**QUADERNO 4**

// Domanda 1

db.bikeStations.aggregate([

  {

      $group: {

          \_id: "$extra.status",

          count: { $sum: 1 }

      }

  },

  {

      $match: {

          \_id: { $in: ["online", "offline"] }

      }

  }

]);

// Risposta:

{

  \_id: 'offline',

  count: 28

}

{

  \_id: 'online',

  count: 33

}

// Domanda 2

db.bikeStations.aggregate([

  {

      $match: {

          "extra.status": { $nin: ["online", "offline"] }

      }

  },

  {

      $count: "stations\_with\_other\_status"

  }

]);

// Risposta:

{

  stations\_with\_other\_status: 4

}

// Domanda 3

db.bikeStations.find(

  { "extra.status": { $nin: ["online", "offline"] } },

  { "extra.status": 1, \_id: 0 }

);

//Risposta:

{

  extra: {

    status: 'maintenance'

  }

}

{

  extra: {

    status: 'maintenance'

  }

}

{

  extra: {

    status: 'maintenance'

  }

}

{

  extra: {

    status: 'maintenance'

  }

}

// Domanda 4

db.bikeStations.find(

  { "extra.status": "online", "extra.score": { $gte: 4 } },

  { name: 1, \_id: 0 }

).sort({ name: 1 });

// Risposta:

{

name: '02. Pettiti'

}

{

  name: '04. Reggia'

}

{

  name: '06. Municipio'

}

{

  name: '08. San Marchese'

}

{

  name: '10. Gallo Praile'

}

{

  name: 'Belfiore'

}

{

  name: 'Borromini'

}

{

  name: 'Castello 1'

}

{

  name: 'Corte d`Appello'

}

{

  name: 'Giolitti 1'

}

{

  name: 'Politecnico 1'

}

{

  name: 'Politecnico 3'

}

{

  name: 'Porta Palatina'

}

{

  name: 'Principi d`Acaja 1'

}

{

  name: 'Principi d`Acaja 2'

}

{

  name: 'San Francesco da Paola'

}

{

  name: 'Sant´Anselmo'

}

{

  name: 'Tribunale'

}

//Domanda 5

db.bikeStations.find(

  {

      "extra.status": "offline",

      $or: [

          { empty\_slots: { $gt: 0 } },

          { free\_bikes: { $gt: 0 } }

      ]

  },

  { name: 1, empty\_slots: 1, free\_bikes: 1, \_id: 0 }

);

// Risposta:

{

  empty\_slots: 1,

  free\_bikes: 0,

  name: '06. Le Serre'

}

{

  empty\_slots: 0,

  free\_bikes: 5,

  name: '05. Corso Garibaldi'

}

// Domanda 6

db.bikeStations.aggregate([

  {

      $group: {

          \_id: null,

          total\_reviews: { $sum: "$extra.reviews" }

      }

  }

]);

// Risposta:

{

  \_id: null,

  total\_reviews: 15311

}

// Domanda 7

db.bikeStations.aggregate([

  {

      $group: {

          \_id: "$extra.score",

          station\_count: { $sum: 1 }

      }

  },

  {

      $sort: { \_id: -1 }

  }

]);

// Risposta:

{

  \_id: 4.7,

  station\_count: 1

}

{

  \_id: 4.5,

  station\_count: 2

}

{

  \_id: 4.4,

  station\_count: 2

}

{

  \_id: 4.3,

  station\_count: 2

}

{

  \_id: 4.2,

  station\_count: 7

}

{

  \_id: 4.1,

  station\_count: 5

}

{

  \_id: 4,

  station\_count: 9

}

{

  \_id: 3.9,

  station\_count: 9

}

{

  \_id: 3.8,

  station\_count: 1

}

{

  \_id: 3.7,

  station\_count: 2

}

{

  \_id: 3.6,

  station\_count: 1

}

{

  \_id: 3.5,

  station\_count: 4

}

{

\_id: 3.4,

station\_count: 3

}

{

  \_id: 3.2,

  station\_count: 1

}

{

  \_id: 3,

  station\_count: 4

}

{

  \_id: 2.8,

  station\_count: 2

}

{

\_id: 2.7,

station\_count: 1

}

{

  \_id: 2.5,

  station\_count: 1

}

{

  \_id: 2.4,

  station\_count: 1

}

{

  \_id: 2.1,

  station\_count: 1

}

// Domanda 8

db.bikeStations.aggregate([

  {

      $match: { "extra.status": { $in: ["online", "offline"] } }

  },

  {

      $group: {

          \_id: "$extra.status",

          average\_score: { $avg: "$extra.score" }

      }

  }

]);

// Risposta:

{

  \_id: 'online',

  average\_score: 3.8454545454545452

}

{

  \_id: 'offline',

  average\_score: 3.0285714285714285

}

// Domanda 9 (non ho usato mapReduce in quanto non lo conoscevo abbastanza e non ho trovato documentazione utile al suo utilizzo)

db.bikeStations.aggregate([

  {

      $group: {

          \_id: {

              has\_bikes: { $gt: ["$free\_bikes", 0] } // true: con bici, false: senza bici (spacca nei 2 casi)

          },

          average\_score: { $avg: "$extra.score" } // calcolo media

      }

  },

  {

      $project: { // uso project per migliore leggibilità dei risultati dei 2 casi (con relativa media)

          \_id: 0,

          has\_bikes: { $cond: ["$\_id.has\_bikes", "with\_bikes", "no\_bikes"] },

          average\_score: 1

      }

  }

]);

// Risposta:

{

  average\_score: 3.2305555555555556,

  has\_bikes: 'no\_bikes'

}

{

  average\_score: 3.8758620689655174,

  has\_bikes: 'with\_bikes'

}

// Domanda 10

db.bikeStations.aggregate([

  {

      $match: {

          "extra.status": "online" // Considera solo le stazioni attive

      }

  },

  {

      $group: {

          \_id: {

              has\_bikes: { $gt: ["$free\_bikes", 0] } // true: con bici, false: senza bici

          },

          average\_score: { $avg: "$extra.score" }

      }

  },

  {

      $project: { // guarda commento sopra (domanda 9)

          \_id: 0,

          has\_bikes: { $cond: ["$\_id.has\_bikes", "with\_bikes\_online", "no\_bikes\_online"] },

          average\_score: 1

      }

  }

]);

// Risposta:

{

  average\_score: 3.8642857142857143,

  has\_bikes: 'with\_bikes\_online'

}

{

  average\_score: 3.7399999999999998,

  has\_bikes: 'no\_bikes\_online'

}

// Domanda 11

// creazione indice location (per usare $near)

db.bikeStations.createIndex({ location: "2dsphere" });

// query

db.bikeStations.find(

  {

      free\_bikes: { $gt: 0 },

      location: {

          $near: { // operatore near per calcolo distanza rispetto alle coordinate

              $geometry: { type: "Point", coordinates: [45.07456, 7.69463] }

          }

      }

  },

  { name: 1, free\_bikes: 1, \_id: 0 }

).limit(3); // trovo solo le 3 più vicine

// Risposta:

{

  free\_bikes: 5,

  name: 'Palermo 2'

}

{

  free\_bikes: 5,

  name: 'Castello 1'

}

{

  free\_bikes: 4,

  name: 'San Francesco da Paola'

}

// Domanda 12

// calcolo coordinate di Politecnico 4 per riusarle nella query

const politecnico4 = db.bikeStations.findOne(

  { name: "Politecnico 4" },

  { "location.coordinates": 1, \_id: 0 }

).location.coordinates;

// query

db.bikeStations.find(

  {

      free\_bikes: { $gt: 0 },

      location: {

          $near: {

              $geometry: { type: "Point", coordinates: politecnico4 }

          }

      }

  },

  { name: 1, free\_bikes: 1, \_id: 0 }

).limit(3);

// Risposta:

{

  free\_bikes: 9,

  name: 'Politecnico 1'

}

{

  free\_bikes: 5,

  name: 'Politecnico 3'

}

{

  free\_bikes: 3,

  name: 'Tribunale'

}