# Projeto de Bases de Dados (CC2005)

## 1. Elementos do grupo

**Grupo nº** G1G

|  |  |
| --- | --- |
| **Nº mecanográfico** | **Nome** |
| 202305085 | Afonso Castro Coelho |
| 202305187 | Diogo do Amaral Azevedo |
| 202305229 | Miguel Martins Ferreira Camocho de Carvalho |

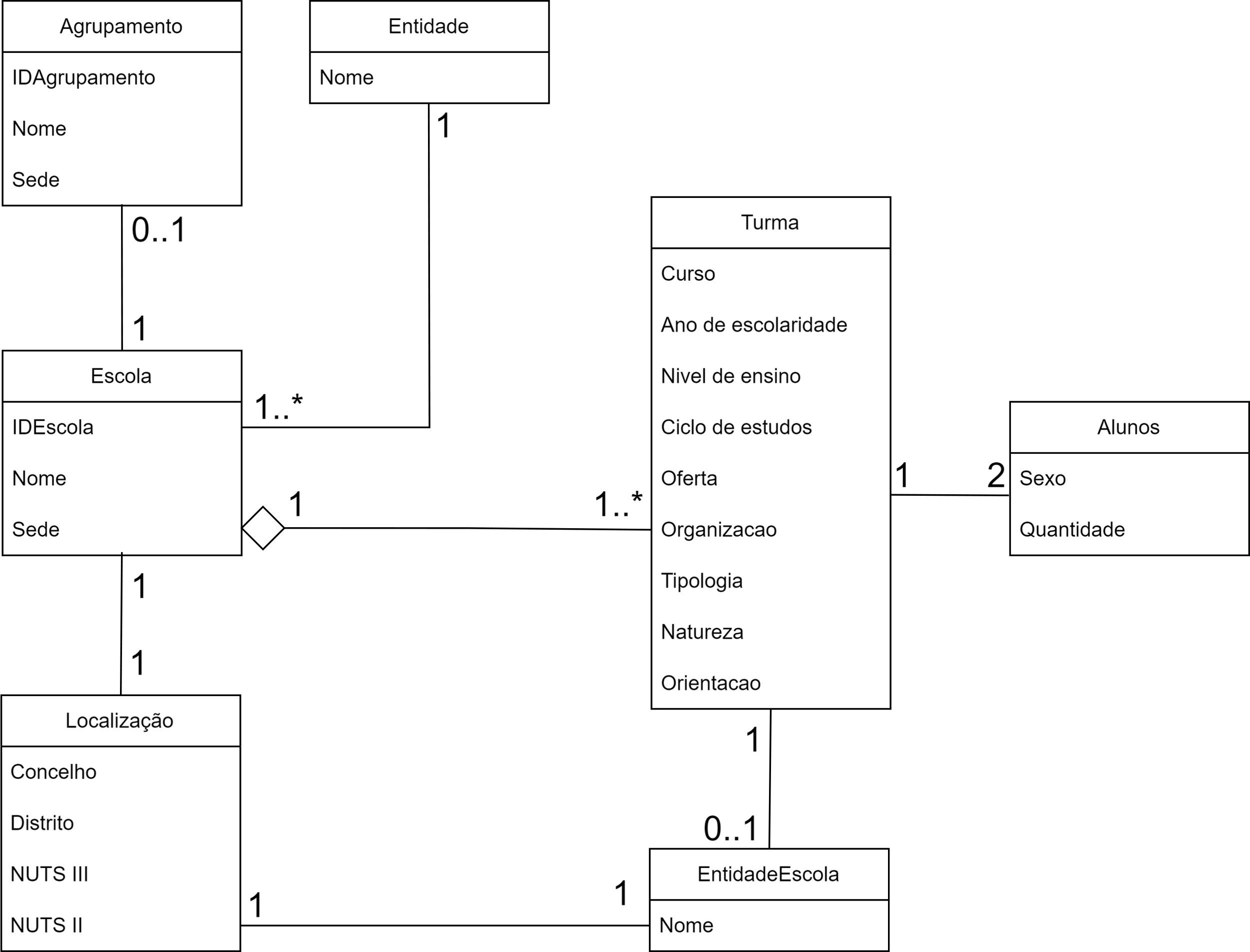
## 2. Universo considerado e modelo de classes UML

**Tema:** Alunos Matriculados

O dataset fornecido (com exatamente 82606 registos) descreve o universo dos alunos matriculados no ano letivo de 2017/2018, a nível de Portugal Continental. Relativamente a estes alunos, apresenta-se a escola ou entidade a que pertence cada conjunto de alunos, o agrupamento de tal escola se existir e a localidade em que está inserida, bem como outras informações características a grupos de alunos.

Assim sendo, criamos as seguintes 7 classes para representar o universo em questão:

* **Agrupamento:** relativo ao agrupamento de uma escola (cada escola tem um agrupamento diferente neste dataset).
* **Escola:** relativo a uma escola. Está associada a uma entidade detentora e a uma localidade. É composta por turmas.
* **Entidade:** relativa à entidade detentora de uma escola (cada escola tem uma diferente).
* **Localidade:** relativa à área geográfica de uma escola ou entidade-escola. Guarda o concelho, distrito, NUTS III e NUTS II.
* **Turma:** relativo a uma turma e às suas características específicas (ano, ciclo, etc.). É composta por alunos.
* **Alunos:** relativo a um grupo de alunos com as mesmas características (sexo neste caso).
* **EntidadeEscola:** relativa às entidades que, apesar de lecionadoras, não são uma escola. Este caso específico será melhor explicado no capítulo seguinte.



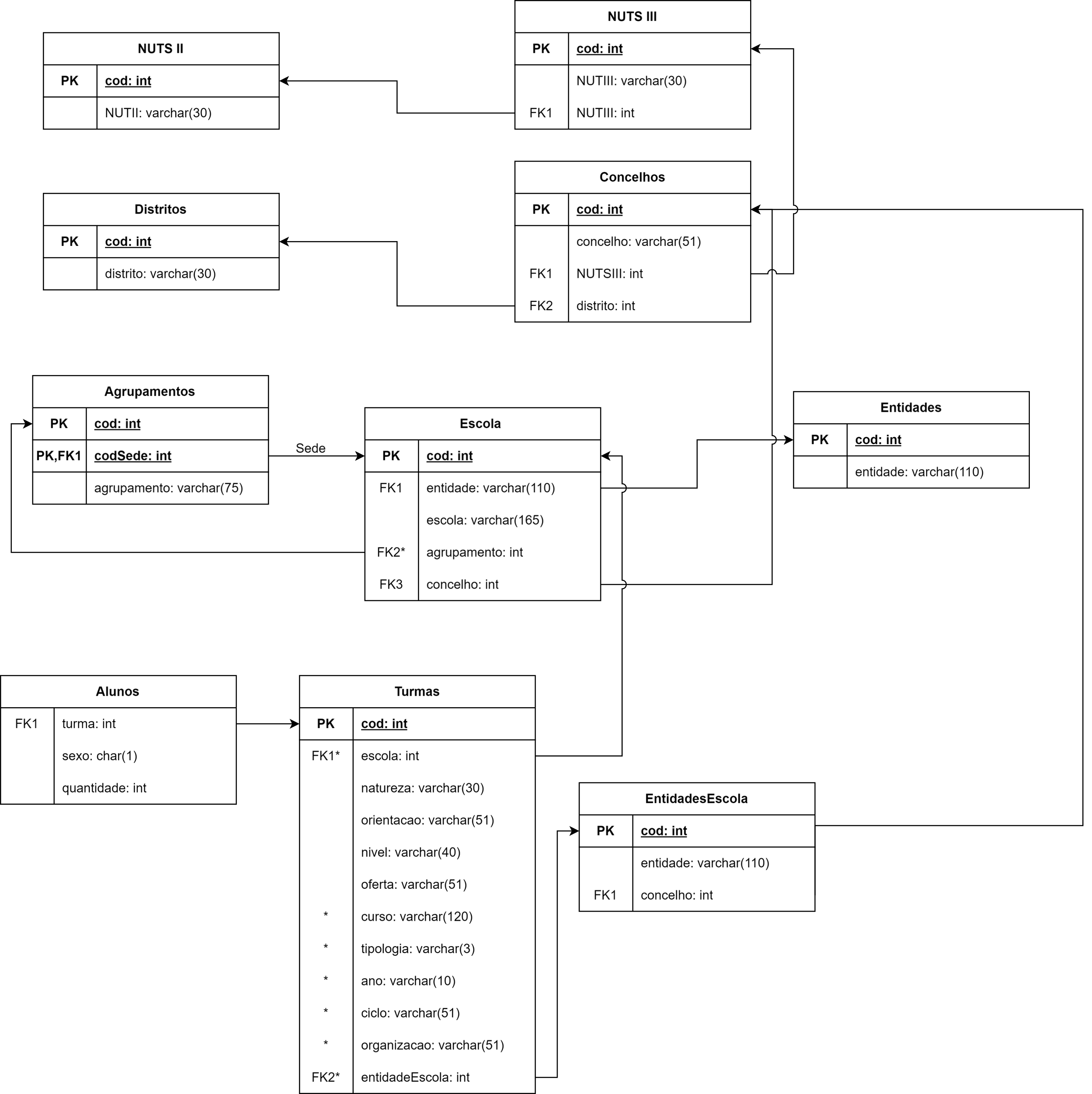
## 3. Modelo relacional

Atentando às localidades, podemos ver que 4 classes foram criadas: Concelhos, Distritos, NUTS III e NUTS II. Vale notar que estão ligadas como apresentado porque, apesar de um dado concelho só fazer parte de uma NUTS III, tal não é o caso para os Distritos e as NUTS II. Isto é, existem distritos que pertencem a mais que uma NUTS II. Assim, como está garantido que cada NUTS III pertence a uma só NUTS II, optamos por tornar NUTS II acessível apenas através de NUTS III.

O próximo ponto de interesse seria provavelmente a classe Turmas. À primeira vista, pode parecer estranha a quantidade de atributos aqui inseridos, nomeadamente de atributos anuláveis. No entanto, existem variados casos em que, um campo estar preenchido significa deixar outro nulo. Um exemplo seria o nível “Educação pré-escolar”, que não tem tanto ciclo como ano.

Ainda sobre as Turmas, optamos por lhe associar uma classe Alunos. Esta tabela teria duas linhas por turma (ambas identificadas com a chave da turma) que guardam o número de mulheres da turma, bem como o número de homens da mesma.

Finalmente, e decerto não menos importante, atentemos na classe EntidadesEscola. A sua existência é fulcral para o correto funcionamento desta base de dados, tudo devido ao seguinte facto: existem turmas (conjuntos de alunos) que não pertencem a nenhuma escola. Em vez disso, estão associadas a uma entidade educativa, como é o caso da “Santa Casa da Misericórdia de Lisboa”. Assim, após cuidada consideração e várias versões deste modelo, optamos pelo seguinte sistema: uma turma terá 2 *foreign keys (FK)*, escola e escolaEntidade. Quando uma turma pertencer a uma escola, popula-se escola com o código dessa escola, deixando-se o campo entidadeEscola nulo. Caso contrário, faz-se o oposto.

****

## 3. Povoamento de tabelas

## 3.1 Fontes de dados

Temos como fontes dos dados o ficheiro Excel fornecido, na sua totalidade exceto as colunas Ano Letivo, NUTS II (2002) e NUTS III (2002).

## 3.2 Metodologia do povoamento

Já estando construído o modelo relacional da base de dados, começamos por criar um ficheiro AlunosMatriculados.db bem como as suas tabelas, tudo em SQLiteStudio.

De seguida, num programa python, e com auxílio das bibliotecas *pandas* e *sqlite3*, povoamos

a base de dados como descrito no ficheiro *popular.py*.

Note o ficheiro está em docstring na sua totalidade pois foi feito por partes, partes essas que foram sendo ou não utilizadas em várias fases.

## 3.3 Número de linhas por tabela

|  |  |
| --- | --- |
| **Nome da tabela** | **Nº de entradas** |
| Agrupamentos | 713 |
| Escolas | 8072 |
| Entidades | 8387 |
| EntidadesEscola | 315 |
| Turmas | 42859 |
| Alunos | 82605 |
| Concelhos | 278 |
| Distritos | 18 |
| NUTS\_III | 23 |
| NUTS\_II | 5 |

## 4. Interrogações SQL

## 4.1 Pergunta 1

#### 4.1.1 Enunciado

Em cada concelho, quantas escolas que não pertencem a um agrupamento existem?

**Utilize vários tipos de JOIN.**Ordene por concelho.

#### 4.1.2 Resolução

SELECT

    concelhos.concelho,

    COUNT(escolas.cod) as "Nº Escolas"

FROM concelhos

JOIN escolas ON escolas.concelho = concelhos.cod

LEFT JOIN agrupamentos ON agrupamentos.cod = escolas.agrupamento

WHERE escolas.agrupamento IS NULL

GROUP BY concelhos.cod;

#### 4.1.3 Resultado

|  |  |
| --- | --- |
| **Concelhos** | **Nº Escolas** |
| Águeda | 19 |
| Albergaria-a-Velha | 9 |
| Anadia | 12 |
| Arouca | 1 |
| … | … |

## 4.2 Pergunta 2

#### 4.2.1 Enunciado

#### Quais os cursos que um aluno do Porto ou Lisboa pode frequentar? Ordene por distrito e por curso.

#### 4.2.2 Resolução

SELECT DISTINCT

d.distrito AS Distrito,

t.oferta AS Curso

FROM Alunos a

JOIN Turmas t ON a.turma = t.cod

JOIN Escolas e ON t.escola = e.cod

JOIN Concelhos c ON e.concelho = c.cod

JOIN Distritos d ON c.distrito = d.cod

LEFT JOIN EntidadesEscola ee ON t.entidadeEscola = ee.cod

LEFT JOIN Concelhos cee ON ee.concelho = cee.cod

LEFT JOIN Distritos dee ON cee.distrito = dee.cod

WHERE

(

d.distrito IN ('Porto', 'Lisboa')

OR dee.distrito IN ('Porto', 'Lisboa')

)

AND t.oferta NOT NULL

GROUP BY d.distrito, t.oferta

ORDER BY d.distrito ASC, t.oferta ASC;

#### 4.2.3 Resultado

|  |  |
| --- | --- |
| **Distrito** | **Curso** |
| Lisboa | Cursos CEF |
| Lisboa | Cursos CET |
| … | … |
| Porto | Cursos profissionais |
| Porto | Cursos vocacionais |
| … | … |

## 4.3 Pergunta 3

#### 4.3.1 Enunciado

De todas as escolas sede, escolhe aquelas que não são nem do Centro nem do Alentejo (NUTSII).

#### 4.3.2 Resolução

SELECT

    escolas.escola AS NomeEscola

FROM escolas

JOIN agrupamentos ON escolas.cod = agrupamentos.codSede

JOIN concelhos ON escolas.concelho = concelhos.cod

JOIN NUTS\_III ON concelhos.NUTSIII = NUTS\_III.cod

JOIN NUTS\_II ON NUTS\_III.NUTII = NUTS\_II.cod

WHERE NUTS\_II.NUTII NOT IN ('Centro', 'Alentejo');

#### 4.3.3 Resultado

|  |
| --- |
| **NomeEscola** |
| Escola Básica e Secundária de Ínfias, Vizela |
| Escola Secundária de Santo André, Barreiro |
| Escola Secundária de Alcochete |
| Escola Básica e Secundária de Santo António, Barreiro |
| … |

## 4.4 Pergunta 4

#### 4.4.1 Enunciado

Qual é o distrito com menos turmas de nível de ensino secundário? Selecione o nome do distrito e o número de turmas.

#### 4.4.2 Resolução

SELECT

sub.distrito AS Distrito,

MIN(sub.num) AS "Nº Turmas"

FROM (

SELECT

d.distrito AS distrito,

COUNT(t.cod) AS num

FROM distritos d

JOIN concelhos c ON c.distrito = d.cod

JOIN escolas e ON e.concelho = c.cod

JOIN turmas t ON t.escola = e.cod

WHERE t.nivel LIKE "Ensino Secundário"

GROUP BY d.cod, d.distrito

) AS sub;

#### 4.4.3 Resultado

|  |  |
| --- | --- |
| **Concelho** | **Nº Turmas** |
| Portalegre | 131 |

## 4.5 Pergunta 5

#### 4.5.1 Enunciado

Selecione o nome do concelho, a escola e o número de turmas como "Nº de turmas" das turmas que a organização contém "S3" no nome.

Ordene por número de turmas descendente e por nome dos concelhos.

#### 4.5.2 Resolução

SELECT

        c.concelho,

        e.escola,

        COUNT(t.cod) AS "Nº de turmas"

FROM Turmas t

JOIN Escolas e ON t.escola = e.cod

JOIN Concelhos c ON e.concelho = c.cod

WHERE t.organizacao LIKE '%S3%'

GROUP BY

    c.concelho,

    e.escola

ORDER BY

    "Nº de turmas" DESC,

    c.concelho ASC;

#### 4.5.3 Resultado

|  |  |  |
| --- | --- | --- |
| **Concelho** | **Escola** | **Nº de Turmas** |
| Amadora | Escola Secundária Seomara da Costa Primo, Amadora | 6 |
| Loures | Escola Secundária José Afonso, Loures | 6 |
| Amadora | Escola Secundária da Amadora | 5 |
| Setúbal | Escola Secundária Sebastião da Gama, Setúbal | 5 |
| … | … | … |

## 4.6 Pergunta 6

#### 4.6.1 Enunciado

Selecione todos os agrupamentos em concelhos com mais de 30 escolas de natureza pública e tipologia "EB" ordena por agrupamento

#### 4.6.2 Resolução

SELECT

    concelhos.concelho,

    agrupamentos.agrupamento

FROM agrupamentos

JOIN escolas ON escolas.agrupamento = agrupamentos.cod

JOIN concelhos ON concelhos.cod = escolas.concelho

JOIN turmas ON turmas.escola = escolas.cod

WHERE turmas.natureza = 'Público' AND turmas.tipologia = 'EB'

GROUP BY concelhos.concelho, agrupamentos.agrupamento

HAVING COUNT(escolas.cod) > 30

ORDER BY

    concelhos.concelho,

    agrupamentos.agrupamento;

#### 4.6.3 Resultado

|  |  |
| --- | --- |
| **Concelho** | **Agrupamento** |
| Abrantes | Escolas Nº 1 de Abrantes |
| Albergaria-a-Velha | Escolas de Albergaria-a-Velha |
| Albufeira | Escolas de Ferreiras, Albufeira |
| Alcanena | Escolas de Alcanena |
| … | … |

## 4.7 Pergunta 7

#### 4.7.1 Enunciado

Quais os 5 distritos com menor número de raparigas por turma em média? Considere apenas turmas pertencentes a escolas. Ordene do menor número de raparigas para o maior.

#### 4.7.2 Resolução

WITH MediaRaparigasPorDistrito AS (

SELECT

d.distrito,

AVG(a.quantidade) AS media\_raparigas

FROM distritos d

JOIN concelhos c ON d.cod = c.distrito

JOIN escolas e ON c.cod = e.concelho

JOIN turmas t ON e.cod = t.escola

JOIN alunos a ON t.cod = a.turma

WHERE a.sexo = 'Mulheres'

GROUP BY d.distrito

),

DistritosOrdenados AS (

SELECT

distrito,

media\_raparigas,

ROW\_NUMBER() OVER (ORDER BY media\_raparigas ASC) AS rank

FROM MediaRaparigasPorDistrito

)

SELECT

distrito AS Distrito,

media\_raparigas AS "Média de Raparigas"

FROM DistritosOrdenados

WHERE rank <= 5

ORDER BY rank;

#### 4.7.3 Resultado

|  |  |
| --- | --- |
| **Distrito** | **Média de Raparigas** |
| Guarda | 11.60393258426966 |
| Bragança | 12.70188679245283 |
| Castelo Branco | 12.768779342723 |
| Beja | 12.82193548387097 |
| Portalegre | 13.29128014842301 |

## 4.8 Pergunta 8

#### 4.8.1 Enunciado

Qual o nome da escola com o maior número de alunos em distritos com mais de 1000 escolas?

Selecione o nome do distrito, o nome da escola e a quantidade de alunos.

#### 4.8.2 Resolução

WITH c1 AS (

        SELECT

            distritos.distrito,

            SUM(alunos.quantidade) AS max\_quantidade,

            escolas.escola

        FROM distritos

        JOIN concelhos ON concelhos.distrito = distritos.cod

        JOIN escolas ON escolas.concelho = concelhos.cod

        JOIN turmas ON turmas.escola = escolas.cod

        JOIN alunos ON alunos.turma = turmas.cod

        GROUP BY escolas.escola

        HAVING distritos.cod IN (

            SELECT d.cod

            FROM distritos d

            WHERE(

                SELECT COUNT(\*)

                FROM concelhos c

                JOIN escolas e ON e.concelho = c.cod

                WHERE c.distrito = d.cod

            ) > 1000

    )    )

    SELECT

        c1.distrito,

        c1.escola,

        MAX(c1.max\_quantidade) AS alunos

    FROM c1;

#### 4.8.3 Resultado

|  |  |  |
| --- | --- | --- |
| **Distrito** | **Escola** | **Alunos** |
| Lisboa | Salesianos de Lisboa - Colégio Oficinas de São José | 2295 |

## 4.9 Pergunta 9

#### 4.9.1 Enunciado

Faz uma classificação qualitativa das escolas quanto ao número de turmas e média de alunos:

* Se o número de turmas < média de turmas por escola 🡪 Tem poucas turmas
* Se o número de turmas >= média de turmas por escola e o número de alunos < média de alunos por escola (globalmente) 🡪 Poucos alunos por escola
* Se o número de turmas >= média de turmas por escola e o número de alunos >= média de alunos por escola (globalmente) 🡪 A escola esta sobrepopulada

O resultado deve ter o nome da escola a quantidade de alunos e a classificação obtida ordenada por nome da escola

#### 4.9.2 Resolução

WITH EstatisticasEscolas AS (

SELECT

escolas.escola AS NomeEscola,

COUNT(DISTINCT turmas.cod) AS NumeroTurmas,

SUM(alunos.quantidade) AS NumeroAlunos

FROM alunos

JOIN turmas ON turmas.cod = alunos.turma

JOIN escolas ON escolas.cod = turmas.escola

GROUP BY escolas.escola

),

MediasGlobais AS (

SELECT

AVG(NumeroTurmas) AS MediaTurmasPorEscola,

AVG(NumeroAlunos) AS MediaAlunosPorEscola

FROM EstatisticasEscolas

)

SELECT

ee.NomeEscola,

ee.NumeroAlunos,

CASE

WHEN ee.NumeroTurmas < mg.MediaTurmasPorEscola THEN 'Tem poucas turmas'

WHEN ee.NumeroTurmas >= mg.MediaTurmasPorEscola AND ee.NumeroAlunos < mg.MediaAlunosPorEscola THEN 'Poucos alunos'

WHEN ee.NumeroTurmas >= mg.MediaTurmasPorEscola AND ee.NumeroAlunos >= mg.MediaAlunosPorEscola THEN 'A escola esta sobrepopulada'

END AS Classificacao

FROM EstatisticasEscolas ee

CROSS JOIN MediasGlobais mg

ORDER BY ee.NomeEscola;

#### 4.9.3 Resultado

|  |  |  |
| --- | --- | --- |
| **NomeEscola** | **NumeroAlunos** | **Classificacao** |
| A Creche Sempre em Flor | 100 | Tem poucas turmas |
| A Escolinha Encantada | 45 | Tem poucas turmas |
| A Escolinha da Estação Cooperativa de Solidariedade Social | 19 | Tem poucas turmas |
| A Mamã Coelha | 19 | Tem poucas turmas |
| … | … | … |

## 4.10 Pergunta 10

#### 4.10.1 Enunciado

Recolha a soma de todos os alunos 'Homens' e 'Mulheres', separadamente, de todas escolas sede de um concelho.

CONDIÇÃO: Caso o concelho seja situado num distrito que pertença a mais do que uma NUTSII, registe paralelamente a soma dos alunos 'Homens' e 'Mulheres' das turmas que tenham nível 'Ensino Básico'.

Deve retornar o Nome do Concelho, a Soma dos Homens com a condição aplicada, a Soma dos Homens sem a condição aplicada, a Soma das Mulheres com a condição aplicada e a Soma das Mulheres sem a condição aplicada.

Ordene por concelho.

#### 4.10.2 Resolução

**WITH ConcelhosComMaisDeUmNUTSII AS(**

WITH NUTSII\_Por\_Distrito AS (

SELECT

d.cod AS DistritoCod,

CASE

WHEN COUNT(DISTINCT NUTS\_II.NUTII) > 1 THEN TRUE

ELSE FALSE

END AS MaisDeUmNUTSII

FROM Distritos d

JOIN Concelhos c ON c.distrito = d.cod

JOIN NUTS\_III ON NUTS\_III.cod = c.NUTSIII

JOIN NUTS\_II ON NUTS\_II.cod = NUTS\_III.NUTII

GROUP BY d.cod

)

SELECT

d.distrito,

c.concelho as concelho,

c.cod as concelhoCod,

n.MaisDeUmNUTSII AS MaisDeUmNUTSII

FROM Concelhos c

JOIN Distritos d ON c.distrito = d.cod

JOIN NUTSII\_Por\_Distrito n ON n.DistritoCod = d.cod

ORDER BY

d.distrito, c.concelho

),

AlunosPorConcelho AS (

SELECT

c.Concelho as NomeConcelho,

SUM(CASE WHEN a.sexo like 'Homens' and ((c.MaisDeUmNUTSII = TRUE AND t.nivel like 'Ensino básico') OR c.MaisDeUmNUTSII = FALSE) THEN a.quantidade ELSE 0 END) AS SomaHomensComCondicao,

SUM(CASE WHEN a.sexo like 'Mulheres' and ((c.MaisDeUmNUTSII = TRUE AND t.nivel like 'Ensino básico') OR c.MaisDeUmNUTSII = FALSE) THEN a.quantidade ELSE 0 END) AS SomaMulheresComCondicao,

SUM(CASE WHEN a.sexo like 'Homens' THEN a.quantidade ELSE 0 END) AS SomaHomensSemCondicao,

SUM(CASE WHEN a.sexo like 'Mulheres' THEN a.quantidade ELSE 0 END) AS SomaMulheresSemCondicao

FROM ConcelhosComMaisDeUmNUTSII c

JOIN Escolas e ON e.concelho = c.ConcelhoCod

JOIN Agrupamentos ag ON ag.codSede = e.cod

JOIN Turmas t ON t.escola = e.cod

JOIN Alunos a ON a.turma = t.cod

GROUP BY c.Concelho

)

SELECT

NomeConcelho,

SomaHomensComCondicao,

SomaHomensSemCondicao,

SomaMulheresComCondicao,

SomaMulheresSemCondicao

FROM AlunosPorConcelho

JOIN ConcelhosComMaisDeUmNUTSII c on c.concelho=AlunosPorConcelho.NomeConcelho

ORDER BY NomeConcelho;

#### 4.10.3 Resultado

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **NomeConcelho** | **SomaHomensComCondicao** | **SomaHomensSemCondicao** | **SomaMulheresComCondicao** | **SomaMulheresSemCondicao** |
| Abrantes | 622 | 1090 | 542 | 1000 |
| Aguiar da Beira | 116 | 176 | 103 | 171 |
| Alandroal | 120 | 120 | 130 | 130 |
| Albergaria-a-Velha | 345 | 624 | 345 | 601 |
| … | … | … | … | … |

## 5. Aplicação Python

|  |  |
| --- | --- |
| **“Endpoint”** | **Funcionalidade** |
| **/** | Página de entrada |
| /explicação/<tema> | Explicação breve do tópico de SQL <tema> |
| /pergunta/<n> | Formulação da pergunta <n>, apresentação e explicação da sua resolução e visualização do output |
| /distritos | Listagem de todos os distritos |
| /distritos/<codDistrito> | Dados do distrito com código <codDistrito> e listagem dos concelhos nesse distrito |
| /concelhos | Listagem de todos os concelhos |
| /concelhos/<codConcelho> | Dados do concelho com código <codConcelho> e listagem das escolas nesse concelho |
| /escolas | Listagem de todas as escolas |
| /escolas/<codEscola> | Dados da escola com código <codEscola> e listagem das turmas nessa escola |
| /escolas/<codEscola>/turmas/<codTurma> | Dados da turma com código <codTurma> da escola com código <codEscola> |
| /agrupamentos | Listagem de todos os agrupamentos |
| /agrupamentos/<codAgrupamento> | Dados do agrupamento com código <codAgrupamento> |
| /AlunosMatriculados | Página dedicada a pesquisa de escolas de acordo com filtros ajustados pelo utilizador |

## 6. Conclusão

Apesar da natureza complexa do dataset e do teu grande tamanho, consideramos que conseguimos modelar e popular uma base de dados representativa do universo e, ao mesmo tempo, intuitiva na sua utilização. Cremos também ter desenvolvido um website que é facilmente navegado, e visualmente apelativo dentro do que a nossa habilidade e o tempo nos permite. Finalmente, preenchemos tal website não só com perguntas escritas e resolvidas por nós, bem como outras funcionalidades já descritas acima.

## 7. Referências

*ChatGPT*. (n.d.). ChatGPT. [https://chatgpt.com](https://chatgpt.com/)

*GitHub Copilot* (Pro, version 1.248.0) [Computer software]. (2021). GitHub.

FCUP. (n.d.). *Bases de Dados (Slides)*. Moodle 24/25 - UP.