**Análise e Desenvolvimento de Sistemas**

**Projeto Integrado II**

**Rafael Nikolas Puggione Rodrigues Querino**

**Desenvolvendo Projeto Integrado II**

**Tutor:** Tiago Barroso dos Santos

**Unidade:** Antônio Carlos

**Belo Horizonte**

**Novembro de 2021**

**Rafael NIkolas Puggione Rodrigues Querino**

**Desenvolvendo Projeto Integrado II**

Trabalho desenvolvido e apresentado à disciplina Projeto Integrado II como exigência para a avaliação em Análise e Desenvolvimento de Sistemas.

Tutor: Tiago Barroso dos Santos

**Belo Horizonte**

**Novembro de 2021**

**Sumário**

**Introdução**………………………………………………………………………………4

**Tarefa 1**………………………………………………………………………………….5

**Tarefa 2**………………………………………………………………………………….6

**Tarefa 3**………………………………………………………………………………….7

**Tarefa 4**………………………………………………………………………………….9

**Referencias**………………………………………………………………………….....10

**Introdução**

O presente trabalho visa abordar a resolução das tarefas apresentadas.

Para a resolução do trabalho, utilizei o conteúdo das disciplinas do semestre tanto quanto de pesquisa em sites bibliográficos.

**Tarefa 1 -** Explique de forma sucinta os conceitos de Lógica Indutiva e Lógica Dedutiva, citando exemplos de cada uma delas.

Lógica Dedutiva: É um tipo de raciocínio que parte do geral para um particular. Onde se todas as informações são verdadeiras, a conclusão deve ser verdadeira. Toda informação da conclusão já estava implícita nas premissas.

* Exemplo: “Todos os mineiros gostam de pão de queijo. Rafael é mineiro. Então Rafael gosta de pão de queijo uai!”.

Lógica Indutiva: É um tipo de raciocínio que parte de um particular para o geral. Onde se todas as premissas são verdadeiras, a conclusão é provavelmente verdadeira. A conclusão tem informações que não estavam na premissa.

* Exemplo: “O Atlético Mineiro tem ganhado mais jogos do que o Cruzeiro recentemente. Logo, o Atlético ganhará do Cruzeiro.

**Tarefa 2 -** Realizar uma breve pesquisa sobre as ferramentas CASE utilizadas para modelagem de dados mais comuns no mercado, elencando suas principais características e diferenças entre elas. Cite ao menos duas ferramentas.

## MySQL Workbench - Desenvolvida exclusivamente para se trabalhar com o SGBD MySQL, é uma das ferramentas gratuitas mais conhecidas e mais utilizadas para quem trabalha com este SGBD, tanto para modelagem ER, como para execução de scripts SQL. Sua principal característica é a facilidade de se trabalhar com o SGBD MySql, onde seu foco está na modelagem física.

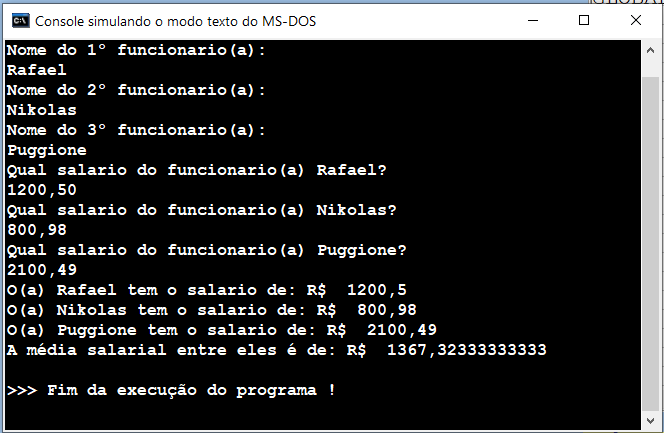
Oracle Designer - O Oracle SQL Developer, uma interface de usuário gráfica gratuita, permite que usuários e administradores de banco de dados executem suas tarefas de banco de dados com menos cliques e pressionamentos de tecla. Uma ferramenta de produtividade, o principal objetivo do SQL Developer é ajudar o usuário final a economizar tempo e maximizar o retorno do investimento na pilha de tecnologia do Oracle Database. Além da base de dados, a Oracle desenvolve uma suíte de desenvolvimento chamada de [Oracle Developer Suite](https://pt.wikipedia.org/wiki/Oracle_Developer_Suite), utilizada na construção de programas de computador que interagem com a sua base de dados. A Oracle também criou a [linguagem de programação](https://pt.wikipedia.org/wiki/Linguagem_de_programa%C3%A7%C3%A3o) [PL/SQL](https://pt.wikipedia.org/wiki/PL/SQL), utilizada no processamento de transações.

Draw.io - Trata-se de uma ferramenta web de fácil acesso e utilizada para o desenvolvimento de diversos diagramas (incluindo, obviamente, o diagrama ER). Talvez não seja uma ferramenta tão conhecida quanto as outras apresentadas, mas ela se destaca justamente pela sua facilidade de uso e acesso, onde não há a necessidade de realizar downloads, podendo ser utilizada de qualquer ambiente, o único requisito para acessá-la é ter internet.

**Tarefa 3** - O pseudocódigo é uma maneira genérica de escrever um algoritmo, utilizando uma linguagem simples sem a necessidade de conhecer a sintaxe de uma linguagem de programação. É importante estar atento para algumas regras básicas quando utilizar pseudocódigos:

* Escolher um nome.
* Avaliar as variáveis, dar atenção aos seus tipos e características.
* Descrever de forma clara o que será armazenado e se as variáveis destinadas a essa informação estão corretas.
* Verificar se as instruções fazem sentido e se as mesmas têm uma sequência lógica.
* Avaliar o resultado e quando pertinente, mostre-o na tela.
* Finalizar o algoritmo.

Sabendo disso, escreva um pseudocódigo que calcula a média. Para este problema entre com três valores. No seu pseudocódigo, apresente o valor final.



Código feito no VISUALG

1. Algoritmo "Media salarial de funcionarios"
2. Var
3. nome: vetor[1..3] de caractere
4. salario: vetor[1..3] de real
5. indicador: inteiro
6. media: real
7. Inicio
8. para indicador de 1 ate 3 faca
9. escreval("Nome do", indicador, "º funcionario(a): ")
10. leia(nome[indicador])
11. fimpara
12. para indicador de 1 ate 3 faca
13. escreval("Qual salario do funcionario(a) ", nome[indicador], "?")
14. leia(salario[indicador])
15. fimpara
16. para indicador de 1 ate 3 faca
17. escreval("O(a) ", nome[indicador]," tem o salario de: R$ ", salario[indicador])
18. media <- media + salario[indicador]
19. fimpara
20. escreval("A média salarial entre eles é de: R$ ", media/3)
22. Fimalgoritmo

**Tarefa 4 -** A especificação de requisitos de software é uma etapa muito importante do desenvolvimento, pois ela pode ser determinante para o sucesso de um software. Nesta fase, é definido os objetivos e funções que um software precisa executar, bem como as que ele não pode ter (restrições). Sabendo disso, defina o que são requisitos funcionais e requisitos não-funcionais, dê exemplos para cada um deles.

Requisitos Funcionais: São as funcionalidades que o software deve ter para atender as necessidades da empresa e dos usuários. Aqui também se enquadram a maneira de como o software responderá a determinadas ações executadas dentro do programa.

Exemplos:

* Divisões de usuários que lhes oferecem diferentes permissões;
* Possibilidade de inclusão ou inclusão de informações;
* Opção de compartilhamento e trocas de informações entre os usuários;
* Consulta e alterações de dados pessoais de clientes;
* Geração de relatório de determinado período de vendas.

Requisitos Não Funcionais: São oque precisam ser determinados para o bom desempenho do sistema, os requisitos não-funcionais são intrínsecos ou implícitos. Eles fazem parte da característica central do software e já é esperado que existam, antes de o cliente pedir.

Exemplos:

* Armazenamento;
* Acessibilidade;
* Existência de ambiente de segurança;
* Salvamento de informações;
* Velocidade.

**Referências Bibliográficas:**

[**https://becode.com.br/diagramas-er-ferramentas/**](https://becode.com.br/diagramas-er-ferramentas/)

[**https://www.oracle.com/br/tools/technologies/whatis-sql-developer.html**](https://www.oracle.com/br/tools/technologies/whatis-sql-developer.html)

[**https://www.metodologiacientifica.org/metodos-de-abordagem/metodo-dedutivo-e-metodo-indutivo/**](https://www.metodologiacientifica.org/metodos-de-abordagem/metodo-dedutivo-e-metodo-indutivo/)

[**https://codificar.com.br/requisitos-funcionais-nao-funcionais/**](https://codificar.com.br/requisitos-funcionais-nao-funcionais/)

[**https://www.monitoratec.com.br/blog/especificacao-de-requisitos-de-software/**](https://www.monitoratec.com.br/blog/especificacao-de-requisitos-de-software/)