MONGODB

1. Instalar extension MongoDB for VS Code

2. Descargar docker

3. Crear el siguiente archivo docker-compose.yaml en una carpeta

```
version: '3.8'
services:
  mongodb:
    image: mongo:latest
    container_name: mongodb_container
    environment:
      MONGO_INITDB_ROOT_USERNAME: root
      MONGO_INITDB_ROOT_PASSWORD: 123456
    volumes:
      - mongodb_data_container:/data/db
    ports:
      - 27017:27017
volumes:
  mongodb_data_container:
```

3. Entrar en dicha carpeta y ejecutar el siguiente comando

```
docker compose up --build -d
```

4. En la extension de MongoDB conectarse con URL y introducir la siguiente url:

```
mongodb://root:123456@localhost:27017/
```

XSLT

1. Instalar extension XSLT

2. Descargar e instalar java

3. Descargar saxon-he

4. En ajustes de xslt poner el valor saxon jar a la direccion del archivo de saxon-he

5. Ir a Terminal -> Run Build Task y elejir xslt: Saxon Transform

6. I al archivo .vscode/tasks.json e remplazar por el siguiente json

```
{
    "version": "2.0.0",
    "tasks": [
        {
            "type": "xslt",
            "label": "xslt: Saxon Transform (New)",
            "saxonJar": "${config:XSLT.tasks.saxonJar}",
            "xsltFile": "${command:xslt-xpath.pickXsltFile}",
            "xmlSource": "${command:xslt-xpath.pickXmlSourceFile}",
            "resultPath": "${command:xslt-xpath.pickResultFile}",
            "allowSyntaxExtensions40": "off",
            "messageEscaping": "adaptive",
            "group": {
                "kind": "build",
                "isDefault": true
            },
            "problemMatcher": [
                "$saxon-xslt"
            ]
        },
        {
            "type": "xslt",
            "label": "xslt: xslt: Saxon Transform (New)",
            "saxonJar": "${config:XSLT.tasks.saxonJar}",
            "xsltFile": "${command:xslt-xpath.pickXsltFile}",
            "xmlSource": "${command:xslt-xpath.pickXmlSourceFile}",
            "resultPath": "${command:xslt-xpath.pickResultFile}",
            "allowSyntaxExtensions40": "off",
            "messageEscaping": "adaptive",
            "group": {
                "kind": "build",
                "isDefault": true
            },
            "problemMatcher": [
                "$saxon-xslt"
            ]
        }
    ],
    "inputs": [
        /*
        1. the specialist 'pickFile' command equivalents for the inputs defined
here are:
            "xsltFile": "${command:xslt-xpath.pickXsltFile}",             //
can select from current file - if xslt - or recently used xslt files
            "xmlSource": "${command:xslt-xpath.pickXmlSourceFile}",       //
can select from current file or recently used stage1 source files
            "xmlSource": "${command:xslt-xpath.pickStage2XmlSourceFile}", //
can select from recently used stage2 source files or stage1 result files
            "resultPath": "${command:xslt-xpath.pickResultFile}",         //
can save to recently used stage1 result files
```

```
            "resultPath": "${command:xslt-xpath.pickStage2ResultFile}",    //
can save to recently used stage2 result files

        2. these inputs invoke the 'pickFile' command with args for custom
behaviour:
        */
        {
            /* --- Usage: ---
            "xmlSource": "${input:xmlFile}",
            */
            "id": "xmlFile",
            "type": "command",
            "command": "xslt-xpath.pickFile",
            "args": {"label": "Select XML File", "extensions": ["xml", "xhtml",
"svg"] }
        },
        {
            /* --- Usage: ---
            "xmlSource": "${input:xmlFile2}",
            */
            "id": "xmlFile2",
            "type": "command",
            "command": "xslt-xpath.pickFile",
            "args": {"label": "Select Stage1 XML File", "extensions": ["xml",
"xhtml", "svg"], "prevStageLabel": "Select Result File", "prevStageGroup":
"recent files from previous stage" }
        },
        {
            /* --- Usage: ---
            "xsltFile": "${input:xsltFile}",
            */
            "id": "xsltFile",
            "type": "command",
            "command": "xslt-xpath.pickFile",
            "args": {"label": "Select XSLT Stylesheet", "extensions": ["xsl",
"xslt"] }
        },
        {
            /* --- Usage: ---
            "resultPath": "${input:resultFile}",
            */
            "id": "resultFile",
            "type": "command",
            "command": "xslt-xpath.pickFile",
            "args": {"label": "Select Result File", "isResult": true }
        }
    ]
}
```

6. Para ejecutar: Terminal -> Run Task