

**KEVIN:**

1. Contar usuarios agrupados por género, cuenta cuántos usuarios hay por cada género (MALE, FEMALE, OTHER):

```
db.users.aggregate([
  {
    $match: { U_GENDER: { $exists: true } }
  },
  {
    $group: {
      _id: "$U_GENDER",
      totalUsers: { $sum: 1 }
    }
  }
])
```

2. Buscar usuarios cuyo email pertenezca a un dominio concreto, obtiene los usuarios cuyo email sea de un dominio específico (por ejemplo @sandia.com):

```
db.users.find(
  { P_EMAIL: { $regex: "@sandia\\.com$" } },
  { P_USERNAME: 1, P_EMAIL: 1, _id: 0 }
)
```

3. Obtener los usuarios ordenados alfabéticamente por apellido, devuelve todos los usuarios ordenados por P\_LASTNAME (y nombre como segundo criterio)

```
db.users.find(
  {},
  { P_NAME: 1, P_LASTNAME: 1, P_USERNAME: 1, _id: 0 }
).sort({ P_LASTNAME: 1, P_NAME: 1 })
```

**XABI:**

1. Usuarios que tienen tarjeta y cuyo número empieza por 4 (Visa-like)

```
db.users.find(
  {
    U_CARD: { $exists: true },
    U_CARD: { $regex: "^4" }
  },
  { P_USERNAME: 1, U_CARD: 1, _id: 0 }
)
```

2. Nombres que contienen letra 'e'

```
db.users.find(  
  { P_NAME: { $regex: "e" } },  
  { P_NAME: 1, P_USERNAME: 1, _id: 0 }  
)
```

3. Tarjetas que tienen al menos un '3'

```
db.users.find(  
  {  
    U_CARD: {  
      $exists: true,  
      $regex: "3"  
    }  
  },  
  { P_USERNAME: 1, U_CARD: 1, _id: 0 }  
)
```

#### **MOSI:**

1. Usuarios que no tengan cuenta bancaria introducido

```
db.users.find({ "A_CURRENT_ACCOUNT": { $exists: false } })
```

2. Usuarios cuyo número de teléfono empieza por 432

```
db.users.find({ "U_CARD": /^432/ })
```

3. Usuario con el primer nombre mas corto

```
db.users.aggregate([  
  { $project: { P_NAME: 1, P_NAME_LENGTH: { $strLenCP: "$P_NAME" } } },  
  { $sort: { P_NAME: 1 } },  
  { $limit: 1 }  
)
```

#### **EKAIN:**

1. agrega un telefono por defecto en caso de que alguno sea null

```
db.users.aggregate(  
  [  
    {  
      $project: {  
        item: 1,  
        description: { $ifNull: [ "$telefono", "0000000000" ] }  
      }  
    }  
  ]  
)
```

2. cuenta los usuarios cuyo nombre empieze por p  
db.users.aggregate([

```
{  
  $match: { P_Uname: {$regex: '^P' }  
},  
{  
  $count: "usuarios_Con_P."  
}  
])
```

3.

db.users.getIndexes()