

## Anotações importantes!

- **Padrões de Design:**
  - **Singleton** (DatabaseQueryManager): evita instanciamentos desnecessários da conexão ao banco relacional. O padrão Singleton para acesso a dados não precisa ser usado em caso de utilização do web framework Django, uma vez que o ORM provido pelo framework já faz o controle de instâncias ideal, além de oferecer outras funcionalidades relacionadas a desempenho e segurança. A utilização desse padrão pode ser mais útil caso a opção seja utilizar um banco não relacional, como o MongoDB. No entanto, para essa aplicação, o Singleton será utilizado como centralizador de consultas e escritas no banco de dados.
  - **TemplateMethod** (ModeloBase): **método template é executar\_modelo()**, de forma que as classes concretas implementam as etapas da execução de um modelo, que pode ser específica dependendo do tipo de previsão (por imagem, por série temporal, etc.). Os métodos **pre\_preprocessar\_dados()** e **formatar\_previsao()** são **ganchos** e devem ser utilizados em caso de necessidades específicas de ajustes nas entradas e saídas do processo de previsão. Implementa uma classe abstrata para execução de modelos de estimativa em tempo presente e previsão em tempo futuro, utilizando o padrão TemplateMethod. Modelos de estimativa ou previsão especializam a classe ModeloBase, retornando sempre um objeto da classe Estimativa, indicando a quantidade de pessoas em um determinado horário (no caso de estimativas, o horário fica definido como o horário da execução do modelo). O método template executa as seguintes etapas:
    - carregar\_dados (obtem dados de entrada)
    - preprocessar\_dados (prepara dados para processamento)
    - calcular\_estimativa (executa lógica do modelo predictor ou estimador)
    - gerar\_saida (retornar ou salvar resultados)
  - \*Any em python permite a definição de qualquer tipo de dado
- **Observer** (Local e Usuario): permite que usuários sejam notificados sobre mudanças no status de um local que eles possam ter salvado

*Misc.*

- Incluir lógica de estimadores (via Imagem - nao faz sentido prever a partir de uma imagem presente, apenas estimar) X lógica de previsao (via historico e outros dados - usar modelos preditivos com apoio de informacoes suplementares como previsao meteorologica, calendario de eventos da riotur/prefeitura, utilização de transporte publico, etc.

