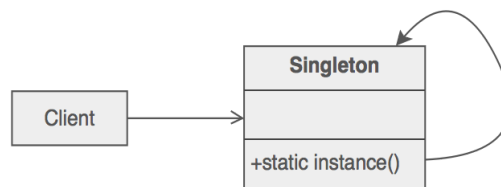


## O Padrão de Desenho Singleton

Os padrões de desenho são uma **abstração** de uma **solução recorrente** para um **problema de desenho** específico. O conceito foi introduzido por **Christopher Alexander em 1979**.

Existem **3 tipos de padrões**: de **criação** (facilitam a criação de objetos), **estruturais** (ajudam a compor classes e objetos) e **comportamentais** (introduzem formas de os objetos interagir e distribuir responsabilidade).

O **Singleton** é um **padrão de criação**, uma vez que condiciona a forma como um objeto é criado. Este padrão obriga à criação de uma **única instância de um objeto** na aplicação, quando ele for **necessário pela primeira vez** (**lazy instantiation**).



A classe Singleton é responsável pela criação do objeto (static), enquanto que os **clientes dessa classe chamam um método assessor para aceder à instância do objecto**. Desta forma, nenhum objeto adquire a posse do objeto Singleton.

### Exercício

Muitas vezes as aplicações necessitam de uma estrutura global que permita o registo de variáveis transversais a toda a aplicação. Essas variáveis têm de ser registadas num objeto único, cuja criação deve ser feita de uma forma centralizada e controlada, de forma a evitar a criação de várias instâncias dessa classe.

Pretende-se neste exercício aplicar o padrão **Singleton** a uma estrutura global da aplicação responsável por fazer o registo da *path* para a pasta onde a aplicação vai guardar os ficheiros e a *connection string* para a base de dados.

#### Specification

Documentação [JavaDocs](#)

#### Deadline

2020-02-29 23:55

[Submit for Testing](#)[Submit for Evaluation](#)