

Trabalho Final 2019/1

Acidentes de Trânsito em Porto Alegre

1 Descrição Geral do Trabalho

Este trabalho, que deverá ser feito **individualmente** ou **em grupos de até três alunos**, tem como objetivo modelar e implementar um sistema de consultas sobre os acidentes de trânsito de Porto Alegre utilizando os conceitos estudados na disciplina de Programação Orientada a Objetos.

Para isso, inicialmente deve ser feita a leitura de um arquivo contendo informações sobre os acidentes, para posteriormente serem implementadas diferentes consultas sobre estes acidentes. Leia **atentamente** o enunciado do problema, identifique os elementos e estude as classes que deverão ser criadas para a implementação das consultas.

Antes de mais nada, faça o *download* deste arquivo contendo um projeto com um exemplo inicial de visualização, e os dados de acidentes no formato CSV.

2 Modelagem das classes, armazenamento e leitura do arquivo

A primeira tarefa para a realização desse trabalho é criar uma estrutura de classes capaz de manipular as informações dos acidentes que forem lidas do arquivo. As estruturas de dados necessárias devem ser definidas pelo grupo considerando os conceitos de POO vistos em aula e considerando as consultas solicitadas pelo trabalho. As informações dos acidentes de 2010 a 2016 estão em um arquivo CSV e que possui a seguinte estrutura:

Coluna	Descrição
LOG1 e LOG2	Rua(s)/Avenida(s) (ambas, se cruzamento)
LOCAL	Logradouro ou Cruzamento
PREDIAL1	Número na rua (se não for cruzamento)
TIPO_ACID	Colisão, Abalroamento, Atropelamento, ...
DATA_HORA	Data/hora

DIA_SEM	Dia da semana
FERIDOS, MORTES, ... FATAIS	Estatísticas
AUTO, TAXI, LOTACAO, ...OUTRO	Qtd. de veículos envolvidos
TEMPO	Clima
NOITE_DIA	NOITE ou DIA
FONTE	Origem dos dados
DIA, MES e ANO	Data
LATITUDE e LONGITUDE	Localização

Este arquivo foi criado a partir das informações disponíveis no *DataPOA* (<http://www.datapoa.com.br/dataset/acidentes-de-transito>). Foram coletados os dados de 2010 a 2016, nem todos os dados serão utilização neste trabalho e podem ser descartados na leitura.

A análise estes dados para depois executar a **primeira tarefa**, que consiste na modelagem de uma ou mais classes para representá-los, note que se o acidente ocorrer num cruzamento entre ruas, o acidente deverá aparecer nas estatísticas das duas ruas envolvidas.

A **segunda tarefa** consiste na leitura do arquivo em si. Conforme visto em aula, faça a leitura do arquivo texto no formato CSV e faça o correto armazenamento das informações lidas nas estruturas de dados que pretendes utilizar para realizar as consultas solicitadas.

É importante atentar para o fato de que os nomes das ruas/avenidas possuem “Av.” ou “Rua” na frente. Para facilitar as consultas, sugere-se que esta informação seja guardada de forma separada (por exemplo, um atributo guarda se é rua ou avenida, e o outro atributo guarda apenas o nome). Quando os dados forem ordenados deve-se considerar **apenas** o atributo que guarda o nome.

3 Consultas sobre a estrutura

A partir dos dados lidos o sistema deve ser capaz de responder as seguintes perguntas:

1. Verificar a hora do dia quando ocorrem mais acidentes (dados intervalos de horas cheias: 13:00 até 13:59, 14:00h até 14:59h, etc). Mostre no mapa os locais dos acidentes que ocorrem neste horário;
2. Informar um nome de rua/avenida e um dia da semana e visualizar todos os acidentes que ocorreram na rua e no dia indicados;
3. Identificar para um determinado período (dia inicial e dia final, informado/selecionado pelo usuário), quais avenidas ou ruas que exibiram acidentes de um ou mais tipos, escolhidos pelo usuário (ex: colisão, atropelamento, queda...). Mostrar no mapa os locais;
4. A partir de uma determinada localização mostrar todos os acidentes que ocorreram a até uma determinada distância deste local (valor informado pelo usuário);
5. Identificar a data específica em que houve mais acidentes;
6. Verificar o dia da semana em que houve mais acidentes;
7. Verificar e informar se ocorrem mais acidentes de dia ou à noite;
8. Verificar se ocorrem mais acidentes com tempo chuvoso, nublado ou bom.

As consultas 1, 2 e 3, 4 são **obrigatórias**. Outras duas consultas devem ser implementadas, sendo possível escolher entre as consultas 4, 5, 6, 8.

4 GUI

Deve ser elaborada uma interface com as seguintes funcionalidades:

- Permitir a leitura do arquivo de entrada (deve ser possível ler qualquer arquivo com o formato especificado - após a leitura, os dados em memória devem ser substituídos pelos novos);
- Permitir a escolha de uma rua específica, e visualizar os acidentes encontrados;
- Permitir a escolha de uma data específica e visualizar os acidentes encontrados;
- Permitir a realização e exibição dos resultados das consultas da seção 3.

Para tornar mais interessante a visualização dos acidentes, utilizaremos uma biblioteca capaz de exibir um mapa de ruas, denominada *JXMapView*. Essa biblioteca acessa automaticamente servidores de mapas e exibe as imagens correspondentes. Também é possível acrescentar pontos de referência ao mapa exibido, que é o que utilizaremos para visualizar a localização de cada acidente.

Para um exemplo comentado de uso da biblioteca, veja o código disponível no arquivo *zip*, juntamente com os dados fornecidos.

5 Critérios de Avaliação

Leia **com atenção** os critérios de avaliação e siga corretamente as instruções para entrega e apresentação:

- Leitura de arquivos com a atualização dos dados: 1 ponto
- Modelagem das classes usando interfaces, sobrecarga, sobrescrita: 1 ponto
- Uso adequado das estruturas vistas em aula: 1 pontos
- Interface com o usuário: 2 pontos
- Consultas sobre os dados: 5 pontos
 - Consultas obrigatórias valem 1 ponto cada
 - Consultas extras valem 0,5 ponto cada (se forem executadas todas o trabalho tem um ponto extra)

5.1 Entrega e Apresentação

- Os trabalhos são **individuais** ou **em grupos de até três alunos**.
- O diretório contendo o **projeto completo** do *Intellij*, *Eclipse* ou *NetBeans* deve ser compactado **no formato .zip** e submetido pelo *MOODLE* até a data especificada no plano da disciplina para entrega e apresentação dos trabalho. O trabalho deve ser postado até **ANTES** do horário de início da aula.

- O arquivo compactado deve ter o(s) nome(s) e sobrenome(s) do(s) aluno(s), da seguinte forma: **nome_ultimosobrenome.zip**. Se for em duplas ou grupos de três, usar **nome1_ultimosobrenome1_nome2_ultimosobrenome2_....zip**. Os nomes de todos os componentes do grupo devem estar nos comentários iniciais de todas as classes alteradas pelo grupo.
- Trabalhos entregues, mas não apresentados, terão sua nota **anulada ou reduzida** pelo professor. Durante a apresentação será avaliado o domínio da resolução do problema, podendo inclusive ser possível **invalidar o trabalho** quando constatada a falta de conhecimento sobre o código implementado.
- **A cópia parcial ou completa do trabalho terá como consequência a atribuição de nota ZERO ao trabalho dos alunos envolvidos.**