BANK

CODE	NAME	ADDRESS	PHONE

BRANCH

BR_CODE	NAME	ADDRESS	PHONE	B_CODE

ACCOUNT

AC_NO	AC_TYPE	BALANCE	BR_CODE

CUSTOMERS

SSN	NAME	ADDRESS	PHONE	AGE

CUST_AC

AC_NO	SSN

LOAN

L_NO	DURATION	INTEREST	AMOUNT	BR_CODE

CUST_LOAN

L_NO	SSN

```
// Insert data into MongoDB collections
db.bank.insertMany([
  { code: 'SBI', name: 'State Bank of India', address: 'Bangalore', phone: 1234567890 },
  { code: 'HDFC', name: 'HDFC Bank', address: 'Mumbai', phone: 9876543210 }
]);

db.branch.insertMany([
  { _id: 1, name: 'SBI Bangalore Main', address: 'Bangalore Main Street', phone: 1111111111,
b_code: 'SBI' },
  { _id: 2, name: 'SBI Bangalore Branch 2', address: 'Bangalore Street 2', phone: 2222222222,
b_code: 'SBI' },
  { _id: 3, name: 'HDFC Mumbai Main', address: 'Mumbai Main Street', phone: 3333333333,
b_code: 'HDFC' }
]);

db.account.insertMany([
  { ac_no: 1, ac_type: 'joint', balance: 1000, bid: 1 },
```

```
  { ac_no: 2, ac_type: 'sb', balance: 500, bid: 1 },
  { ac_no: 3, ac_type: 'joint', balance: 1500, bid: 2 },
  { ac_no: 4, ac_type: 'sb', balance: 200, bid: 3 }
]);

db.customers.insertMany([
  { ssn: 123456789, name: 'Alice', address: 'Address 1', phone: 1111111111, age: 25 },
  { ssn: 234567890, name: 'Bob', address: 'Address 2', phone: 2222222222, age: 30 },
  { ssn: 345678901, name: 'Charlie', address: 'Address 3', phone: 3333333333, age: 35 },
  { ssn: 456789012, name: 'David', address: 'Address 4', phone: 4444444444, age: 40 }
]);

db.cust_ac.insertMany([
  { acc_no: 1, ssn: 123456789 },
  { acc_no: 2, ssn: 123456789 },
  { acc_no: 3, ssn: 234567890 },
  { acc_no: 4, ssn: 345678901 }
]);

db.loan.insertMany([
  { lno: 1, duration: 12, interest: 5.5, amount: 2000, bid: 1 },
  { lno: 2, duration: 24, interest: 6.2, amount: 3000, bid: 2 },
  { lno: 3, duration: 18, interest: 5.8, amount: 2500, bid: 3 },
  { lno: 4, duration: 36, interest: 7.0, amount: 4000, bid: 1 }
]);

db.cust_loan.insertMany([
  { lno: 1, ssn: 123456789 },
  { lno: 2, ssn: 234567890 },
  { lno: 3, ssn: 345678901 },
  { lno: 4, ssn: 456789012 }
]);

// Queries

// a) List the details of customers who have a joint account and also have at least one loan.
db.customers.find({
  ssn: {
    $in: db.cust_ac.distinct("ssn", {
      acc_no: {
        $in: db.account.distinct("ac_no", { ac_type: "joint" })
      }
    }),
    $in: db.cust_loan.distinct("ssn")
```

```
  }
});


// b) List the details of the branch which has given the maximum loan.
db.branch.findOne({
  _id: db.loan.find().sort({ amount: -1 }).limit(1).next().bid
});


// c) List the details of saving accounts opened in the SBI branches located at Bangalore.
db.account.find({
  ac_type: 'sb',
  bid: {
    $in: db.branch.find({
      b_code: 'SBI',
      address: /Bangalore/
    }).toArray().map(b => b._id)
  }
});


// d) List the name of the branch along with its bank name and total amount of loan given by it.
db.branch.aggregate([
  {
    $lookup: {
      from: "loan",
      localField: "_id",
      foreignField: "bid",
      as: "loans"
    }
  },
  {
    $group: {
      _id: "$_id",
      branch_name: { $first: "$name" },
      bank_name: { $first: "$b_code" },
      total_loan_amount: { $sum: "$loans.amount" }
    }
  }
]);
```

```
// e) Retrieve the names of customers who have accounts in all the branches located in a
specific city.
db.customers.find({
  ssn: {
    $in: db.cust_ac.aggregate([
      {
        $lookup: {
          from: "account",
          localField: "acc_no",
          foreignField: "ac_no",
          as: "accounts"
        }
      },
      {
        $lookup: {
          from: "branch",
          localField: "accounts.bid",
          foreignField: "_id",
          as: "branches"
        }
      },
      {
        $match: {
          "branches.address": "Bangalore"
        }
      },
      {
        $group: {
          _id: "$ssn",
          branch_count: { $sum: 1 }
        }
      },
      {
        $match: {
          branch_count: db.branch.count({ address: "Bangalore" })
        }
      }
    ]).map(ca => ca._id)
  }
});
```