# Node.js Pre-Assessment Questions

## 1. Find the Account Number of the first customer with duplicate accounts from customers array given below.

```
const customers = [
   {
     "AccountNo": 1001,
     "Name": "John",
     "City": "Bangalore"
   },
   {
     "AccountNo": 1002,
     "Name": "Tom",
     "City": "Mysore"
   },
   {
     "AccountNo": 1003,
     "Name": "Kate",
     "City": "Pune"
   },
   {
     "AccountNo": 1004,
     "Name": "Tom",
     "City": "Delhi"
   },
   {
     "AccountNo": 1005,
     "Name": "Kate",
     "City": "Mumbai"
   }
]
```

## 2. Write a Node.js function to populate the following transactions data into MongoDB

```
const transactions = [
  {
    "Date": "12-01-2022",
    "From": 1001,
    "To": 1003,
    "Amount": 15000
  },
  {
    "Date": "12-01-2022",
    "From": 1003,
    "To": 1002,
    "Amount": 12000
  },
  {
    "Date": "12-01-2022",
    "From": 1002,
    "To": 1005,
    "Amount": 2000
  },
  {
    "Date": "13-01-2022",
    "From": 1003,
    "To": 1001,
    "Amount": 1000
  },
  {
    "Date": "13-01-2022",
    "From": 1002,
    "To": 1001,
    "Amount": 1200
  },
  {
    "Date": "14-01-2022",
    "From": 1001,
    "To": 1002,
    "Amount": 3000
  },
  {
    "Date": "14-01-2022",
    "From": 1005,
    "To": 1003,
    "Amount": 1500
  },
  {
    "Date": "15-01-2022",
    "From": 1005,
    "To": 1003,
    "Amount": 4000
  }
]
```

## 3. Create an API using Express.js to read above transactions data from MongoDB

Method: GET
Route: /transactions?sortBy=FIELDNAME

## 4. Calculate the final balance of each Customer from the transactions array, if initial balance is zero.

```
const customers = [
  {
    "AccountNo": 1001,
    "Name": "John",
    "City": "Bangalore"
  },
  {
    "AccountNo": 1002,
    "Name": "Tom",
    "City": "Mysore"
  },
  {
    "AccountNo": 1003,
    "Name": "Kate",
    "City": "Pune"
  },
  {
    "AccountNo": 1004,
    "Name": "Paul",
    "City": "Delhi"
  },
  {
    "AccountNo": 1005,
    "Name": "Riya",
    "City": "Mumbai"
  }
]
```

```
const transactions = [
  {
    "Date": "12-01-2022",
    "From": 1001,
    "To": 1003,
    "Amount": 15000
  },
  {
    "Date": "12-01-2022",
    "From": 1003,
    "To": 1002,
    "Amount": 12000
  },
  {
    "Date": "12-01-2022",
    "From": 1002,
    "To": 1005,
    "Amount": 2000
  },
  {
    "Date": "13-01-2022",
    "From": 1003,
    "To": 1001,
    "Amount": 1000
  },
  {
    "Date": "13-01-2022",
    "From": 1002,
    "To": 1001,
    "Amount": 1200
  },
  {
    "Date": "14-01-2022",
    "From": 1001,
    "To": 1002,
    "Amount": 3000
  },
  {
    "Date": "14-01-2022",
    "From": 1005,
    "To": 1003,
    "Amount": 1500
  },
  {
    "Date": "15-01-2022",
    "From": 1005,
    "To": 1003,
    "Amount": 4000
  }
]
```

## 5. Find the Account Number of the Customer that has sent the most amount of money from his account.

```
const customers = [
  {
    "AccountNo": 1001,
    "Name": "John",
    "City": "Bangalore"
  },
  {
    "AccountNo": 1002,
    "Name": "Tom",
    "City": "Mysore"
  },
  {
    "AccountNo": 1003,
    "Name": "Kate",
    "City": "Pune"
  },
  {
    "AccountNo": 1004,
    "Name": "Paul",
    "City": "Delhi"
  },
  {
    "AccountNo": 1005,
    "Name": "Riya",
    "City": "Mumbai"
  }
]
```

```
const transactions = [
  {
    "Date": "12-01-2022",
    "From": 1001,
    "To": 1003,
    "Amount": 15000
  },
  {
    "Date": "12-01-2022",
    "From": 1003,
    "To": 1002,
    "Amount": 12000
  },
  {
    "Date": "12-01-2022",
    "From": 1002,
    "To": 1005,
    "Amount": 2000
  },
  {
    "Date": "13-01-2022",
    "From": 1003,
    "To": 1001,
    "Amount": 1000
  },
  {
    "Date": "13-01-2022",
    "From": 1002,
    "To": 1001,
    "Amount": 1200
  },
  {
    "Date": "14-01-2022",
    "From": 1001,
    "To": 1002,
    "Amount": 3000
  },
  {
    "Date": "14-01-2022",
    "From": 1005,
    "To": 1003,
    "Amount": 1500
  },
  {
    "Date": "15-01-2022",
    "From": 1005,
    "To": 1003,
    "Amount": 4000
  }
]
```