4) Obtener el poder promedio de los movimientos de cada tipo de Pokémon, mostrando solo aquellos tipos cuyo poder promedio sea mayor a 60, ordenados de mayor a menor poder promedio.

```
[
  {
    $unwind: {
      path: "$moves"
    }
  },
  {
    $unwind: {
      path: "$type"
    }
  },
  {
    $project: {
      _id: 0,
      //uso para ocultar el idpoke
      typeID: "$type.IDtype",
      typename: "$type.typename",
      power: "$moves.power"
    }
  },
  {
    $group: {
      _id: {
        id: "$typeID",
        name: "$typename"
      },
      promPower: {
        $avg: "$power"
      }
    }
  },
  {
    $match: {
      promPower: {
        $gt: 60
      }
    }
  },
  {
    $sort: {
      promPower: -1
    }
  }
]
```

5) Crear un ranking de los 10 Pokémon con mayor suma total de estadísticas base (HP + attack + defense + spattack + spdefense + speed), incluyendo el nombre del Pokémon, su suma total de estadísticas, y la cantidad de tipos que posee. Ayuda: la expresión $add tiene como parámetro una lista que suma los valores de la misma. Ejemplo: { $add: [$HP, 1] } le suma 1 al HP.

```
[
  {
    $addFields: {
      totalStats: {
        $add: [
          "$HP",
          "$attack",
          "$defense",
          "$spattack",
          "$spdefense",
          "$speed"
        ]
      }
    }
  },{
    $addFields: {
      typeCount: {
        $size: "$type"
      }
    }
  },{
    $project: {
      _id: 0,
      pokename: 1,
      totalStats: 1,
      typeCount: 1
    }
  },{
    $sort: {
      totalStats: -1
    }
  },{
    $limit: 10
  }
]
```

6) Encontrar el nombre, ataque y defensa de todos los pokémon cuyo poder de todos sus movimientos de categoría 'Physical' sea mayor a 85.

```
[
  {
    $unwind: {
      path: "$moves"
```

```
    }
  },
  {
    $match: {
      "moves.category": "Physical"
    }
  },
  {
    $group: {
      _id: "$pokename",
      attack: { $first: "$attack" },
      defense: { $first: "$defense" },
      totalPhysical: { $sum: 1 },
      physicalGT85: {
        $sum: {
          $cond: [
            { $gt: ["$moves.power", 85] },
            1,
            0
          ] //si moves.power> 85 suma uno
        }
      }
    }
  },
  {
    $match: {
      $expr: {
        $eq: ["$totalPhysical", "$physicalGT85"]
      }
    }
  },
  {
    $project: {
      _id: 0,
      pokename: "$_id",
      attack: 1,
      defense: 1
    }
  }
]
```

PARCIALITO 6

2. [sobre la base de Pokémon] Se quiere construir una nueva tabla con posibles enfrentamientos, para ello se desea categorizar a los Pokémon según su nivel de poder. Para esto, se quieren generar 3 grupos por las estadísticas de ataque y defensa. Cada grupo estará compuesto por una lista de los pókemon que pertenecen a ese grupo. Para el cálculo de los grupos se deberá utilizar el promedio de esa estadística y operar para obtener los intervalos de valores que definen a cada grupo (AVG): ○ [0 ; AVG) → grupo 1 “normal” ○

[AVG ; 2 AVG) → grupo 2 “avanzado” ○ > 2 AVG → grupo 3 “legendario” [{ _id: “attack”, // grupos normal: [/* pokemon con atk entre 0 y avg de atk */], avanzado: [/* pokemon con atk entre avg y 2 avg de atk */], legendario: [/* pokemon con atk mayor a 2 avg de atk */] }, { _id: “defense”, … /* idem grupos pero con defensse */ }, } Ayuda: se puede utilizar "$noval" para no agregar un valor a una lista durante el uso del agregador $push . Ejemplo: { $push: { $cond: [{ falso / verdadero }, "$campo_valido", "$noval" // no se agrega nada a la lista }

```
[
  {
    $group: {
      _id: null,
      avgAttack: {
        $avg: "$attack"
      },
      avgDefense: {
        $avg: "$defense"
      },
      pokes: {
        $push: {
          name: "$pokename",
          attack: "$attack",
          defense: "$defense"
        }
      }
    }
  },
  {
    $unwind: {
      path: "$pokes"
    }
  },
  {
    $group: {
      _id: null,
      normal: {
        $push: {
          $cond: [
            {
              $lte: [
                "$pokes.attack",
                "$avgAttack"
              ]
            },
            "$pokes.name",
            "$noval"
          ]
        }
      },
      avanzado: {
```

```
  $push: {
    $cond: [
      {
        $and: [
          {
            $gte: [
              "$pokes.attack",
              "$avgAttack"
            ]
          },
          {
            $lt: [
              "$pokes.attack",
              {
                $multiply: ["$avgAttack", 2]
              }
            ]
          }
        ]
      },
      "$pokes.name",
      "$noval"
    ]
  }
},
legendario: {
  $push: {
    $cond: [
      {
        $gte: [
          "$pokes.attack",
          {
            $multiply: ["$avgAttack", 2]
          }
        ]
      },
      "$pokes.name",
      "$noval"
    ]
  }
},
normal_def: {
  $push: {
    $cond: [
      {
        $lt: [
          "$pokes.defense",
          "$avgDefense"
```

```
        ]
      },
      "$pokes.name",
      "$noval"
    ]
  }
},
avanzado_def: {
  $push: {
    $cond: [
      {
        $and: [
          {
            $gte: [
              "$pokes.defense",
              "$avgDefense"
            ]
          },
          {
            $lt: [
              "$pokes.defense",
              {
                $multiply: [
                  "$avgDefense",
                  2
                ]
              }
            ]
          }
        ]
      },
      "$pokes.name",
      "$noval"
    ]
  }
},
legendario_def: {
  $push: {
    $cond: [
      {
        $gte: [
          "$pokes.defense",
          {
            $multiply: ["$avgDefense", 2]
          }
        ]
      },
      "$pokes.name",
```

```
          "$noval"
        ]
       }
      }
     }
    },
    {
     $project: {
      _id: 0,
      attack: {
       normal: "$normal",
       avanzado: "$avanzado",
       legendario: "$legendario"
      },
      defense: {
       normal: "$normal_def",
       avanzado: "$avanzado_def",
       legendario: "$legendario_def"
      }
     }
    }
]
```

1. [sobre la base de tweets] La captura de tweets es toda en un mismo día: hallar el hashtag más utilizado en cada hora del día (y cuántos usos tuvo).

```
[
  {
    $addFields: {
      hour: { $hour: "$created_at" }
    }
  },
  {
    $unwind: {
      path: "$entities.hashtags"
    }
  },
  {
    $group: {
      _id: {
        hour: "$hour",
        text: "$entities.hashtags.text"
      },
      sumatoriaHashtag: {
        $sum: 1
      }
    }
  },
  {
```

```
    $sort: {
      sumatoriaHashtag: -1
    }
  },
  {
    $group: {
      _id: "$_id.hour",
      texto: { $first: "$_id.text" },
      cantidad: { $first: "$sumatoriaHashtag" }
    }
  },
  {
    $sort: {
      _id: 1
    }
  }
]
```

tweets:

1. Mostrar de los 10 tweets con más retweets, su usuario y la cantidad de retweets. Ordenar la salida de forma ascendente.

```
[
  {
    $sort: {
      retweet_count: -1
    }
  },
  {
    $limit: 10
  },
  {
    $sort: {
      retweet_count: 1
    }
  },
  {
    $project: {
      _id: 0,
      usuario: "$user.screen_name",
      cantidadRetweet: "$retweet_count"
    }
  }
]
```

2. Encontrar los 10 hashtags más usados.

```
[
```

```
  {
    $unwind: {
      path: "$entities.hashtags"
    }
  }, {
    $group: {
      _id: "$entities.hashtags.text",
      cantidadHashtag: {
        $sum: 1
      }
    }
  },{
    $sort: {
      cantidadHashtag: -1
    }
  },{
    $limit: 10
  }
]
```

3- Por cada usuario obtener una lista de ids de tweets y el largo de la misma. (pista: $push función de agregación que agrega elementos a una lista, ARRAY_AGG de SQL)

```
[
  {
    $group: {
      _id: "$user.screen_name",
      idsTweets: {
        $push: "$_id"
      }
    }
  },{
    $project:{
      _id: 0,
      user: "$_id",
      listaTweets: "$idsTweets",
      largoLista: {$size: "$idsTweets"}
    }
  }
]
```

4- Encontrar los 5 usuarios más mencionados (les hicieron @, podemos encontrar esta información en entities.user_mentions)

```
[
  {
    $unwind: {
      path: "$entities.user_mentions"
    }
  },{
```

```
    $group: {
      _id: "$entities.user_mentions.screen_name",
      cantMenciones: {
        $sum: 1
      }
    }
  },{
    $sort: {
      cantMenciones:-1
    }
  },{
    $limit: 5
  }
]
```

5-Hallar la máxima cantidad de retweets totales que tuvo algún usuario.

```
[
  {
    $group: {
      id: "$user", //si se quiere también el usuario, usar user.screen_name y no hacer la proy
      totalRetweet: {
        $sum: "$retweet_count"
      }
    }
  },{
    $project: {
      _id:0,
      totalRetweet: 1
    }
  },{
    $sort: {
      totalRetweet: -1
    }
  },{
    $limit: 1
  }
]
```

6- Hallar la cantidad de retweets promedio para los tweets que se hayan hecho desde Argentina (place.contry == "Argentina" ó place.country_code == "Argentina") y aquellos que no. Ayuda: similar a CASE WHEN tenemos la expresión: { $cond: [, ,] } error

```
[
  {
    $group: {
      _id: null,
      sumatoriaRetweetsArg: {
        $sum: {
          $cond: [
```

```
      {
        $or: [
          {
            $eq: [
              "$place.country",
              "Argentina"
            ]
          },
          {
            $eq: [
              "$place.country_code",
              "Argentina"
            ]
          }
        ]
      },
      "$retweet_count",
      0
    ]
  }
},
sumatoriaTweetsArg: {
  $sum: {
    $cond: [
      {
        $or: [
          {
            $eq: [
              "$place.country",
              "Argentina"
            ]
          },
          {
            $eq: [
              "$place.country_code",
              "Argentina"
            ]
          }
        ]
      },
      1,
      0
    ]
  }
},
sumatoriaRetweetsGral: {
  $sum: {
    $cond: [
```

```
        {
          $and: [
            {
              $ne: [
                "$place.country",
                "Argentina"
              ]
            },
            {
              $ne: [
                "$place.country_code",
                "Argentina"
              ]
            }
          ]
        },
        "$retweet_count",
        0
      ]
    }
  },
  sumatoriaTweetsGral: {
    $sum: {
      $cond: [
        {
          $and: [
            {
              $ne: [
                "$place.country",
                "Argentina"
              ]
            },
            {
              $ne: [
                "$place.country_code",
                "Argentina"
              ]
            }
          ]
        },
        1,
        0
      ]
    }
  }
 }
},
{
```

```
    $project: {
      _id: 0,
      promedioRetweetsArgentina: {
        $divide: [
          "$sumatoriaRetweetsArg",
          "$sumatoriaTweetsArg"
        ]
      },
      promedioRetweetsGeneral: {
        $divide: [
          "$sumatoriaRetweetsGral",
          "$sumatoriaTweetsGral"
        ]
      }
    }
  }
]
```

7- Hallar para cada intervalo de una hora cuántos tweets realizó cada usuario. Ayuda: el operador $hour permite extraer la hora de un campo de tipo Fecha

```
[
  {
    $addFields: {
      hour: { $hour: "$created_at" }
    }
  },{
      $group: {
        _id: {usuario: "$user.screen_name",hora: "$hour"},
        cantidadTweets: {
          $sum: 1
        }
      }
  },{
    $project: {
      _id:0,
      usuario: "$_id.usuario",
      hora: "$_id.hora",
      cantidadTweets: 1
    }
  }
]
```

EJERCICIOS DE NEO4J

7. Encuentre todos los equipos participantes del mundial 2022 que no participaron en el mundial 2018.

```
//Encuentre todos los equipos participantes del mundial 2022 que no participaron en el mundial 2018
MATCH (t:Team)-[:PARTICIPATES_IN]-(wc:WorldCup{year:2022})
WHERE NOT EXISTS {
(t)-[:PARTICIPATES_IN]-(wc2018:WorldCup{year:2018})
}
RETURN t
```

8. Encuentre los jugadores finalistas del mundial 2022 que nunca jugaron un partido entrando como suplente en dicho mundial.

```
MATCH (wc:WorldCup{year: 2022})<-[:PART_OF]-(gFinal:Game{stage:
"final"})-[:IS_HOME|IS_AWAY]-(t:Team)
MATCH (p:Player)-[:REPRESENTS]-(t)
WHERE NOT EXISTS{
(p)-[:PLAYED_IN{substitute:true}]-(g:Game)-[:PART_OF]-(wc)
}
RETURN p.first_name,p.last_name
```

12. Encuentre al jugador más joven en haber marcado un gol.
sugerencia: investigue las funciones:
- duration.between(from, to).<days|months|years>
- duration.inDays(from, to).days