**Atividade ETL**

Prof. Eduardo

Aluno: Gabriel Boos Duarte

RA: 03231030

Disciplina: Cálculo Computacional

**1• Passo:**

Capturar 200 dados com o python e salvar no banco de dados a cada 2 segundos

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

**2• Passo:**

Exportar o select do componente desejado como .csv para ter o primeiro raw

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente



(Arquivo salvo no diretório)

**3• Passo:**

Salvar os dados do Open Hardware Monitor como .csv e capturar a cada 2 segundos para obter o segundo raw, ao mesmo tempo que salva os dados do python.

Interface gráfica do usuário, Tabela

Descrição gerada automaticamente



(Arquivo salvo no diretório)

**4• Passo:**

Colocar os dados brutos da captura do python (raw) em um excel e formatar transformando em trusted

Tabela

Descrição gerada automaticamente

Dados (Python)

**5• Passo:**

Colocar os dados brutos da captura do Open Hardware Monitor (raw) em um excel e formatar a porcentagem da cpu total transformando em trusted

Tabela

Descrição gerada automaticamente

**6• Passo:**

Colocar o trusted do Open Hardware Monitor no R e fazer o summary() para comparação de dados (foi observado a média de 21.9% de utilização de CPU entre os 200 dados capturados)

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

**7• Passo:**

Colocar o trusted de captura do python no R e fazer o summary() para comparação de dados (foi observado a média de 24% de utilização de CPU entre os 200 dados capturados, muito perto de 21% capturado do Open Hardware)

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

**8• Passo:**

Após ver no sumary() do R e comparar os dados entre os dois Trusteds tratados, é possível ver a semelhança entre os dados capturados, remetendo a uma captura de dados satisfatória, tento em vista que os dados não tem uma variância e dispersão grande entre si.

Abaixo foi feito dois gráficos das duas captura de dados com Python por onde pode notar-se a semelhança da utilização da CPU.

Para plotar esses dois gráficos no Python, foi necessário utilizar a biblioteca pandas para carregar o arquivo .csv no Python, colocando os dados em array, e assim gerando os gráficos com outras bibliotecas (symply, numpym, matplotlib, etc).

Interface gráfica do usuário, Gráfico, Gráfico de dispersão

Descrição gerada automaticamente

**Perguntas Da Atividade:**

**1) Como vc poderá comparar 200 leituras e observar por meio de gráficos se a variação da temperatura apresenta alguma influência no desempenho na máquina ( CPU, RAM e Disco)?**

Você pode ver se esses gráficos são parecidos, ou possuem alguma proporção notória, assim pode fazer uma análise de dados em cima disso.

**2) Como vc irá demonstrar se há ou não influência?? Responda as perguntas a seguir com exemplos:**

Com a análise de dados citada na pergunta acima pode-se ver se um componente tem influencia no outro, ou se os dados em que estão sendo pegos, estão corretos. Ex: primeiro é feito o Raw (dado bruto da captura), Trusted(formatação dos dados) para depois fazer a análise em cima disso. Dessa forma pode-se ter uma visualização primordial do seu estudo em cima da captura de dados. Nas fotos acima foi utilizado a linguagem Python para captura de dados, com a biblioteca Psutil, formatação no Excel e exibição descritiva em R, para depois plotar o gráfico em Python, o mesmo processo foi feito com o Open Hardware Monitor, a diferença é que ele captura os dados direto e já passa para a fase de formatação dos dados.

**3) O que um arquivo csv?**

Um arquivo CSV (Comma-Separated Values) é um formato de arquivo amplamente usado para armazenar e trocar dados tabulares de forma simples e legível por máquina. Ex: transportar uma tabela do banco de dados para outras linguagens de programação, a exemplo de Excel, Python, R.

**4) Como podemos transformar um arquivo csv em txt e em json, e qual a diferença, mostre com exemplos.**

Você pode transformar um arquivo CSV em um arquivo de texto (TXT) e em um arquivo JSON usando várias ferramentas e linguagens de programação, mas existem outras formas simples como sites de automação que fazem essa troca automaticamente. A diferença desses arquivos é que o csv possui dados tabulares (dados relacionais), txt apenas em texto e json por meio de objetos (dados não relacionais)

**5) O que é uma ETL e e quais etapas você aplicou o E, T e o L?. Apresente exemplos também.**

ETL é uma sigla que representa três etapas-chave no processamento e integração de dados em um sistema de informações. As letras E, T e L representam:

Extração (Extract):

A etapa de extração envolve a coleta de dados brutos de diferentes fontes de dados. Isso pode incluir bancos de dados, sistemas legados, arquivos CSV, APIs da web, sensores IoT e muitos outros.

O objetivo da extração é reunir os dados de várias fontes em um local centralizado para processamento adicional.

Exemplo de extração na atividade realizada: Captura dos dados em Python e com Open Hardware Monitor, para futura comparação

Transformação (Transform):

A etapa de transformação envolve a limpeza, padronização e modificação dos dados coletados para torná-los adequados para análise ou carregamento em um sistema de destino.

As transformações podem incluir a remoção de duplicatas, a conversão de tipos de dados, a agregação de informações e a criação de novas variáveis com base nos dados existentes.

Exemplo de transformação: Conversão de datas em um formato consistente, tirar % para pegar apenas o dado da utilização da CPU

Carregamento (Load):

A etapa de carregamento envolve a inserção dos dados transformados em um repositório de destino, um banco de dados relacional ou um sistema de análise.

Exemplo de carregamento: Análise dos dados e comparação entre o trusted do Python e do Open Hardware Monitor, verificando se esses dados estão corretos, e se os componentes possuem alguma correlação, afim de se obter informação acerca desses dados.