



PROYECTO MONGODB

BASE DE DATOS

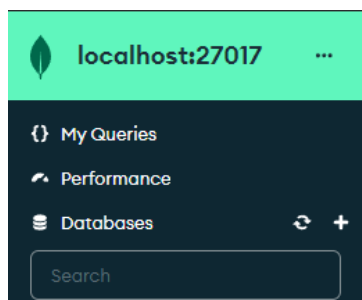


ÍNDICE

1. CREACIÓN DE BBDD Y COLECCIONES.
 2. INSERCIÓN DE INFORMACIÓN.
 3. CONSULTAS DE BÚSQUEDA.
 4. CONSULTAS AVANZADAS.
 5. CONEXIÓN A PYTHON.
-

1. Creación de base de datos y colecciones.

En primer lugar, deberemos crear la base de datos en la cual vamos a trabajar. Nos dirigiremos al programa MongoDB Compass y clicamos en el símbolo de + para agregar una nueva base de datos:

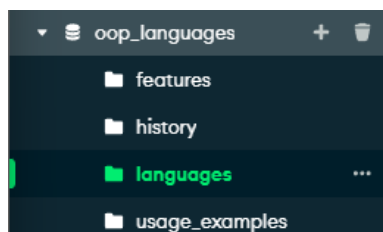


Al clicar, nos pedirá un nombre para la base de datos y una primera colección:

Introducimos el nombre de la base de datos y una de las colecciones que deseamos crear.

Una vez introducidos los dos valores, hacemos clic en “Create Database”.

Con la base de datos creada, añadimos las demás colecciones. Para ello, clicamos en el signo + que se encuentra a la derecha del nombre de la base de datos previamente creada e indicamos el nombre de la colección a crear.



Una vez creadas las diferentes colecciones, ya tendremos creada la base de datos completa.

2. Inserción de información.

El siguiente paso será introducir información en cada una de las colecciones de la base de datos. Comenzaremos con la colección

language

```
db.language.insert({
  _id: 1,
  Nombre: "C++",
  Anyo: 1980,
  Paradigma: "Multiparadigma",
  "Desarrollador/es": "Bjarne Stroustrup",
  Descripcion: "Es un lenguaje de programación que proviene de la
extensión del lenguaje C para que pudiese manipular objetos."
})

db.language.insert({
  _id: 2,
  Nombre: "Java",
  Anyo: 1995,
  Paradigma: "Programación Orientada a Objetos",
  "Desarrollador/es": "James Gosling",
  Descripcion: "Java es un lenguaje multiplataforma, orientado a
objetos y centrado en la red que se puede utilizar como una
plataforma en sí mismo."
})

db.language.insert({
  _id: 3,
  Nombre: "Ruby",
  Anyo: 1993,
  Paradigma: "Modelo Vista Controlador (MVC)",
  "Desarrollador/es": "Yukihiro Matsumoto",
  Descripcion: "Ruby es un lenguaje de programación que se utiliza
tanto para el desarrollo de aplicaciones web hasta el análisis de
datos, entre otras muchas cosas."
})

db.language.insert({
  _id: 4,
```

```
Nombre: "Visual Basic",
Anyo: 1991,
Paradigma: "Programación Orientada a Objetos",
"Desarrollador/es": "Alan Cooper",
Descripcion: "Es un lenguaje de programación dirigido por
Eventos. Este agiliza y simplifica la creación de aplicaciones."
})

db.language.insert({
  _id: 5,
  Nombre: "Visual C#",
  Anyo: 2002,
  Paradigma: "Programación Orientada a Objetos",
  "Desarrollador/es": "Anders Hejlsberg",
  Descripcion: "Lenguaje de programación moderno, basado en objetos
y con seguridad de tipos. C# permite a los desarrolladores crear
aplicaciones seguras y sólidas que se ejecutan en .NET."
})

db.language.insert({
  _id: 6,
  Nombre: "TypeScript",
  Anyo: 2010,
  Paradigma: "Programación Orientada a Objetos y programación
funcional",
  "Desarrollador/es": "Microsoft",
  Descripcion: "Es un superset de JavaScript que añade tipado
estático y otros elementos que facilitan el desarrollo de
aplicaciones a gran escala."
})

db.language.insert({
  _id: 7,
  Nombre: "Smalltalk",
  Anyo: 1980,
  Paradigma: "Programación Orientada a Objetos",
  "Desarrollador/es": "Alan Key, Adele Goldberg, Daniel Henry
Holmes Ingalls",
  Descripcion: "Java es un lenguaje multiplataforma, orientado a
objetos y centrado en la red que se puede utilizar como una
plataforma en sí mismo."
})

db.language.insert({
  _id: 8,
```

```
Nombre: "PHP",
Anyo: 1994,
Paradigma: "Imperativo, funcional, orientado a objetos,
procedural y reflexivo",
"Desarrollador/es": "Rasmus Lerdorf",
Descripcion: "Lenguaje de programación destinado a desarrollar
aplicaciones para la web y crear páginas web, favoreciendo la
conexión entre los servidores y la interfaz de usuario."
})

db.language.insert({
  _id: 9,
  Nombre: "Python",
  Anyo: 1989,
  Paradigma: "Imperativo, Orientado a objetos (POO) y Funcional.",
  "Desarrollador/es": "Guido van Rossum",
  Descripcion: "Es un lenguaje de programación ampliamente
utilizado en las aplicaciones web, el desarrollo de software, la
ciencia de datos y el machine learning."
})

db.language.insert({
  _id: 10,
  Nombre: "Perl",
  Anyo: 1987,
  Paradigma: "Imperativo, Orientado a objetos (POO), Funcional y
Procedimental.",
  "Desarrollador/es": "Larry Wall",
  Descripcion: "Es un lenguaje de programación ampliamente
utilizado en las aplicaciones web, el desarrollo de software, la
ciencia de datos y el machine learning."
})
```

features

```
db.features.insert({
  Nombre: "C++",
  Herencia: true,
  Multiplataforma: false,
  Polimorfismo: true,
  Encapsulamiento: true
})
```

```
db.features.insert({
  Nombre: "Java",
  Herencia: true,
  Multiplataforma: true,
  Polimorfismo: true,
  Encapsulamiento: true
})
```

```
db.features.insert({
  Nombre: "Ruby",
  Herencia: true,
  Multiplataforma: true,
  Polimorfismo: true,
  Encapsulamiento: true
})
```

```
db.features.insert({
  Nombre: "Visual Basic",
  Herencia: true,
  Multiplataforma: false,
  Polimorfismo: true,
  Encapsulamiento: true
})
```

```
db.features.insert({
  Nombre: "Visual C#",
  Herencia: true,
  Multiplataforma: false,
  Polimorfismo: true,
  Encapsulamiento: true
})
```

```
db.features.insert({
  Nombre: "TypeScript",
  Herencia: true,
  Multiplataforma: true,
  Polimorfismo: true,
  Encapsulamiento: true,
  Compilacion_estatica: true
})
```

```
db.features.insert({
  Nombre: "Smalltalk",
  Herencia: true,
  Multiplataforma: true,
  Polimorfismo: true,
  Encapsulamiento: true
})
```

```
db.features.insert({
  Nombre: "PHP",
  Herencia: true,
  Multiplataforma: true,
  Polimorfismo: true,
  Encapsulamiento: true
})
```

```
db.features.insert({
  Nombre: "Python",
  Herencia: true,
  Multiplataforma: true,
  Polimorfismo: true,
  Encapsulamiento: true
})
```

```
db.features.insert({
  Nombre: "Perl",
  Herencia: false,
  Multiplataforma: true,
  Polimorfismo: true,
  Encapsulamiento: true
})
```

history

```
db.history.insert({
  Nombre: "C++",
  Eventos: [
    {
      Fecha: "1983",
      Evento: "Se publica la especificación original de C++ por
Bjarne Stroustrup.",
      Proyectos: [
        {
          Nombre: "Mozilla Firefox",
          Descripcion: "Uno de los navegadores web más
populares del mundo, desarrollado en C++ y JavaScript. Es un proyecto
de software libre."
        }
      ]
    },
    {
      Fecha: "1998",
      Evento: "Se publica el estándar ISO/IEC 14882:1998, que
formaliza C++ como un estándar internacional.",
      Proyectos: []
    },
    {
      Fecha: "2011",
      Evento: "Se publica el estándar C++11, introduciendo
importantes mejoras al lenguaje.",
      Proyectos: []
    }
  ]
})

db.history.insert({
  Nombre: "Java",
  Eventos: [
    {
      Fecha: "1995",
      Evento: "Java es lanzado públicamente por Sun
Microsystems.",
      Proyectos: [
        {
```

```

        Nombre: "Apache Tomcat",
        Descripcion: "Un servidor web de código abierto y
contenedor servlet, desarrollado en Java. Es un proyecto de software
libre."
    },
    {
        Fecha: "2004",
        Evento: "Se publica el estándar Java SE 6, introduciendo
mejoras significativas.",
        Proyectos: []
    },
    {
        Fecha: "2017",
        Evento: "Oracle anuncia el lanzamiento de Java SE 9, que
introduce el sistema de módulos.",
        Proyectos: []
    }
]
}))

db.history.insert({
    Nombre: "Ruby",
    Eventos: [
        {
            Fecha: "1995",
            Evento: "Yukihiro Matsumoto (Matz) lanza la primera
versión pública de Ruby.",
            Proyectos: [
                {
                    Nombre: "Ruby on Rails",
                    Descripcion: "Un popular framework de desarrollo
web basado en Ruby. Es un proyecto de software libre."
                }
            ]
        },
        {
            Fecha: "2005",
            Evento: "Se lanza Ruby on Rails, un popular framework de
desarrollo web basado en Ruby.",
            Proyectos: []
        },
        {
            Fecha: "2013",

```

```
        Evento: "Se publica Ruby 2.0, con mejoras significativas
en rendimiento y usabilidad.",
        Proyectos: []
    }
]
}))

db.history.insert({
    Nombre: "Visual Basic",
    Eventos: [
        {
            Fecha: "1991",
            Evento: "Microsoft lanza Visual Basic 1.0.",
            Proyectos: []
        },
        {
            Fecha: "2008",
            Evento: "Se publica Visual Basic .NET como parte de
Visual Studio 2008.",
            Proyectos: []
        },
        {
            Fecha: "2020",
            Evento: "Microsoft anuncia el fin del soporte para Visual
Basic 6.0, marcando el fin de una era.",
            Proyectos: []
        }
    ]
}))

db.history.insert({
    Nombre: "Visual C#",
    Eventos: [
        {
            Fecha: "2000",
            Evento: "Microsoft anuncia C# como parte de la plataforma
.NET.",
            Proyectos: []
        },
        {
            Fecha: "2005",
            Evento: "Se lanza C# 2.0, introduciendo características
como genéricos y delegados anónimos.",
            Proyectos: []
        },
    ],
})
```

```
{
    Fecha: "2017",
    Evento: "Microsoft lanza C# 7.0, con mejoras en
expresiones lambda y tuplas.",
    Proyectos: []
}
]
})

db.history.insert({
    Nombre: "TypeScript",
    Eventos: [
        {
            Fecha: "2012",
            Evento: "Microsoft anuncia TypeScript como un superset de
JavaScript para el desarrollo de aplicaciones a gran escala.",
            Proyectos: []
        },
        {
            Fecha: "2014",
            Evento: "Se lanza TypeScript 1.0, con soporte para
clases, módulos y interfaces.",
            Proyectos: []
        },
        {
            Fecha: "2020",
            Evento: "Se publica TypeScript 4.0, con características
como tipado variádico y soporte para el operador de encadenamiento
opcional.",
            Proyectos: []
        }
    ]
})

db.history.insert({
    Nombre: "Smalltalk",
    Eventos: [
        {
            Fecha: "1972",
            Evento: "Alan Kay propone la idea de objetos y clases en
su tesis doctoral.",
            Proyectos: []
        },
        {
            Fecha: "1980",
```

```

        Evento: "Xerox PARC desarrolla Smalltalk-80, la primera
implementación completa del lenguaje.",
        Proyectos: []
    },
    {
        Fecha: "1996",
        Evento: "Smalltalk es estandarizado por ANSI, bajo el
nombre de 'ANSI Smalltalk'.",
        Proyectos: []
    }
]
})

db.history.insert({
    Nombre: "PHP",
    Eventos: [
        {
            Fecha: "1995",
            Evento: "Rasmus Lerdorf lanza la primera versión de PHP,
originalmente llamada 'Personal Home Page Tools'.",
            Proyectos: []
        },
        {
            Fecha: "2005",
            Evento: "Se lanza PHP 5, con mejoras significativas en
rendimiento y soporte para la programación orientada a objetos.",
            Proyectos: []
        },
        {
            Fecha: "2020",
            Evento: "Se publica PHP 8.0, con importantes mejoras de
rendimiento y nuevas características como atributos y union types.",
            Proyectos: []
        }
    ]
})

db.history.insert({
    Nombre: "Python",
    Eventos: [
        {
            Fecha: "1991",
            Evento: "Guido van Rossum publica la primera versión de
Python.",
            Proyectos: [

```

```

        {
            Nombre: "Django",
            Descripcion: "Un framework de desarrollo web de
alto nivel, desarrollado en Python. Es un proyecto de software
libre."
        }
    ]
},
{
    Fecha: "2008",
    Evento: "Se lanza Python 3.0, con importantes cambios y
mejoras que no son retrocompatibles con versiones anteriores.",
    Proyectos: []
},
{
    Fecha: "2020",
    Evento: "Se publica Python 3.9, con mejoras en sintaxis y
rendimiento, así como nuevas características como dict union y
TypeHinting.",
    Proyectos: []
}
]
}))

db.history.insert({
    Nombre: "Perl",
    Eventos: [
        {
            Fecha: "1987",
            Evento: "Larry Wall lanza Perl 1.0.",
            Proyectos: []
        },
        {
            Fecha: "2000",
            Evento: "Se lanza Perl 5.6.0, con importantes mejoras en
estabilidad y características.",
            Proyectos: []
        },
        {
            Fecha: "2015",
            Evento: "Se publica Perl 5.22.0, introduciendo mejoras en
seguridad y rendimiento.",
            Proyectos: []
        }
    ]
})

```

```
})
```

usage_examples

```
db.usage_examples.insert({
  Nombre: "C++",
  Ejemplos: [
    {
      Contexto: "Aplicaciones de alto rendimiento",
      Ejemplo: "Desarrollo de motores de juegos y software de
gráficos en 3D utilizando librerías como OpenGL."
    },
    {
      Contexto: "Sistemas embebidos",
      Ejemplo: "Programación de controladores de dispositivos y
sistemas embebidos debido a su rendimiento y capacidad de bajo
nivel."
    },
    {
      Contexto: "Desarrollo de sistemas operativos",
      Ejemplo: "C++ se utiliza en el desarrollo de sistemas
operativos como Windows y Linux debido a su eficiencia y control de
hardware."
    }
  ]
})

db.usage_examples.insert({
  Nombre: "Java",
  Ejemplos: [
    {
      Contexto: "Desarrollo web",
      Ejemplo: "Desarrollo de aplicaciones web utilizando
frameworks como Spring o Java EE para crear aplicaciones
empresariales escalables."
    },
    {
      Contexto: "Aplicaciones móviles",
      Ejemplo: "Desarrollo de aplicaciones móviles para Android
utilizando el kit de desarrollo de software (SDK) de Android con Java
como lenguaje principal."
    }
  ],
})
```

```
{
    Contexto: "Aplicaciones empresariales",
    Ejemplo: "Desarrollo de aplicaciones empresariales
seguras y fiables debido a la robustez y portabilidad de Java."
}
]
}))

db.usage_examples.insert({
    Nombre: "Ruby",
    Ejemplos: [
        {
            Contexto: "Desarrollo web",
            Ejemplo: "Desarrollo de aplicaciones web dinámicas
utilizando el framework Ruby on Rails para la creación rápida de
prototipos."
        },
        {
            Contexto: "Automatización de tareas",
            Ejemplo: "Escritura de scripts para automatizar tareas
repetitivas, como procesamiento de datos, utilizando la simplicidad y
expresividad de Ruby."
        },
        {
            Contexto: "Desarrollo de juegos",
            Ejemplo: "Desarrollo de juegos y aplicaciones multimedia
utilizando la biblioteca Gosu o frameworks como Rubygame."
        }
    ]
})

db.usage_examples.insert({
    Nombre: "Visual Basic",
    Ejemplos: [
        {
            Contexto: "Desarrollo de aplicaciones de escritorio",
            Ejemplo: "Desarrollo de aplicaciones de escritorio para
Windows utilizando el entorno de desarrollo integrado (IDE) de Visual
Studio y las librerías de Windows Forms."
        },
        {
            Contexto: "Automatización de tareas",
            Ejemplo: "Escritura de macros y scripts para automatizar
tareas en aplicaciones de Microsoft Office, como Excel y Access."
        }
    ],
```

```
        {
            Contexto: "Desarrollo de sistemas embebidos",
            Ejemplo: "Programación de sistemas embebidos y
controladores utilizando Visual Basic y .NET Micro Framework."
        }
    ]
})

db.usage_examples.insert({
    Nombre: "Visual C#",
    Ejemplos: [
        {
            Contexto: "Desarrollo de aplicaciones de escritorio",
            Ejemplo: "Desarrollo de aplicaciones de escritorio
modernas para Windows utilizando Windows Presentation Foundation
(WPF) o Universal Windows Platform (UWP).",
        },
        {
            Contexto: "Desarrollo de aplicaciones móviles",
            Ejemplo: "Desarrollo de aplicaciones móviles para Windows
Phone utilizando Visual Studio y el framework Xamarin."
        },
        {
            Contexto: "Desarrollo de juegos",
            Ejemplo: "Desarrollo de juegos utilizando la biblioteca
de juegos Unity y C# como lenguaje de scripting."
        }
    ]
})

db.usage_examples.insert({
    Nombre: "TypeScript",
    Ejemplos: [
        {
            Contexto: "Desarrollo web",
            Ejemplo: "Desarrollo de aplicaciones web modernas
utilizando Angular o React con TypeScript para una mejor organización
y mantenimiento del código."
        },
        {
            Contexto: "Desarrollo de aplicaciones empresariales",
            Ejemplo: "Desarrollo de aplicaciones empresariales
escalables utilizando Node.js con TypeScript en el lado del
servidor."
        }
    ],
})
```

```

        {
            Contexto: "Desarrollo de aplicaciones de escritorio",
            Ejemplo: "Desarrollo de aplicaciones de escritorio
multiplataforma utilizando Electron y TypeScript."
        }
    ]
})

db.usage_examples.insert({
    Nombre: "Smalltalk",
    Ejemplos: [
        {
            Contexto: "Investigación científica",
            Ejemplo: "Modelado y simulación en investigación
científica en áreas como biología, física y matemáticas utilizando
entornos de Smalltalk."
        },
        {
            Contexto: "Educación",
            Ejemplo: "Enseñanza de conceptos de programación y
desarrollo de habilidades de resolución de problemas en entornos de
aprendizaje basados en Smalltalk."
        },
        {
            Contexto: "Desarrollo de aplicaciones comerciales",
            Ejemplo: "Desarrollo de aplicaciones comerciales
personalizadas, como sistemas de gestión de inventario y
contabilidad, utilizando Smalltalk."
        }
    ]
})

db.usage_examples.insert({
    Nombre: "PHP",
    Ejemplos: [
        {
            Contexto: "Desarrollo web",
            Ejemplo: "Desarrollo de sitios web dinámicos y
aplicaciones web utilizando frameworks como Laravel, Symfony o
CodeIgniter."
        },
        {
            Contexto: "Desarrollo de sistemas de gestión de contenido
(CMS)",
            Ejemplo: "Creación de sistemas de gestión de contenido

```

personalizados utilizando WordPress o Drupal y programación en PHP para la funcionalidad específica del sitio."

```

    },
    {
        Contexto: "Desarrollo de aplicaciones de comercio
electrónico",
        Ejemplo: "Desarrollo de tiendas en línea y plataformas de
comercio electrónico utilizando Magento o WooCommerce, con
programación en PHP para personalización y funcionalidad adicional."
    }
]
})

```

```

db.usage_examples.insert({
    Nombre: "Python",
    Ejemplos: [
        {
            Contexto: "Desarrollo web",
            Ejemplo: "Desarrollo de aplicaciones web utilizando
frameworks como Django o Flask para la creación rápida y escalable de
sitios y servicios web."
        },
        {
            Contexto: "Análisis de datos y ciencia de datos",
            Ejemplo: "Análisis de datos, visualización y modelado
predictivo utilizando bibliotecas como Pandas, NumPy y SciPy en
Python."
        },
        {
            Contexto: "Automatización de tareas",
            Ejemplo: "Automatización de tareas de administración de
sistemas, procesamiento de archivos y operaciones de red utilizando
Python."
        }
    ]
})

```

```

db.usage_examples.insert({
    Nombre: "Perl",
    Ejemplos: [
        {
            Contexto: "Procesamiento de texto y archivos",
            Ejemplo: "Procesamiento de archivos de texto, extracción
de datos y manipulación de cadenas utilizando expresiones regulares y

```

```
herramientas de procesamiento de texto de Perl."
    },
    {
        Contexto: "Automatización de sistemas",
        Ejemplo: "Automatización de tareas de administración de
sistemas y redes, como configuración de servidores y gestión de
sistemas de archivos."
    },
    {
        Contexto: "Desarrollo de scripts",
        Ejemplo: "Escritura de scripts para tareas
administrativas, generación de informes y procesamiento por lotes en
sistemas Unix y Windows."
    }
]
})
```

3. Consultas de búsqueda.

A continuación, realizaremos consultas para buscar determinados valores.

Primera consulta

```
/*Buscar información sobre un lenguaje de programación
OOP específico.*/
db.language.aggregate([
  {
    $match: {
      Nombre: "Java"
    }
  }
])

/*Encontrar todos los lenguajes de programación OOP
que soportan la herencia.*/
db.features.aggregate([
  {
    $match: {
      Herencia: true
    }
  }
])
```

```
    },  
    {  
      $project: {  
        _id: 0,  
        Nombre: 1,  
        Herencia: 1  
      }  
    }  
  ])  
])
```

Segunda consulta

```
/*Buscar eventos históricos importantes de un lenguaje  
de programación OOP en particular.*/  
db.history.aggregate([  
  {  
    $match: {  
      Nombre: "Python"  
    }  
  },  
  {  
    $project: {  
      _id: 0,
```

```
        "Eventos.Fecha": 1,  
        "Eventos.Evento": 1  
    }  
}  
])
```

Tercera consulta

```
/*Encontrar ejemplos de uso de un lenguaje  
de programación OOP en proyectos de software libre.*/  
db.history.aggregate([  
    {  
        $match: {  
            "Eventos.Proyectos.Descripcion": /software libre/,  
            Nombre: "Java"  
        }  
    },  
    {  
        $project: {  
            _id: 0,  
            Nombre: 1,  
            "Eventos.Proyectos": 1  
        }  
    }  
])
```

```
    }  
  ])
```

Cuarta consulta

```
/*Buscar lenguajes de programación OOP que sean multiplataforma,  
es decir, que puedan ser ejecutados en diferentes sistemas  
operativos.*/  
db.features.aggregate([  
  {  
    $match: {  
      Multiplataforma: true  
    }  
  },  
  {  
    $project: {  
      _id: 0,  
      Nombre: 1  
    }  
  }  
])
```

Cuarta consulta

```
/*Añade una consulta personalizada.*/  
  
/*Mostrar los nombres de lenguajes OOP cuyo año de creación sea  
posterior al año 2000*/  
db.language.aggregate([  
  {  
    $match: {  
      Anyo: {  
        $gt: 2000  
      }  
    }  
  },  
  {  
    $project: {  
      _id: 0,  
      Nombre: 1  
    }  
  }  
])
```

4. Consultas avanzadas.

Por último, en cuanto a consultas se trata, realizaremos diversas consultas más complejas que las anteriores. En estas utilizaremos operadores aritméticos, agrupaciones, limitadores, etc.

Primera consulta

```
/*Calcular el número total de lenguajes de programación OOP en la
base de datos.*/

db.language.aggregate([
  {
    $group: {
      _id: null,
      NumeroLenguajes: {
        $sum: 1
      }
    }
  }
])
```

Segunda consulta

```
/*Encontrar el lenguaje de programación
OOP con más características registradas.*/
db.features.aggregate([
  {
    $project: {
      Nombre: 1,
      totalCaracteristicas: {
        $size: {
          $filter: {
            input: { $objectToArray: "$$ROOT" },
            cond: { $eq: ["$$this.v", true] }
          }
        }
      }
    }
  },
  {
    $sort: { totalCaracteristicas: -1 }
  },
  {
    $limit: 1
  }
])
```

```
}  
[])
```

Tercera consulta

```
/*Agrupar los lenguajes de programación OOP  
por año de creación y contar cuántos hay en cada año.*/  
  
db.language.aggregate([  
  {  
    $group: {  
      _id: "$Anyo",  
      nombres: {  
        $push: "$Nombre"  
      },  
      total: {  
        $sum: 1  
      }  
    }  
  }  
])
```

Cuarta consulta

```
/*Calcular el promedio de eventos históricos por lenguaje de
programación OOP.*/
db.history.aggregate([
  {
    $unwind: "$Eventos"
  },
  {
    $group: {
      _id: "$Nombre",
      totalEventos: {
        $sum: 1
      }
    }
  },
  {
    $project: {
      promedioEventos: {
        $avg: "$totalEventos"
      }
    }
  }
])
```

Quinta consulta

```
/*Encontrar los lenguajes de programación
OOP con los ejemplos de uso más extensos y mostrar los tres
primeros.*/
db.usage_examples.aggregate([
  {
    $unwind: "$Ejemplos"
  },
  {
    $project: {
      _id: "$Nombre",
      LongitudEjUso: {
        $strLenCP: "$Ejemplos.Ejemplo"
      }
    }
  },
  {
    $sort: {
      LongitudEjUso: -1
    }
  },
  {
    $limit: 3
  }
])
```

```
}  
])
```

Sexta consulta

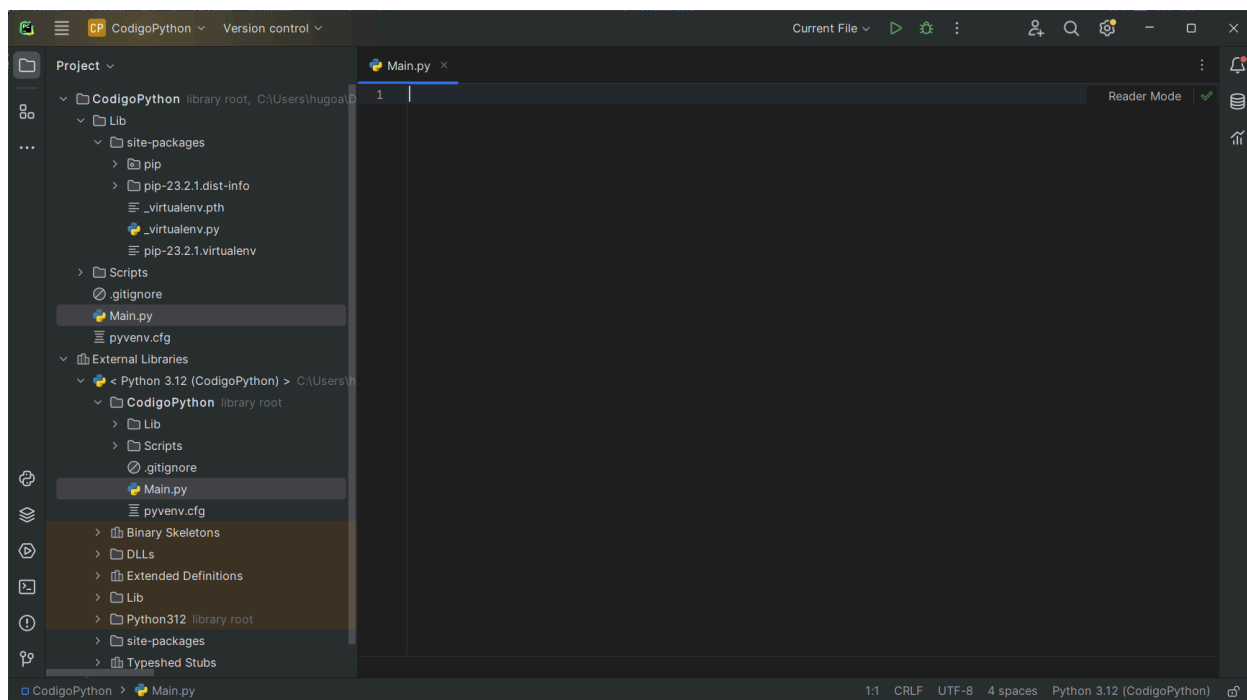
```
/*Añade una agregación personalizada.*/  
/*Indica cuál es el evento más antiguo  
y cuál es el más nuevo de cada Lenguaje de OOP.*/  
db.history.aggregate([  
  {  
    $unwind: "$Eventos"  
  },  
  {  
    $group: {  
      _id: "$Nombre",  
      masAntiguo: {  
        $min: "$Eventos.Fecha"  
      },  
      masNuevo: {  
        $max: "$Eventos.Fecha"  
      }  
    }  
  },  
  {  
    $project: {  
      _id: 1,  
      masAntiguo: 1,  
      masNuevo: 1  
    }  
  }  
])
```

```
{  
  $project: {  
    masAntiguo: 1,  
    masNuevo: 1  
  }  
}  
])
```

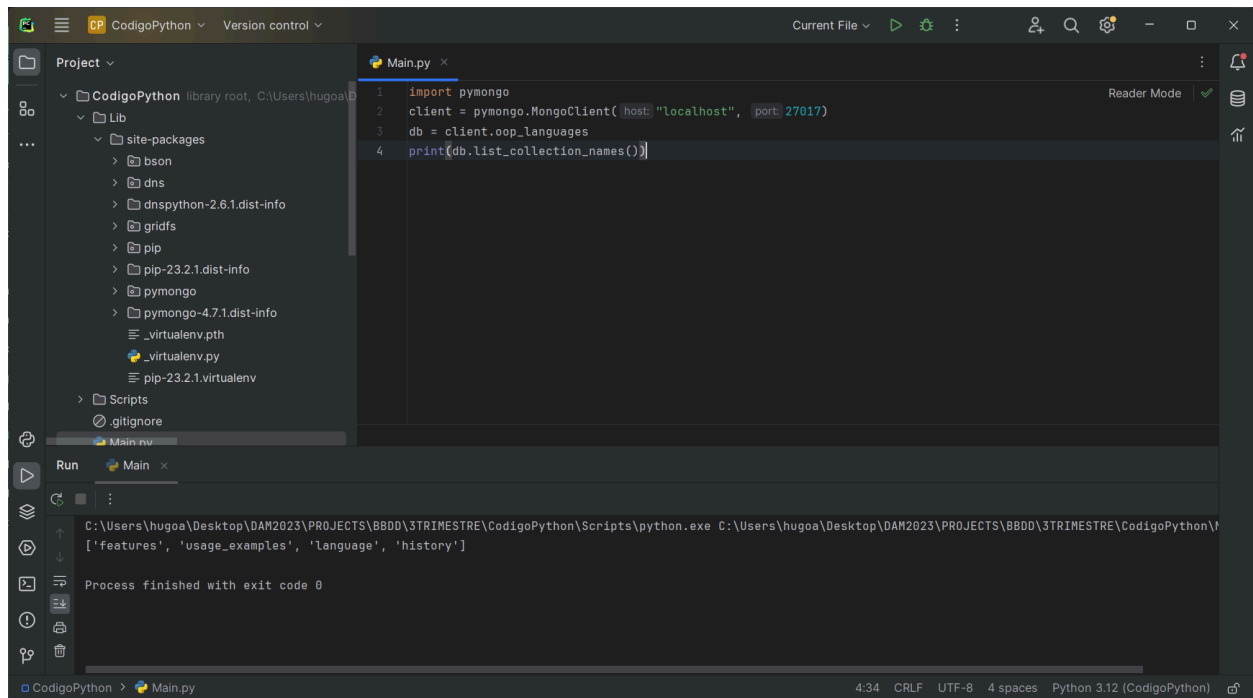
5. Conexión a Python.

El último paso a realizar en este proyecto será conectar nuestra base de datos a nuestro proyecto en PyCharm.

Primero, crearemos nuestro proyecto en PyCharm:



Una vez creado, lo conectaremos a nuestra base de datos.



The screenshot shows the CodigoPython IDE interface. The left sidebar displays the project structure under 'Project' and 'Lib' folders. The main editor window shows a file named 'Main.py' with the following Python code:

```
1 import pymongo
2 client = pymongo.MongoClient(host="localhost", port=27017)
3 db = client.oop_languages
4 print(db.list_collection_names())
```

The bottom panel shows the output of the script, indicating that the process finished with exit code 0. The output text is:

```
C:\Users\hugoa\Desktop\DAM2023\PROYECTS\BBDD\3TRIMESTRE\CodigoPython\Scripts\python.exe C:\Users\hugoa\Desktop\DAM2023\PROYECTS\BBDD\3TRIMESTRE\CodigoPython\Main.py
['features', 'usage_examples', 'language', 'history']
```

The status bar at the bottom indicates the file encoding (CRLF, UTF-8), indentation (4 spaces), and the Python version (Python 3.12 (CodigoPython)).

A continuación, realizaremos una consulta de búsqueda:

```
import pymongo

client = pymongo.MongoClient("localhost", 27017)

db = client.oop_languages

mycol = db["language"]

myquery = { "Nombre": "Java"}

mydoc = mycol.find(myquery)

for x in mydoc:
    print(x)
```

Y una agregación:

```
db.language.insert_one({"Nombre": "Simula", "Herencia": "true",
"Multiplataforma": "false", "Polimorfismo": "true",
"Encapsulamiento": "true"})
```