***ATIVIDADES OBJETIVAS***

***Jogos Digitais – Universidade Estácio de Sá (UNESA)***

***Disciplina: RAD Python***

***Discente: Breno Freitas Aguirres***

**Questão 1**

Na imagem a seguir temos o script exercicio\_1\_modulo\_1.py e o arquivo de texto exercicio\_1.txt.

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

Identifique, nas alternativas a seguir, qual o conteúdo do arquivo exercicio\_1.txt após executar o script. Desconsidere eventuais linhas em branco:

**Parabéns! A alternativa b está correta.**

Sempre que utilizamos o modo de acesso w para abrir um arquivo, o conteúdo do arquivo é apagado (truncado). Na linha 11, realizamos a abertura do arquivo utilizando esse modo, excluindo todo conteúdo existente. Nas linhas 12 e 13, escrevemos o conteúdo final que constará no arquivo: Teste 7 e Teste 8.

**Questão 2**

Imagine que fomos contratados para desenvolver um programa em que uma das funcionalidades é extrair a primeira linha de um arquivo. Durante os testes, verificamos que a função que desenvolvemos para extrair as linhas, extrair\_linha, precisava ser refeita, pois, como os arquivos eram muito grandes, a função consumia muita memória, travando o programa.

Texto

Descrição gerada automaticamente

Dentre as opções de scripts apontadas na imagem, identifique a alternativa que atende ao problema apresentado:

**Parabéns! A alternativa d está correta.**

Quando lidamos com arquivos muito grandes, devemos utilizar o método readline() ou iterar diretamente sobre o arquivo. Esses métodos retornam uma linha por vez do arquivo, sem ler todo seu conteúdo, resolvendo o problema de leitura de arquivos muito grandes.

**Questão 1**

No script a seguir, executamos uma sequência de métodos de manipulação de string sobre o arquivo texto exercício\_1\_modulo\_2.txt, também gravamos o resultado no arquivo resultado1.txt.

Texto

Descrição gerada automaticamente

Arquivo texto exercício\_1\_modulo\_2.txt.

Identifique qual é o resultado correto contido no arquivo resultado1.txt.

**Parabéns! A alternativa b está correta.**

No script, na linha 2, lemos todo o conteúdo do arquivo utilizando o método read e, na sequência, utilizamos o método *split* para quebrar o texto em palavras. Cada palavra é iterada na linha 5 e utilizada para montar a f-string da linha 6. Para cada iteração, o conteúdo da f-string será: “Novo conteúdo” + palavra + “ ”. Cada f-string gerada é escrita no arquivo resultado1.txt.

**Questão 2**

Considere que estamos desenvolvendo um programa e, em determinado ponto, temos uma variável chamada frase com a seguinte string: “Ômega;Opala;Monza”.  
  
Qual dos seguintes métodos devemos utilizar para criar uma lista com as palavras Ômega; Opala e Monza?

**Parabéns! A alternativa c está correta.**

O método *split* é utilizado para quebrar uma string em uma lista de palavras. O parâmetro passado para esse método é utilizado como separador. Ao executar o comando frase.split(“;”), criamos a lista [“Ômega”, “Opala”, “Monza”].

**Questão 1**

Quando realizamos operações com arquivos, é uma boa prática tratar as exceções que podem ocorrer. Qual é a sequência de tratamento de exceções que devemos utilizar para garantir que as exceções que podem ser lançadas ao remover um arquivo utilizando a função os.remove() sejam capturadas e tratadas corretamente?

**Parabéns! A alternativa d está correta.**

A função os.remove() pode lançar as exceções FileNotFoundError, PermissionError e IsADirectoryError. Para conseguir tratar essas exceções, isso precisa ser feito antes das exceções mais abrangentes Exception e OSError. A opção D é a única na qual essas três exceções estão antes das mais abrangentes.

**Questão 2**

No script a seguir, utilizamos o método scandir para mostrar os arquivos do diretório explicitado na linha 3.

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

Aplicação do método scandir.

Considere a árvore de diretórios à esquerda da imagem e assinale a resposta que contém a saída do programa.

**Parabéns! A alternativa c está correta.**

A função scandir retorna todo o conteúdo do diretório passado como argumento, incluindo arquivos e subdiretórios. Ela não entra em subdiretórios de forma recursiva.