



20483 - PROGRAMAÇÃO EM C#

CONTEÚDO PROGRAMÁTICO

Módulo 1: Revisão de sintaxe c #

Este módulo de Clientes uma sintaxe de núcleo e Características da linguagem de Programação C #
He also fornece Uma INTRODUÇÃO PARA O depurador Visual Studio 2012.

Lições

- Visão Geral de escrever aplicativos usando c #
- Tipos de Dados, Operadores e expressões
- Construções de linguagem de Programação C #

Laboratório: Desenvolver o Pedido de INSCRIÇÃO de classe

- Implementar um functionality de edição Pará uma Lista de Alunos
- Implementar um functionality de Inserção Pará uma Lista de Alunos
- Implementar um functionality de Exclusão Pará uma Lista de Alunos
- Exibindo uma Idade do aluno

DEPOIS de concluir este módulo, OS Alunos Serao capazes de:

- Descrever a Arquitetura de aplicativos .NET Framework e USAR OS Recursos de Visual Studio 2012 ec # oferecem parágrafo APOIAR o Desenvolvimento do .NET Framework.
- Use OS Tipos Básicos de Dados, Operadores e expressões fornecidas cabelo c #.
- Use Padrão Construções c # Programação.

Módulo 2: Métodos de Criação, Tratamento de exceções e Aplicações de monitoramento

Este módulo Explica Como CRIAR e Chamar methods, capturar e Manipular exceções . Este módulo descreve TAMBÉM OS Requisitos de Monitorização de Aplicações em larga escala

Lições

- Criando e chamando methods
- Criando methods sobrecarregados e usando Parâmetros OPCIONAIS e saídos
- Tratamento de exceções
- Monitoramento de Aplicações

Laboratório: Estender um aplicativo functionality fazer de Inscrição classe

- Refatorando o código de Inscrição
- Validando como INFORMAÇÕES dos Alunos
- Salvando alterações à Lista de classe

DEPOIS de concluir este módulo, OS Alunos Sero capazes de:

- CRIAR e Métodos invocar, Passar Parâmetros parágrafo methods e Valores de retorno de methods.
- Métodos sobrecarregados, e de CRIAR USAR OPCIONAIS Parâmetros e Parâmetros de SAÍDA.
- Capturar e Manipular exceções e gravar como INFORMAÇÕES nenhum log de eventos.
- Explica o Requisito Para uma Implementação de log, Rastreamento e Criação de PERFIL Durante a Criação de Aplicações em larga escala.

Módulo 3: Desenvolvendo o código parágrafo Uma Aplicação gráfica

Este módulo descreve Como implementar uma Estrutura básica e Elementos Essenciais de Uma típica Aplicação desktop, incluindo o Uso de Estruturas e enumerações, Coleções e eventos

Lições

- Implementação de Estruturas e enumerações
- Organizando Dados em Coleções
- Manipulação de eventos

Laboratório: Escrever o código PARA O aplicativo de Protótipo de notas

- Adicionando lógica de Navegação PARA O aplicativo
- Criando Tipos de Dados Para armazenar INFORMAÇÕES de Usuário e grau
- Exibindo INFORMAÇÕES de Usuário e grau

DEPOIS de concluir este módulo, OS Alunos Sero capazes de:

- Definir e USAR Estruturas e enumerações.

- CRIAR e USAR Coleções simples parágrafo armazenar Dados na Memória.
- Criar, inscrever-se e elevar eventos.

Módulo 4: Criando Classes e implementar Coleções fortemente tipadas

Este módulo Explica Como CRIAR aulas, Definir e implementar as interfaces e CRIAR e USAR Coleções genéricas. Este módulo descreve como also Diferenças entre Tipos de Referência e Tipos de valor em c #.

Lições

- Criando Classes
- Definir e implementar Interfaces
- Implementação de Coleções fortemente tipadas

Laboratório: Adicionando Dados Validação e tipo de Segurança PARA O aplicativo de notas

- Implementando o do professor, aluno e da classe Tipos Como Classes
- Adicionar Validação de Dados Para uma classe de grau
- Exibição de Alunos em Ordem de Nome
- Permitindo Que OS Professores possam MODIFICAR uma classe e da classe de Dados

DEPOIS de concluir este módulo, OS Alunos Serao capazes de:

- CRIAR e USAR como classes personalizadas.
- Definir e implementar as interfaces personalizadas.
- Use OS Genéricos parágrafo implementar Coleções fortemente tipadas.

Módulo 5: Criando Uma Hierarquia de classe usando um Herança

Este módulo Explica Como usar um Herança parágrafo CRIAR Uma Hierarquia de classe e estender Uma classe do .NET Framework . Este módulo descreve also Como CRIAR aulas genéricas e Definir Métodos de Extensão

Lições

- Criando hierarquias de classe
- Estendendo como Classes fazer .NET Framework
- Criando Tipos Genéricos

Laboratório: Refatoração functionality Comum parágrafo uma classe de Usuário

- Criando e herdando a classe base de Usuário

- Aplicação de Complexidade de Senha usando um Método Abstrato
- Criando uma classe de ClassFullException

DEPOIS de concluir este módulo, OS Alunos Serão capazes de:

- Definir as classes Abstratas e herdar de classes de base de Parágrafo CRIAR Uma Hierarquia de classe.
- Herdar as classes de fazer .NET Framework e utilizar Métodos de Extensão Para adicionar functionality personalizada parágrafo uma classe herdada.
- Métodos CRIE e categorias genéricas.

Módulo 6: Leitura e Escrita de Dados LOCAIS

Este módulo Explica Como ler e gravar Dados usando o Arquivo de entrada / SAÍDA (e / s) e Córregos Como serializar e desserializar Dados em Diferentes Formatos

Lições

- Leitura e Escrita de Arquivos
- Serialização e desserialização de Dados
- Executar e / s usando Fluxos

Laboratório: Gerando o Relatório de notas

- Serializar OS Dados PARA O Relatório de notas Como XML
- Visualizando o Relatório de notas
- Persistir OS Dados serializados notas Parágrafo hum Arquivo

DEPOIS de concluir este módulo, OS Alunos Serão capazes de:

- Ler e Escrever Dados de parágrafo e o Sistema de Arquivos usando o Arquivo i / o.
- Converta Dados em formato hum Que Ser gravado PODE OU balneários de hum Arquivo OU Outra fonte de Dados.
- Use Fluxos parágrafo enviar e Receber Dados parágrafo UO de hum Arquivo OU Outra fonte de Dados.

Módulo 7: Acessando hum Banco de Dados

Este módulo Explica Como CRIAR e USAR UM modelo de Dados de Entidade parágrafo hum Acessar Banco de Dados e Como usar LINQ parágrafo Consultar e atualizar Dados

Lições

- Criando e usando modelos de Dados de Entidade

- Consultando Dados usando LINQ
- Atualizando Dados usando LINQ

Laboratório: Recuperar e MODIFICAR Dados de grau

- Criando um modelo de Entidade fazer Banco de Dados Escola de Belas Artes
- ATUALIZAÇÃO fazer aluno e da classe de Dados usando o Entity Framework
- Estendendo o modelo de Entidade parágrafo validar OS Dados

DEPOIS de concluir este módulo, OS Alunos Serão capazes de:

- CRIAR UM modelo de Dados de Entidade, descrever como aulas de chave contidas no modelo e personalizar o código Gerado.
- Use LINQ parágrafo Consultar e trabalhar Há com Dados.
- Use o LINQ parágrafo inserir, atualizar e excluir Dados.

Módulo 8: Acessando Dados Remotos

Este módulo Explica Como usar OS Tipos nenhum espaço para nome System.Net e WCF Data Services, parágrafo Consultar e MODIFICAR Dados Remotos.

Lições

- Acessando Dados em Toda a Web
- Acessando Dados de na Nuvem

Laboratório: Recuperando e modificação da classe de Dados na Nuvem

- Criando um WCF Data Services PARA O Banco de Dados SchoolGrades
- Integrar o Serviço de Dados do WCF fazer aplicativo
- Recuperando o estudante Fotografias Sobre o Web (se o ritmo permitir)

DEPOIS de concluir este módulo, OS Alunos Serão capazes de:

- Use como classes nenhum espaço para nome System.Net parágrafo enviar e Receber Dados através da Web.
- CRIAR e USAR UM Serviço WCF de Dados parágrafo Acessar Dados na Nuvem.

Módulo 9: Projetando uma interface de usuário parágrafo um aplicativo Gráfico

Este módulo Explica Como Construir e Interface do Usuário Uma gráfica do estilo usando XAML . Este módulo descreve also Como exibir Dados em Uma interface de de Usuário usando vinculação de Dados

Lições

- Usando PARA O XAML Projetar Uma interface de Usuário
- Controles de vinculação de Dados
- Denominando Uma interface de Usuário

Laboratório: personalização de o estudante Fotografias e denominando o aplicativo

- Personalização de uma aparência de Fotografias de estudante
- Denominando a vista de Logon
- Animando o Controle StudentPhoto (se o ritmo permitir)

DEPOIS de concluir este módulo, OS Alunos Serão capazes de:

- Definir Modos de exibição XAML e Controles parágrafo Projetar Uma interface de de Usuario gráfica simples.
- Use Técnicas de vinculação de Dados XAML parágrafo Ligar Elementos XAML parágrafo Uma fonte de Dados e exibir Dados.
- Adicione estilo e interface de Dinâmicas de Transformações parágrafo Uma fazer XAML Usuário.

Módulo 10: Melhorando a CAPACIDADE de RESPOSTA E O Desempenho do aplicativo

Este módulo Explica Como melhorar o taxa de transferência e ritmo de RESPOSTA de aplicativos POR Meio de Tarefas e Operações assíncronas

Lições

- Execuções multitarefa usando Tarefas e Expressões Lambda
- Executar Operações de forma assíncrona
- Sincronizando o Acesso simultâneo AOS Dados

Laboratório: Melhorando a CAPACIDADE de RESPOSTA E O Desempenho do aplicativo

- Garantindo that o Usuário interface permanece responsivo when recuperando Dados parágrafo Professores
- Fornecendo feedback visual Durante Operações de longa Duração

DEPOIS de concluir este módulo, OS Alunos Serão capazes de:

- CRIE Tarefas e expressões lambda parágrafo implementar multitarefa.
- Definir e USAR methods assíncronos Para melhorar a CAPACIDADE de fazer RESPOSTA aplicativo.
- Coordene o Acesso simultâneo AOS Dados Compartilhados Entre varias Tarefas usando primitivos síncronos e Coleções simultâneas.

Módulo 11: Integração com código NÃO gerenciado

Este módulo Explica Como Integrar Componentes Dinâmicos e Bibliotecas NÃO Gerenciadas em Um aplicativo c # . Este módulo descreve also Como Controlar o ritmo de vida de Recursos Nao gerenciados

Lições

- Criando e usando Objetos Dinâmicos
- Gerenciar o ciclo de vida de Objetos e Controlar Recursos NÃO gerenciados

Laboratório: Atualizando o Relatório de notas

- Gerando o Relatório de notas usando o Microsoft Office Word
- Controlando o ritmo de vida de Objetos fazer Palavra implementando o Padrão Descarte

DEPOIS de concluir este módulo, OS Alunos Serao capazes de:

- Integre o código NÃO gerenciado em Um aplicativo c # usando o Dynamic Language Runtime.
- Controlar a vida Útil de Recursos NÃO gerenciados e certifique-se de that they São descartados corretamente.

Módulo 12: Criação de Módulos (assembléias) e Tipos reutilizáveis

Este módulo Explica Como examinar OS metadados dos Tipos POR Meio de Reflexão, e USAR CRIAR Atributos Personalizados, Gerar código gerenciado em Tempo de Execução e gerenciar versões Diferentes de Módulos (montagens)

Lições

- Examinar metadados OS fazer Objeto
- Criando e usando Atributos Personalizados
- Geração de código gerenciado
- Controle de Versão, assinar e implantar Assembléias

Laboratório: Especificando um sistema operacional Dados Incluir nenhum Relatório de notas

- Criando o atributo IncludeInReport
- Gerar o Relatório
- Armazenando o Grades.Utilities Assembleia centralmente

DEPOIS de concluir este módulo, OS Alunos Serao capazes de:

- Examine OS metadados de Objetos em Tempo de Execução usando Reflexão.
- CRIAR e USAR uma classe de atributo Personalizado.
- Gere código gerenciado em Tempo de Execução usando o CodeDOM.
- Gerenciar como Diferentes versões de montagem e hum hum implantar montagem sem cache de montagens globais.

Módulo 13: Criptografando e descriptografando Dados

Este módulo Explica Como criptografar e descriptografar usando criptografia simétrica Dados e assimétrica

Lições

- Implementação de criptografia simétrica
- Implementação de criptografia assimétrica

Laboratório: Criptografando e descriptografando Relatórios de notas

- Criptografando o Relatório de notas
- Descriptografando o Relatório de notas

DEPOIS de concluir este módulo, OS Alunos Serão capazes de:

- Executar uma criptografia simétrica usando como classes não namespace System. Segurança.
- Executar uma criptografia assimétrica usando como classes não namespace System. Segurança.