

LP3 – Exercícios

Aula 3 – Funções Predefinidas da Biblioteca de Classes .NET

Instruções para entrega das listas de exercícios:

Meio de Entrega: As resoluções das listas de exercícios devem ser entregues exclusivamente por meio do ambiente Moodle (<http://eadcampus.spo.ifsp.edu.br>). Usar o mesmo usuário e senha do Sistema Aurora.

Forma de Entrega: Para exercícios com implementação de programas console, devem ser entregues os arquivos das classes (extensão CS). Para exercícios com implementação de programas Windows Forms, devem ser entregues as pastas dos projetos que contêm as aplicações desenvolvidas. Para exercícios com banco de dados, também devem ser entregues as instruções SQL usadas (extensão SQL ou TXT). Todos os arquivos da lista devem ser compactados em um único arquivo (extensão RAR ou ZIP), cujo nome deverá conter a aula, o nome e um sobrenome do aluno. Por exemplo: Aula2_JoaoSilva.zip.

Prazo de Entrega: O prazo de entrega está definido na própria página de exercícios do Moodle, lembrando que o sistema bloqueia o envio de arquivos após a data e horário indicados.

Obs.: O material da disciplina e as listas de exercícios estão disponíveis no ambiente Moodle.

1. Faça um programa em C# que apresente ao usuário as opções a seguir, enquanto ele não digitar a opção 0 (zero). De acordo com o número da opção informado, o programa deverá efetuar a operação, solicitando as informações necessárias ao usuário. Ao final de cada operação, a aplicação deve apresentar o resultado ao usuário.
 - 1) **Arredondar número** – Solicitar ao usuário o valor decimal e a quantidade de casas decimais.
 - 2) **Calcular raiz quadrada** – Solicitar ao usuário um número maior ou igual a zero.
 - 3) **Calcular a potência** – Solicitar ao usuário a base e o expoente.
2. Faça um programa em C# que execute as operações a seguir no texto abaixo. Nas operações que alteram o texto, a aplicação deve reapresentá-lo ao usuário com as modificações realizadas.

Uma ideia traz outra; o boticário imaginou que, uma vez preso o alienista, viriam também buscá-lo a ele, na qualidade de cúmplice. Esta ideia foi o melhor dos vesicatórios. Crispim Soares ergueu-se, disse que estava bom, que ia sair; e apesar de todos os esforços e protestos da consorte vestiu-se e saiu. Os velhos cronistas são unânimes em dizer que a certeza de que o marido ia colocar-se nobremente ao lado do alienista consolou grandemente a esposa do boticário; e notam com muita perspicácia, o imenso poder moral de uma ilusão; porquanto, o boticário caminhou resolutamente ao palácio do governo, não à casa do alienista. Ali chegando, mostrou-se admirado de não ver o barbeiro, a quem ia apresentar os seus protestos de adesão, não o tendo feito desde a véspera por enfermo. E tossia com algum custo. Os altos funcionários que lhe ouviam esta declaração, sabedores da intimidade do boticário com o alienista, compreenderam toda a importância da adesão nova e trataram a Crispim Soares com apurado carinho; afirmaram-lhe que o barbeiro não tardava; Sua Senhoria tinha ido à Casa Verde, a negócio importante, mas não tardava. Deram-lhe cadeira, refrescos, elogios; disseram-lhe que a causa do ilustre Porfírio era a de todos os patriotas; ao que o boticário ia repetindo que sim, que nunca pensara outra coisa, que isso mesmo mandaria declarar a Sua Majestade.

Fonte: O Alienista – Machado de Assis

- a) Substitua as ocorrências de “alienista” por “psiquiatra”.
- b) Retorne se o texto contém as palavras “vila”, “custo” e “receio”.

- c) Retorne quantas vezes a palavra “não” aparece no texto. Sugestão: use o método **indexOf**.
 - d) Retorne quantos caracteres o texto possui.
 - e) Retorne quantos espaços o texto possui. Sugestão: remova todos os espaços do texto, em seguida veja a diferença de caracteres entre o texto original e o texto sem espaços.
3. Faça um programa em C# que leia uma data e, em seguida, apresente ao usuário as opções a seguir, enquanto ele não digitar a opção 0 (zero). De acordo com o número da opção informado, o programa deverá efetuar a operação, solicitando as informações necessárias ao usuário. Ao final de cada operação, a aplicação deve apresentar a nova data ao usuário.
- 1) **Adicionar dias** – Solicitar ao usuário quantos dias devem ser adicionados.
 - 2) **Adicionar meses** – Solicitar ao usuário quantos meses devem ser adicionados.
 - 3) **Adicionar anos** – Solicitar ao usuário quantos anos devem ser adicionados.
 - 4) **Subtrair dias** – Solicitar ao usuário quantos dias devem ser subtraídos.
 - 5) **Subtrair meses** – Solicitar ao usuário quantos meses devem ser subtraídos.
 - 6) **Subtrair anos** – Solicitar ao usuário quantos anos devem ser subtraídos.
 - 7) **Obter dia da semana**
4. Faça um programa em C# que altere os dados abaixo para que fiquem com a formatação indicada:
- 257 -> 00257
 - 2350000 – 2.350.000
 - 5.6 -> 5,600
 - 1.278 -> 127,8%
 - 15/07/2016 -> sexta-feira, 15/julho/2016
5. Faça um programa em C# que leia uma string e apresente ao usuário as opções a seguir, enquanto ele não digitar a opção 0 (zero). De acordo com o número da opção informado, o programa deverá efetuar a operação. Ao final de cada operação, a aplicação deve apresentar o valor convertido ao usuário.
- 1) **Converter para int**
 - 2) **Converter para float**
 - 3) **Converter para double**
 - 4) **Converter para DateTime (data)**
 - 5) **Converter para DateTime (hora)**