

Pretende-se com este projeto que desenvolvam uma aplicação que permita gerir a informação de vários países relacionada com a pandemia COVID-19 tal como população, idade média, nº de casos, nº de óbitos, índices de fatores de risco como idade, diabetes, etc.

A informação encontra-se no ficheiro de texto: `owid-covid-data.csv`

Usando a *Java Collection Framework* desenvolva as classes necessárias para implementar da forma **mais eficiente possível** as seguintes funcionalidades:

1. Carregar e guardar a informação relativa aos países e respetivos dados da pandemia COVID-19 a partir do ficheiro de texto fornecido.
2. Apresentar uma lista de países ordenados por ordem crescente do número mínimo de dias que foi necessário para atingir os **50.000 casos positivos**.

iso_code	continent	location	date	total_cases	mindays
CHN	Asia	China	2020-02-13	59865	43 days
ITA	Europe	Italy	2020-03-22	53578	81 days
ESP	Europe	Spain	2020-03-25	57506	84 days
USA	North America	United States	2020-03-25	55231	84 days
DEU	Europe	Germany	2020-03-29	52547	88 days
FRA	Europe	France	2020-04-01	52128	91 days
...					
DZA	África	Algeria	2020-09-22	50023	265 days
CZE	Europe	Czech Republic	2020-09-22	50764	265 days
CHE	Europe	Switzerland	2020-09-22	50264	265 days
MDA	Europe	Moldova	2020-09-27	50534	270 days

3. Devolver o total de **novos\_casos/novas\_mortes** por continente/mês, ordenado por continente/mês.

continent	month	new_cases	new_deaths
Africa	1	0	0
Africa	2	3	0
Africa	3	5134	166
Africa	4	31598	1425
Africa	5	105535	2480
Africa	6	251824	5807
...			
South America	6	1336057	44261
South America	7	1841061	58034
South America	8	2206025	59364
South America	9	1735437	48690

4. Devolver para cada dia de um determinado mês e para um dado continente, os países ordenados por ordem decrescente do número de **novos casos positivos**. Por exemplo, para o mês de setembro e para o continente Europa:

```
Dia 1 --> Spain (8115)
          Russia (4993)
          France (3082)
          ...
```

```
Dia 2 --> Spain (8581)
          France (4982)
          Russia (4729)
          ...
          ...
Dia 29 --> Russia (8135)
          France (4070)
          United Kingdom (4044)
```

5. Devolver numa estrutura adequada, todos os países com mais de 70% de fumadores, ordenados por ordem decrescente do número de **novas mortes**. Por exemplo:

```
[[Russia, 81.7, 20385], [Chile,75.7,12698],... ]
```

**Regras**

- A avaliação do trabalho será feita principalmente em função das classes propostas, nomeadamente em termos da sua conformidade com o Paradigma da Programação por Objetos e **eficiência** das estruturas de dados usadas e funcionalidades solicitadas.
- O trabalho deverá ser realizado em **grupos de dois alunos**. Os grupos têm de ser formados e enviados por *email* ao docente das aulas PL, até ao final da **1ª semana aulas**.
- O projeto tem de ser desenvolvido em Java e todas as funcionalidades testadas através de testes unitários e usando o ficheiro de dados disponibilizado.
- É obrigatório o uso da ferramenta de **controle de versões Bitbucket**.
- O relatório deverá servir de ferramenta de avaliação posterior à apresentação. Nele devem apresentar as classes definidas e justificação das mesmas em termos de eficiência de acordo com as funcionalidades implementadas, melhoramentos possíveis.
- O trabalho deve ser submetido no Moodle até às **24 horas do dia 1 de novembro**. A partir desta data a nota do trabalho será penalizada **10% por cada dia de atraso** e não se aceitam trabalhos **após dois dias** da data indicada.
- Na semana seguinte à data de entrega o professor das aulas práticas fará com cada grupo de trabalho uma avaliação do projeto submetido.