

Lista:

1. Vamos retomar o programa que fizemos antes, no qual tínhamos que calcular o aumento nos salários dos funcionários. Você deverá fazer um programa que calcule os valores que serão gastos pela empresa com os aumentos/abonos. Para os exercícios em questão utilizar dicionários e salvar e recuperar os arquivos JSON.
  - a. Cada funcionário receberá o equivalente a 20% do seu salário bruto de dezembro; a. Valor mínimo será de 100 reais; O programa deverá solicitar ao usuário a digitação do salário e do ID do funcionário, com o valor de salário igual a 0 (zero) encerra-se a digitação. Após a entrada de todos os dados o programa deverá calcular o valor do abono concedido a cada colaborador, de acordo com a regra definida acima. Ao final, o programa deverá apresentar:
    - b. O ID, salário, novo salário e diferença do abono de cada funcionário;
    - c. O número total de funcionário processados;
    - d. O valor total a ser gasto com o pagamento do abono;
    - e. O número de funcionário que receberá o valor mínimo de 100 reais;
    - f. O maior valor pago como abono.
    - g. O programa deverá salvar todas as informações em um arquivo JSON;
2. Você precisa desenvolver um programa para uma concessionária de veículos, onde se precisa catalogar, os problemas nos veículos dos clientes. A primeira tarefa dele é fazer o levantamento e entrada dos veículos no pátio, cerca de 200 que se encontram lá, anotando data e hora da entrada, nome do cliente, modelo, ano e descrição do problema e selecionar uma categoria de problema como: motor, câmbio, pneus, freios, sistema de suspensão, sistema de arrefecimento, funilaria, elétrico, tapeçaria e apenas revisão. O sistema deverá ter uma função de emitir um relatório com as quantidades de veículos em cada categoria de problema. O sistema também permitirá a saída do veículo, na qual será guardada a descrição dos serviços e peças trocadas no veículo. Utilize o conceito de dicionário e tuplas e salvar o arquivo em formato JSON.
3. Elabore um programa que permita ao usuário gerenciar uma pequena lista de contatos prioritários (capacidade máxima de 20 contatos). Cada contato deve possuir nome usual (o primeiro nome ou apelido da pessoa), o nome completo, telefone fixo e telefone celular. Construa um pequeno menu de opções, permitindo a leitura da opção desejada até o usuário escolher 0, para sair. O menu terá as seguintes opções:
  - a. 0 sair

- b. 1 incluir um novo contato (adicione o novo contato na primeira posição livre vetor de registros, cuidando com o limite de 20 contatos);
  - c. 2 consultar um contato a partir do nome usual (ler o nome usual, procurar pelo contato, em encontrado, mostrar as informações detalhadas dos campos, se não encontrar, mostrar a mensagem de contato inexistente);
  - d. 3 exibir a listagem de todos os contatos em ordem alfabética de nome usuário (ou seja, antes de imprimir os contatos existente na lista, ordene o vetor em ordem alfabética ascendente de nome usual;
  - e. Salve e recupere as informações em arquivo no formato JSON e permita a recuperação deste arquivo.
4. Vamos retornar aos problemas de notas que fizemos anteriormente, faça um programa que lê duas notas de vários alunos e as armazena em um dicionário, onde a chave é o nome do aluno. A entrada de dados deve terminar quando for lida uma string vazia como nome. Escreva uma função que retorna a média do aluno, dado seu nome, escreva uma função que retorne o nome e a média do aluno de maior média. Para todas as funções é obrigatório o uso de dicionários e retornos de função com tuplas. Salve as informações em arquivo JSON.

**5. Faça um programa de votação onde:**

- H para votar em huguinho;
- Z para votar em zezinho;
- L para votar em luizinho;
- N para votar nulo;
- S para terminar a votação;

**ao término da votação você deverá exibir:**

- soma dos votos para cada candidato;
- informar quem é o vencedor;
- quantos votos inválidos;
- quantos votos nulos;

**Alteração de funcionalidades:**

Altere o programa da votação, para que sejam salvos em vetores, os votos com seu respectivo ID e data, horário com horas, minutos, segundo e milissegundo do voto (por exemplo um incremento de contador) de modo a existir uma auditoria nos votos, com

uma opção no menu com um relatório auditável de votação, cada conjunto de votos nos candidatos deverá exibir o ID de votos ocorridos e salvar em um arquivo JSON.

6. Faça um programa que leia o arquivo JSON dos votos e gere um relatório em tela com as somas, e resumos de cada candidato.