```

Návratová hodnota:

```
[...
{
  _id: ObjectId('67ed37daf8bcb9ec6a807e4a'),
  Rank: 352,
  Name: 'Tom Clancy's The Division',
  Platform: 'PS4',
  Year: 2016,
  Genre: 'Shooter',
  Publisher: 'Ubisoft',
  NA_Sales: 1.28,
  EU_Sales: 1.61,
  JP_Sales: 0.15,
  Other_Sales: 0.57,
  Global_Sales: 3.61
}
]
```

Vyhledání všech her žánru Action nebo Shooter:

```
db.games.find({Genre: {$in: ["Action", "Shooter"]}})
```

Návratová hodnota:

```
[...
{
  _id: ObjectId('67ed37daf8bcb9ec6a807d23'),
  Rank: 57,
  Name: 'Grand Theft Auto IV',
  Platform: 'PS3',
  Year: 2008,
  Genre: 'Action',
  Publisher: 'Take-Two Interactive',
  NA_Sales: 4.76,
  EU_Sales: 3.76,
  JP_Sales: 0.44,
  Other_Sales: 1.62,
  Global_Sales: 10.57
},
{
  _id: ObjectId('67ed37daf8bcb9ec6a807d29'),
  Rank: 62,
  Name: 'Call of Duty: Ghosts',

```

Seminární práce z předmětu NoSQL databáze

```
Platform: 'X360',
Year: 2013,
Genre: 'Shooter',
Publisher: 'Activision',
NA_Sales: 6.72,
EU_Sales: 2.63,
JP_Sales: 0.04,
Other_Sales: 0.82,
Global_Sales: 10.21
}
]
```

Vyhledání her s globálním prodejem vyšším než 50 m kusů

```
db.games.find({ "Global_Sales": { $gt: 50 } })
```

Návratová hodnota:

```
[
  {
    _id: ObjectId('67ed37daf8bcb9ec6a807cf2'),
    Rank: 1,
    Name: 'Wii Sports',
    Platform: 'Wii',
    Year: 2006,
    Genre: 'Sports',
    Publisher: 'Nintendo',
    NA_Sales: 41.49,
    EU_Sales: 29.02,
    JP_Sales: 3.77,
    Other_Sales: 8.46,
    Global_Sales: 82.74
  }
]
```

Vyhledání her mezi lety 2000 a 2005 na platformě PS2:

```
db.games.find({"Year": {$gte: 2000, $lte: 2005}, "Platform": "PS2"})
```

Návratová hodnota:

```
[...
  {
    _id: ObjectId('67ed37daf8bcb9ec6a807dde'),
    Rank: 242,
    Name: 'Spider-Man: The Movie',
    Platform: 'PS2',
    Year: 2002,
    Genre: 'Action',
    Publisher: 'Activision',
    NA_Sales: 2.71,
    EU_Sales: 1.51,
    JP_Sales: 0.03,
    Other_Sales: 0.23,
    Global_Sales: 4.48
  }
]
```

Vyhledání her od konkrétního vydavatele s vyšším prodejem v Evropě než 1 m:

```
db.games.find({"Publisher": "Nintendo", "EU_Sales": {$gt: 20}})
```

Návratová hodnota:

Seminární práce z předmětu NoSQL databáze

```
[
  {
    _id: ObjectId('67ed37daf8bcb9ec6a807cf2'),
    Rank: 1,
    Name: 'Wii Sports',
    Platform: 'Wii',
    Year: 2006,
    Genre: 'Sports',
    Publisher: 'Nintendo',
    NA_Sales: 41.49,
    EU_Sales: 29.02,
    JP_Sales: 3.77,
    Other_Sales: 8.46,
    Global_Sales: 82.74
  }
]
```

7.4 Validační schéma:

Healthcare

```
{
  '$jsonSchema': {
    bsonType: 'object',
    required: [
      'Name',
      'Age',
      'Gender',
      'Blood Type',
      'Medical Condition',
      'Date of Admission',
      'Doctor',
      'Hospital',
      'Insurance Provider',
      'Billing Amount',
      'Room Number',
      'Admission Type',
      'Discharge Date',
      'Medication',
      'Test Results'
    ],
    properties: {
      Name: {
        bsonType: 'string',
        description: 'Name of the patient, required.'
      },
      Age: {
        bsonType: 'int',
        description: 'Age of the patient, must be an integer and
required.'
      },
      Gender: {
        bsonType: 'string',
        description: 'Gender of the patient, must be either Male or
Female.'
      },
      'Blood Type': {
        bsonType: 'string',
        description: 'Blood type of the patient, must be one of the
specified types.'
      },
      'Medical Condition': {
        bsonType: 'string',
        description: 'Medical condition of the patient, required.'
      },
      'Date of Admission': {
```

Seminární práce z předmětu NoSQL databáze

```
    bsonType: [ 'string', 'date' ],
    description: 'Date of patient admission, required.'
  },
  Doctor: {
    bsonType: 'string',
    description: 'Attending doctors name, required.'
  },
  Hospital: {
    bsonType: 'string',
    description: 'Hospital name where the patient was admitted,
required.'
  },
  'Insurance Provider': {
    bsonType: 'string',
    description: 'Insurance provider covering the patient, required.'
  },
  'Billing Amount': {
    bsonType: 'double',
    description: 'Billing amount for the patients treatment, must be
a non-negative number.'
  },
  'Room Number': {
    bsonType: 'int',
    description: 'Room number where the patient was accommodated,
required.'
  },
  'Admission Type': {
    bsonType: 'string',
    description: 'Type of admission, must be one of Urgent,
Emergency, or Elective.'
  },
  'Discharge Date': {
    bsonType: [ 'string', 'date' ],
    description: 'Date of patient discharge, required.'
  },
  Medication: {
    bsonType: 'string',
    description: 'Medication prescribed to the patient during
admission.'
  },
  'Test Results': {
    bsonType: 'string',
    description: 'Results of tests conducted during the patients
admission, must be one of Normal, Inconclusive, or Abnormal.'
  }
}
}
```

Pokémon

```
{
  '$jsonSchema': {
    bsonType: 'object',
    required: [
      '#',
      'Type 1',
      'HP',
      'Defense',
      'Speed',
      'Legendary',
      'Name',
      'Total',
      'Attack',
      'Sp',
      'Generation'
    ]
  }
}
```

Seminární práce z předmětu NoSQL databáze

```
],
properties: {
  '#': {
    bsonType: 'int',
    description: 'Pokédex number, must be an integer and required.'
  },
  Name: { bsonType: 'string', description: 'Pokémon name, required.'
},
  'Type 1': {
    bsonType: 'string',
    description: 'Primary type of the Pokémon, required.'
  },
  'Type 2': {
    bsonType: [ 'string', 'null' ],
    description: 'Secondary type of the Pokémon, optional.'
  },
  Total: {
    bsonType: 'int',
    description: 'Total base stats, must be an integer.'
  },
  HP: {
    bsonType: 'int',
    minimum: 1,
    description: 'HP stat, must be an integer and at least 1.'
  },
  Attack: {
    bsonType: 'int',
    minimum: 1,
    description: 'Attack stat, must be an integer and at least 1.'
  },
  Defense: {
    bsonType: 'int',
    minimum: 1,
    description: 'Defense stat, must be an integer and at least 1.'
  },
  Sp: {
    bsonType: 'object',
    required: [ ' Atk', ' Def' ],
    properties: {
      ' Atk': {
        bsonType: 'int',
        minimum: 1,
        description: 'Special Attack stat, must be an integer and at
least 1.'
      },
      ' Def': {
        bsonType: 'int',
        minimum: 1,
        description: 'Special Defense stat, must be an integer and at
least 1.'
      }
    },
    description: 'Special Attack and Special Defense, stored as a
nested object.'
  },
  Speed: {
    bsonType: 'int',
    minimum: 1,
    description: 'Speed stat, must be an integer and at least 1.'
  },
},
```

Seminární práce z předmětu NoSQL databáze

```
Generation: {
  bsonType: 'int',
  minimum: 1,
  description: 'Game generation, must be an integer.'
},
Legendary: {
  bsonType: 'string',
  description: 'Indicates if the Pokémon is legendary
(true/false).'
}
}
}
```

Video Games Sales

```
{
  '$jsonSchema': {
    bsonType: 'object',
    required: [
      'Rank',
      'Name',
      'Platform',
      'Genre',
      'NA_Sales',
      'EU_Sales',
      'JP_Sales',
      'Other_Sales',
      'Global_Sales'
    ],
    properties: {
      Rank: {
        bsonType: 'int',
        description: 'Rank of the game, must be an integer and required.'
      },
      Name: {
        bsonType: 'string',
        description: 'Name of the game, required.'
      },
      Platform: {
        bsonType: 'string',
        description: 'Platform on which the game was released, required.'
      },
      Year: {
        bsonType: 'int',
        description: 'Year of release, must be an integer and required.'
      },
      Genre: {
        bsonType: 'string',
        description: 'Genre of the game, required.'
      },
      Publisher: {
        bsonType: 'string',
        description: 'Publisher of the game, required.'
      },
      NA_Sales: {
        bsonType: 'double',
        minimum: 0,
        description: 'Sales in North America, must be a non-negative
number.'
      },
    },
  },
}
```

Seminární práce z předmětu NoSQL databáze

```
EU_Sales: {
  bsonType: 'double',
  minimum: 0,
  description: 'Sales in Europe, must be a non-negative number.'
},
JP_Sales: {
  bsonType: 'double',
  minimum: 0,
  description: 'Sales in Japan, must be a non-negative number.'
},
Other_Sales: {
  bsonType: 'double',
  minimum: 0,
  description: 'Sales in other regions, must be a non-negative
number.'
},
Global_Sales: {
  bsonType: 'double',
  minimum: 0,
  description: 'Total global sales, must be a non-negative number.'
}
}
}
```