C#

TechGuide - Alura, FIAP e PM3

Level 1

C# - Fundamentals:

- C# is a multi-paradigm, strongly typed programming language developed by Microsoft as part of the .NET platform. The source code is compiled into Common Intermediate Language (CIL) which is interpreted by the Common Language Runtime (CLR) virtual machine. It is designed to run on the Common Language Infrastructure of the .NET Framework platform.
- Knowing the primitive types
- Declaring variables, considering the different types
- Using conditional structures ('if', 'else')
- Knowing the assignment and comparison operators
- Using repetition structures and loops ('while', 'for')
- Using functions, passing parameters and arguments
- Manipulating methods
- Manipulating arrays and lists
- Getting data from an API
- Making asynchronous calls
- Creating constructors

Contents

- `WebSite` Microsoft Docs: C# documentation (https://learn.microsoft.com/en-us/dotnet/csharp/)

- WebSite Microsoft Docs: A tour of the C# language (https://learn.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/)
- WebSite Microsoft Docs: C# types and members (https://learn.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/types)
- WebSite Microsoft Docs: Hello World - Introduction to C# interactive C# tutorial (https://learn.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/tutorials/hello-world)
- WebSite Microsoft Docs: Hello World - Arrays (https://learn.microsoft.com/en-us/dotnet/csharp/programming-guide/arrays/)
- YouTube freeCodeCamp.org: C# Tutorial - Full Course for Beginners (https://www.youtube.com/watch?v=GhQdlIFylQ8)
- YouTube tutorialsEU: Learn C# 10 and .NET 6 in 7 hours (https://www.youtube.com/watch?v=q_F4PyW8GTg)

Object-oriented Programming Concepts:

- Object-oriented programming (OOP) is a programming paradigm based on the concept of 'objects', which can contain data and code: data in the form of fields (often known as attributes or properties), and code, in the form of procedures (often known as methods). A common feature of objects is that procedures (or methods) are attached to them and can access and modify the object's data fields. Some of the main concepts are classes and instances, inheritance, and encapsulation.
- How objects work
- Creating and using constructors
- What classes are
- Creating and using Methods
- How encapsulation works
- What inheritance is
- What polymorphism is
- How interfaces work

- What abstractions are

Contents

- WebSite MDN Web Docs: Object-oriented programming (https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Objects/Object-oriented_programming)
- WebSite Geeks for Geeks: Object Oriented Programming (OOPs) Concept in Java (https://www.geeksforgeeks.org/object-oriented-programming-oops-concept-in-java/)
- WebSite W3Schools: Object-Oriented Programming Style (https://www.w3schools.blog/object-oriented-programming-style)
- Article What is object-oriented programming? OOP explained in depth (https://www.educative.io/blog/object-oriented-programming)
- YouTube Programming with Mosh: Object-oriented Programming in 7 minutes (https://www.youtube.com/watch?v=QJjY2TNyI-8)
- YouTube freeCodeCamp.org: Intro to Object Oriented Programming - Crash Course (https://www.youtube.com/watch?v=SiBw7os-_zI)
- YouTube Traversy Media: JavaScript OOP Crash Course (ES5 & ES6) (https://www.youtube.com/watch?v=vDJpGenyHaA)

C# - Collections:

- A collection represents a group of objects, known as its elements. They are like containers that group multiple items in a single unit. Some collections allow duplicate elements and others do not. Some are ordered and others unordered.
- Learn the difference between Dictionary, List, Queue, SortedList and Stack
- Get to know how to work with ArrayList e HashTable
- Iterators

Contents

- WebSite Microsoft Docs: Collections (C#) (https://learn.microsoft.com/en-us/dotnet/csharp/programming-guide/concepts/collections)

- `Article` C# Collections (Generic Collections and Non-Generic Collections) (https://aemkannika.medium.com/c-collections-c755806d3be6)
- `Article` List in C#: implementation and features (https://medium.com/pvs-studio/list-in-c-implementation-and-features-8ae73271b747)
- `YouTube` Bro Code: C# Lists (https://www.youtube.com/watch?v=vQzREQUhGSA)
- `YouTube` Simplilearn: Collections In C# Explained (https://www.youtube.com/watch?v=SR4-_gWRILU)
- `YouTube` Derek Banas: C# Tutorial 11 Collections (https://www.youtube.com/watch?v=SR4-_gWRILU)

NuGet:

- NuGet is a package manager for the .NET platform. It defines how packages for this platform are created, published and consumed, and provides tools for each of these functions.
- Gerenciar pacotes
- Compartilhar bibliotecas

Contents

- `WebSite` Microsoft Docs: An introduction to NuGet (https://learn.microsoft.com/en-us/nuget/what-is-nuget)
- `Article` Creating and using a local NuGet package repository (https://medium.com/@churi.vibhav/creating-and-using-a-local-nuget-package-repository-9f19475d6af8)
- `Article` Using NuGet to publish .NET packages (https://medium.com/@loginradius/using-nuget-to-publish-net-packages-d42eb746aa62)
- `YouTube` dotnet: What is NuGet? (https://www.youtube.com/watch?v=WW3bO1INDmo)
- `YouTube` IAmTimCorey: Creating NuGet Packages the easy way with .NET Standard in C# (https://www.youtube.com/watch?v=AF1y9gLcxjQ)

C# - System.IO:

- The System.IO namespace consists of IO related classes, structures, delegates and enumerations. These classes can be used to reads and write data to files or data streams. It also contains classes for file and directory support.
- Ler dados de arquivos
- Escrever dados em arquivos
- Gerenciar arquivos com Using

Contents

- `WebSite` Microsoft Docs: System.IO Namespace (https://learn.microsoft.com/en-us/dotnet/api/system.io?view=net-6.0)
- `WebSite` Microsoft Docs: File and Stream I/O (https://learn.microsoft.com/en-us/dotnet/standard/io/)
- `WebSite` JavaTPoint: C# System.IO Namespace (https://www.javatpoint.com/c-sharp-system-io)
- `YouTube` IAmTimCorey: Working With The File System in C# - Managing Folders and Files (https://www.youtube.com/watch?v=9mUuJlKq40M)
- `YouTube` Derek Banas: C# Tutorial - File I/O (https://www.youtube.com/watch?v=HKqMqFJr4SY)

C# - Memory Management:

- Automatic memory management is one of the services that the Common Language Runtime provides during Managed Execution. The Common Language Runtime's garbage collector manages the allocation and release of memory for an application.
- Understand how the memory is managed
- Getting to know the Garbage Collector
- Understanding about the managed Stack and Heap

Contents

- `WebSite` Microsoft Docs: Automatic Memory Management (https://learn.microsoft.com/en-us/dotnet/standard/automatic-memory-management)
- `Article` C# Memory Management - Part 1 (https://medium.com/c-programming/c-memory-management-part-1-c03741c24e4b)
- `Article` .NET Memory Management (https://www.c-sharpcorner.com/UploadFile/26b237/net-memory-management/)
- `YouTube` Programming With Chris: Memory Management in .NET / C# Explained (https://www.youtube.com/watch?v=2kUzyncszeA)
- `YouTube` Ankpro Training: Stack and Heap - What is stack memory? | What is heap memory? (https://www.youtube.com/watch?v=sC1tgnPfa_s)

C# - Testing:

- Software testing is the process of evaluating and verifying that a software product or application does what it is supposed to do. The benefits of testing include preventing bugs, reducing development costs and improving performance.
- Using unit tests
- Using integration testing
- Using behavioral testing
- Using mocks

Contents

- `WebSite` Microsoft Docs: Testing in .NET (https://learn.microsoft.com/en-us/dotnet/core/testing/)
- `WebSite` Microsoft Docs: Walkthrough - Create and run unit tests for managed code (https://learn.microsoft.com/en-us/visualstudio/test/walkthrough-creating-and-running-unit-tests-for-managed-code?view=vs-2022)

- `WebSite` Microsoft Docs: Walkthrough - Test-driven development using Test Explorer (https://learn.microsoft.com/en-us/visualstudio/test/quick-start-test-driven-development-with-test-explorer?view=vs-2022)
- `Article` A Basic Introduction To C# Unit Test For Beginners (https://www.c-sharpcorner.com/article/a-basic-introduction-of-unit-test-for-beginners/)
- `Article` Implementing Unit And Integration Tests On .NET With xUnit (https://www.c-sharpcorner.com/article/implementing-unit-and-integration-tests-on-net-with-xunit/)
- `YouTube` Programming with Mosh: Unit Testing C# Code - Tutorial for Beginners (https://www.youtube.com/watch?v=HYrXogLj7vg)
- `YouTube` tutorialsEU: C# Unit Testing (https://www.youtube.com/watch?v=GSTVFx0KOzI)

ADO.NET:

- ADO.NET is a set of classes that expose data access services for .NET Framework programmers. ADO.NET provides a rich set of components for creating distributed, data-sharing applications. It is an integral part of the .NET Framework, providing access to relational, XML, and application data.
- Handling databases
- Getting to know DataSet and DataTable
- Performing connections
- Handling XML documents

Contents

- `WebSite` Microsoft Docs: ADO.NET Overview (https://learn.microsoft.com/en-us/dotnet/framework/data/adonet/ado-net-overview)
- `YouTube` André Lima: Salvando arquivos no banco de dados com C# e ADO.NET (https://www.youtube.com/watch?v=5E9aUXQCsGA)
- `YouTube` NetSecProf: Using C# ADO.NET for SQL INSERT, DELETE, UPDATE, and SELECT (https://www.youtube.com/watch?v=H0ZcSyl0xws)

Entity Framework Core:

- Entity Framework Core is an object-relational mapper (ORM). Object-relational mapping is a technique that enables developers to work with data in object-oriented way by performing the work required to map between objects defined in an application's programming language and data stored in relational datasources.
- Manipulating databases using .NET objects
- Creating models corresponding to a database
- Performing queries

Contents

- `WebSite` Microsoft Docs: Entity Framework Core (https://learn.microsoft.com/en-us/ef/core/)
- `Article` 10 Best Practices for Entity Framework Core (https://medium.com/@walid.mougharbel.1983/10-best-practices-for-entity-framework-core-d44d340b2046)
- `YouTube` Amichai Mantinband: Entity Framework Core - Ultimate guide for beginners (https://www.youtube.com/watch?v=v19arLqQkP8)
- `YouTube` Patrick God: CRUD with a .NET 6 Web API & Entity Framework Core (https://www.youtube.com/watch?v=Fbf_ua2t6v4)
- `YouTube` IAmTimCorey: Entity Framework Best Practices - Should EFCore Be Your Data Access of Choice? (https://www.youtube.com/watch?v=qkJ9keBmQWo)

Data Structures:

- In the context of computers, the data structure is a specific way of storing and organizing data in the computer's memory so that these data can be easily retrieved and efficiently used when needed later.
- Knowing the main data structures (linked list, stack, queue, tree, etc)
- Implementing the main data structures

Contents

- `WebSite` W3Schools: Data Structure Tutorial (https://www.w3schools.in/data-structures/tutorials/)
- `WebSite` Geeks for Geeks: Data Structures (https://www.geeksforgeeks.org/data-structures/)
- `WebSite` Data Structures Tutorial (https://www.javatpoint.com/data-structure-tutorial)
- `Article` 8 Common Data Structures every Programmer must know (https://towardsdatascience.com/8-common-data-structures-every-programmer-must-know-171acf6a1a42)
- `YouTube` Internet Made Coder: Data Structures Explained for Beginners - How I Wish I was Taught (https://www.youtube.com/watch?v=ZdU4wMyiTSs)
- `YouTube` freeCodeCamp.org: Data Structures Easy to Advanced Course - Full Tutorial from a Google Engineer (https://www.youtube.com/watch?v=ZdU4wMyiTSs)
- `YouTube` Aaron Jack: Data Structures you MUST know (as a Software Developer) (https://www.youtube.com/watch?v=ZdU4wMyiTSs)

Level 2

CLR:

- The Common Language Runtime (CLR) is the virtual machine component of Microsoft .NET Framework that manages the execution of .NET programs.
- Understanding how the CLR works
- Understanding memory management
- Getting to know CIL and JIT

Contents

- `WebSite` Microsoft Docs: Common Language Runtime (CLR) overview (https://learn.microsoft.com/en-us/dotnet/standard/clr)

- `Article` Common Language Runtime(CLR) DotNet (https://medium.com/@mirzafarrukh13/common-language-runtime-dotnet-83e0218edcae)
- `Article` CLR Internals Part I : Memory Management (https://medium.com/@immortals.300/clr-internals-part-i-memory-management-671cf428bf84)
- `YouTube` Easy Technology: Overview of the Common Language Runtime CLR (https://www.youtube.com/watch?v=ttgqRyMPU9c)

LINQ:

- Language-Integrated Query (LINQ) is the name for a set of technologies based on the integration of query capabilities directly into the C# language.
- Criar consultas
- Conhecer as cláusulas Select e Where
- Consultar coleções de objetos em memória
- Mapear o banco de dados com Linq to Sql

Contents

- `WebSite` Microsoft Docs: Language Integrated Query (LINQ) (C#) (https://learn.microsoft.com/en-us/dotnet/csharp/programming-guide/concepts/linq/)
- `WebSite` Microsoft Docs: Introduction to LINQ Queries (C#) (https://learn.microsoft.com/en-us/dotnet/csharp/programming-guide/concepts/linq/introduction-to-linq-queries)
- `Article` LINQ in C# Explained In Five Minutes (https://medium.com/swlh/linq-in-c-explained-in-five-minutes-d596309a57e6)
- `YouTube` dotnet: Introduction to Language Integrated Query (LINQ) (https://www.youtube.com/watch?v=p5myHVOtmiU)
- `YouTube` IAmTimCorey: C# Essentials - Linq for Lists (https://www.youtube.com/watch?v=yClSNQdVD7g)

- YouTube IAmTimCorey: Simplified LINQ Ordering In 10 Minutes or Less (https://www.youtube.com/watch?v=lZNdxWp7R0Y)

C# - Serialization:

- Serialization is the process of converting an object into a stream of bytes to store the object or transmit it to memory, a database, or a file. Its main purpose is to save the state of an object in order to be able to recreate it when needed.
- Sending an object to a remote application using a web service
- Passing an object as a JSON or XML string
- Passing user-specific or security information between applications

Contents

- WebSite Microsoft Docs: Serialization (C#) (https://learn.microsoft.com/en-us/dotnet/csharp/programming-guide/concepts/serialization/)
- WebSite Microsoft Docs: Basic serialization (https://learn.microsoft.com/en-us/dotnet/standard/serialization/basic-serialization)
- Article C# language: Data serialization (https://medium.com/@sawomirkowalski/c-language-data-serialization-2b3783473932)
- Article Conditional serialization using Csharp (https://medium.com/@samueleresca/conditional-serialization-using-csharp-3ba383a5ac36)
- YouTube Derek Banas: C# Serialization (https://www.youtube.com/watch?v=jbwjbbc5PjI)
- YouTube Nick Proud: C# Tutorial - Serializing and Deserializing JSON with NewtonSoft (https://www.youtube.com/watch?v=pJtuuolUhCc)

C# - Networking and Sockets:

- Networking is a concept of connecting two or more computing devices together so that we can share resources. Socket programming provides

facility to share data between different computing devices through a network.

- Opening an interactive communication session between the user's browser and a server
- Sending messages to a server and receive replies without querying the server

Contents

- `WebSite` Microsoft Docs: Sockets in .NET (https://learn.microsoft.com/en-us/dotnet/fundamentals/networking/sockets/sockets-overview)
- `WebSite` Microsoft Docs: Use Sockets to send and receive data over TCP (https://learn.microsoft.com/en-us/dotnet/fundamentals/networking/sockets/socket-services)
- `Article` Socket Programming in C# (https://www.geeksforgeeks.org/socket-programming-in-c-sharp/)
- `Article` Socket Programming in C# (https://www.c-sharpcorner.com/article/socket-programming-in-C-Sharp/)
- `YouTube` Sloan Kelly: Socket Programming in C# (Overview) (https://www.youtube.com/watch?v=vwMLH4Qpupw)
- `YouTube` Brian: C# Socket Programming - Multiple Clients (https://www.youtube.com/watch?v=xgLRe7QV6QI)
- `YouTube` wolfs cry games: TCP Socket Tutorial (https://www.youtube.com/watch?v=g5yEWLJxNmI)

ASP.NET Core:

- ASP.NET Core is an open-source and cross-platform framework for building modern cloud-based applications, such as web apps, IoT apps and mobile backends.
- Creating applications and web services
- Maintaining an MVC application
- Developing a client-side web user interface

- Creating a Web API

Contents

- `WebSite` Microsoft Docs: Overview of ASP.NET Core (https://learn.microsoft.com/en-us/aspnet/core/introduction-to-aspnet-core?view=aspnetcore-6.0)
- `Article` ASP.NET Core - Overview (https://www.tutorialspoint.com/asp.net_core/asp.net_core_overview.htm)
- `YouTube` freeCodeCamp.org: Learn ASP.NET Core MVC (.NET 6) (https://www.youtube.com/watch?v=hZ1DASYd9rk)
- `YouTube` freeCodeCamp.org: ASP.NET Core Crash Course - C# App in One Hour (https://www.youtube.com/watch?v=lwVvrRIS7aI)

Dapper:

- Dapper is an object–relational mapping (ORM) product for the Microsoft .NET platform. It provides a framework for mapping an object-oriented domain model to relational databases.
- Performing database queries such as 'select', 'insert', 'update', 'delete'
- Manipulating databases

Contents

- `Article` Basics of Dapper (https://www.jetbrains.com/dotnet/guide/tutorials/basics/dapper/)
- `Article` Welcome To Learn Dapper (https://www.learndapper.com/)
- `Article` C# | Dapper (https://www.geeksforgeeks.org/c-dapper/)
- `YouTube` Milan Jovanović: Write Faster SQL Queries With Dapper In .NET (https://www.youtube.com/watch?v=neH0_7bti_I)
- `YouTube` Patrick God: CRUD with Dapper in a .NET 6 Web API using SQL Server (https://www.youtube.com/watch?v=n0zkkoL8eNs)
- `YouTube` IAmTimCorey: Simple C# Data Access with Dapper and SQL - Minimal API Project (https://www.youtube.com/watch?v=dwMFg6uxQ0I)

Dependency Injection:

- Dependency Injection is a design pattern in which a class requests dependencies from external sources instead of creating them.
- Avoiding high level of code coupling within an application
- Implementing inversion of control

Contents

- `Article` Dependency Injection (https://www.tutorialsteacher.com/ioc/dependency-injection)
- `Article` A quick intro to Dependency Injection: what it is, and when to use it (https://www.freecodecamp.org/news/a-quick-intro-to-dependency-injection-what-it-is-and-when-to-use-it-7578c84fa88f/)
- `YouTube` Anthony Ferrara: Dependency Injection (https://www.youtube.com/watch?v=IKD2-MAkXyQ)
- `YouTube` Amigoscode: Learn Dependency Injection and Write Better Code (https://www.youtube.com/watch?v=eQ90v7HQT-Q)
- `YouTube` Ryan Schachte: Dependency Injection & Inversion of Control (https://www.youtube.com/watch?v=EPv9-cHEmQw)

C# - Multithreading:

- Multithreading is the ability to perform multiple operations at the same time. Operations with the potential of holding up other operations can execute on separate threads.
- Running multiple tasks simultaneously
- Understanding how threads are executed
- Learning how to make a thread wait at a specific point

Contents

- `WebSite` Microsoft Docs: Using threads and threading (https://learn.microsoft.com/en-us/dotnet/standard/threading/using-threads-and-threading)

- `Article` C# - Multithreading (https://www.tutorialspoint.com/csharp/csharp_multithreading.htm)
- `Article` C# Multithreading (https://www.geeksforgeeks.org/c-sharp-multithreading/)
- `YouTube` Bro Code: C# multithreading (https://www.youtube.com/watch?v=rUbmW4qAh8w)
- `YouTube` kudvenkat: Multithreading in C# (https://www.youtube.com/watch?v=8mjqXiggWNc)
- `YouTube` AngelSix: C# Threads, Tasks, Multi-threading & UI Cross-threading (https://www.youtube.com/watch?v=XXg9g56FS0k)

Level 3

C# - Delegates and Events:

- Delegates are objetcs that are used as function pointers to refer to a method assigned to them.
- Events are the action performed which changes the state of an object. Events are declared using delegates - they provide encapsulation to the delegates.
- Understanding the concept of delegate
- Creating a reference to a function with a certain list of parameters
- Understanding the concept of event
- Handling different types of events

Contents

- `WebSite` Microsoft Docs: Distinguishing Delegates and Events (https://learn.microsoft.com/en-us/dotnet/csharp/distinguish-delegates-events)
- `WebSite` Microsoft Docs: Handle and raise events (https://learn.microsoft.com/en-us/dotnet/standard/events/)

- `Article` Delegate Vs Events (https://www.linkedin.com/pulse/delegate-vs-events-pawan-verma/)
- `Article` Difference between delegates and events in C# (https://www.tutorialsteacher.com/articles/difference-between-delegate-and-event-csharp)
- `YouTube` Programming with Mosh: C# Events and Delegates Made Simple (https://www.youtube.com/watch?v=jQgwEsJlSy0)
- `YouTube` Simplilearn: C# Delegates Explained (https://www.youtube.com/watch?v=vOlLd2SsKM4)
- `YouTube` Ricardo Maroquio: Delegate and Event in C# (https://www.youtube.com/watch?v=xr2C4f5bRZM)

C# - Anonymous methods and lambda expressions:

- Anonymous methods are unnamed methods that can be defined using the delegate keyword.
- Lambda expressions are used like anonymous functions, but you don't need to specify the type of the value that you input, making them more flexible to use.
- Creating anonymous functions that you can use to create delegates
- Creating local functions that can be passed as arguments

Contents

- `WebSite` Microsoft Docs: Lambda expressions and anonymous functions (https://learn.microsoft.com/en-us/dotnet/csharp/language-reference/operators/lambda-expressions)
- `WebSite` Microsoft Docs: Delegates e lambdas (https://learn.microsoft.com/en-us/dotnet/standard/delegates-lambdas)
- `WebSite` Microsoft Docs: Delegates with Named vs. Anonymous Methods (https://learn.microsoft.com/en-us/dotnet/csharp/programming-guide/delegates/delegates-with-named-vs-anonymous-methods)
- `Article` Anonymous Method in C# (https://www.geeksforgeeks.org/anonymous-method-in-c-sharp/)

- `Article` C# - Anonymous Method (https://www.geeksforgeeks.org/anonymous-method-in-c-sharp/)
- `Article` Lambda Expressions in C# (https://www.geeksforgeeks.org/lambda-expressions-in-c-sharp/)
- `YouTube` tutorialsEU: C# Lambda Expressions and Anonymous Functions Tutorial (https://www.youtube.com/watch?v=4gNeU539a4M)
- `YouTube` Tutorials Point: C# - Anonymous Method (https://www.youtube.com/watch?v=vAnXrQTq8F4)
- `YouTube` Tutorials Point: C# - Lambda Expression (https://www.youtube.com/watch?v=kYeKFMf2mO8)

Containers:

- Containers are software packages that contain all the elements needed to run in any environment.
- Kubernetes (also known as k8s or “kube”) is an open source container orchestration platform that automates many of the manual processes involved in deploying, managing, and scaling containerized applications.
- Isolating your software to run independently
- Deploying software in clusters
- Modularizing your system into smaller packages
- Getting to know the Docker platform
- Getting to know Kubernetes

Contents

- `WebSite` IBM: Containers (https://www.ibm.com/en-us/cloud/learn/containers)
- `WebSite` HPE: What are containers? (https://www.hpe.com/us/en/what-is/containers.html)
- `WebSite` Google Cloud: What are Containers? (https://cloud.google.com/learn/what-are-containers)

- WebSite Azure: What is a container? (https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-a-container/)
- WebSite Kubernetes - Documentation (https://kubernetes.io/docs/tutorials/kubernetes-basics/)
- Article Red Hat: What is Kubernetes? (https://www.redhat.com/en/topics/containers/what-is-kubernetes)
- Article Azure: What is Kubernetes? (https://azure.microsoft.com/en-us/topic/what-is-kubernetes/)
- YouTube freeCodeCamp.org: Docker Tutorial for Beginners (https://www.youtube.com/watch?v=eLiHH9Bu04E)
- YouTube freeCodeCamp.org: Docker Containers and Kubernetes Fundamentals – Full Hands-On Course (https://www.youtube.com/watch?v=kTp5xUtcalw)
- YouTube TechWorld with Nana: Kubernetes Tutorial for Beginners [FULL COURSE in 4 Hours] (https://www.youtube.com/watch?v=X48VuDVv0do)

Microservices architecture:

- Microservices are an architectural approach in which software consists of small independent services that communicate with each other and are organized according to their business domains.
- Learning the concept of planned architecture for microservices
- Performing communication using APIs
- Improving the scalability of a system

Contents

- WebSite Google Cloud: Microservice architecture style (https://learn.microsoft.com/en-us/azure/architecture/guide/architecture-styles/microservices)
- WebSite Amazon: What are microservices? (https://aws.amazon.com/en/microservices/)
- Article The What, Why, and How of a Microservices Architecture (https://medium.com/hashmapinc/the-what-why-and-how-of-a-

microservices-architecture-4179579423a9)

- `Article` Understanding Microservices: From Idea To Starting Line (https://medium.com/free-code-camp/microservices-from-idea-to-starting-line-ae5317a6ff02)
- `YouTube` TechWorld with Nana: Microservices explained - the What, Why and How? (https://www.youtube.com/watch?v=rv4LlmLmVWk)
- `YouTube` ByteByteGo: What Are Microservices Really All About? (And When Not To Use It) (https://www.youtube.com/watch?v=_Oyy5PFOIcU)
- `YouTube` InfoQ: Mastering Chaos - A Netflix Guide to Microservices (https://www.youtube.com/watch?v=CZ3wIuvmHeM)
- `YouTube` freeCodeCamp.org: Microservice Architecture and System Design with Python & Kubernetes – Full Course (https://www.youtube.com/watch?v=hmkF77F9TLw)

Reflection and attributes:

- Reflection objects are used for obtaining type information at runtime. The classes that give access to the metadata of a running program are in the System.Reflection namespace.
- Writing code that reads object information and metadata at runtime
- Getting class names at runtime and creating objects of a class

Contents

- `WebSite` Microsoft Docs: Reflection (C#) (https://learn.microsoft.com/en-us/dotnet/csharp/programming-guide/concepts/reflection)
- `WebSite` GeeksforGeeks: Reflection in Python (https://www.geeksforgeeks.org/reflection-in-python/)
- `Article` Reflection in C# (https://medium.com/@ghadeer.kenawi/reflection-in-c-fa27d8f827f8)
- `Article` C# - Reflection (https://www.tutorialspoint.com/csharp/csharp_reflection.htm)
- `YouTube` Raw Coding: C# Reflection Explained (https://www.youtube.com/watch?v=cdG2JxuZvNI)

- YouTube Tutorials Point: C# - Reflection (https://www.youtube.com/watch?v=wfDFI9A56Gs)
- YouTube Derek Banas: Java Reflection Tutorial (https://www.youtube.com/watch?v=agnblS47F18)

MAUI:

- .NET Multi-platform App UI (.NET MAUI) is a cross-platform framework for creating native mobile and desktop apps with C# and XAML.
- Creating native desktop and mobile applications with C# and XAML
- Creating cross-platform applications
- Sharing user interface layout and design across platforms

Contents

- WebSite Microsoft Docs: What is .NET MAUI? (https://learn.microsoft.com/en-us/dotnet/maui/what-is-maui?view=net-maui-7.0)
- Article All About .NET MAUI (https://medium.com/volosoft/all-about-net-maui-5a1e774803ac)
- YouTube IAmTimCorey: .NET MAUI First Look - What is it, how do we use it, and is it ready (https://www.youtube.com/watch?v=HmyfjAaPW0g)
- YouTube James Montemagno: Learn .NET MAUI - Full Course for Beginners | Build cross-platform apps in C# (https://www.youtube.com/watch?v=DuNLR_NJv8U)

Auxiliary Skill: Infrastructure

Git & GitHub - Fundamentals:

- Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.
- GitHub is a hosting service for software development and version control using Git.

- Creating a repository
- Cloning a repository
- Committing, pushing and pulling to and from the repository
- Reversing a commit
- Creating branches and pull requests
- Handling merge and conflicts

Contents

- `WebSite` Git Reference Book (https://git-scm.com/book/en/v2)
- `WebSite` GitHub Documentation (https://docs.github.com/en)
- `WebSite` GitHub Pages Documentation (https://docs.github.com/en/pages/getting-started-with-github-pages/about-github-pages)
- `WebSite` W3Schools: Git Tutorial (https://www.w3schools.com/git/default.asp?remote=github)
- `WebSite` Git School - Visualizing Git (https://git-school.github.io/visualizing-git/)
- `WebSite` Dangit, Git!?! (https://dangitgit.com/)
- `Article` Git Tutorial - Explore The Commands And Operations In Git (https://medium.com/edureka/git-tutorial-da652b566ece)
- `Article` Git and Github Quickstart Tutorial (https://medium.com/@prashantramnyc/git-and-github-quickstart-tutorial-654a71594dca)
- `Article` Getting Started with Git and GitHub: A Complete Tutorial for Beginner (https://towardsdatascience.com/learn-basic-git-commands-for-your-data-science-works-2a75396d530d)
- `YouTube` Programming with Mosh: Git Tutorial for Beginners - Learn Git in 1 Hour (https://www.youtube.com/watch?v=8JJ101D3knE)
- `YouTube` freeCodeCamp.org: Git and GitHub for Beginners - Crash Course (https://www.youtube.com/watch?v=RGOj5yH7evk)

- YouTube Kevin Stratvert: Git and GitHub for Beginners Tutorial (https://www.youtube.com/watch?v=tRZGeaHPoaw)
- YouTube Tech With Tim: Git Tutorial for Beginners - Git & GitHub Fundamentals In Depth (https://www.youtube.com/watch?v=DVRQoVRzMIY)

HTTP - Fundamentals:

- HTTP stands for Hyper Text Transfer Protocol. Communication between client computers and web servers is done by sending HTTP Requests and receiving HTTP Responses.
- Understanding the difference between HTTP verbs
- Testing requests and checking the status codes in the browser
- Learning how to make a HTTP request on the command line with WGET
- Downloading an image with WGET
- Performing a POST

Contents

- WebSite W3Schools: What is HTTP? (https://www.w3schools.com/whatis/whatis_http.asp)
- WebSite MDN Web Docs: An overview of HTTP (https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview)
- WebSite MDN Web Docs: HTTP request methods (https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods)
- WebSite HTTP Cats (https://http.cat/)
- WebSite HTTP Dogs (https://http.dog/)
- Article HTTP codes as Valentine's Day comics (https://medium.com/@hanilim/http-codes-as-valentines-day-comics-8c03c805faa0)
- Article Here Are the most popular ways to make an HTTP request in JavaScript (https://medium.com/free-code-camp/here-is-the-most-popular-ways-to-make-an-http-request-in-javascript-954ce8c95aaa)

- YouTube Traversy Media: HTTP Crash Course & Exploration (https://www.youtube.com/watch?v=iYM2zFP3Zn0)
- YouTube Curious Code: HTTP Request Methods - GET, POST, PUT, DELETE (https://www.youtube.com/watch?v=tkfVQK6UxDI)
- YouTube freeCodeCamp.org: Postman Beginner's Course - API Testing (https://www.youtube.com/watch?v=VywxIQ2ZXw4)

JSON:

- JSON stands for JavaScript Object Notation. It is a text format for storing and transporting data.
- Creating an object
- Transforming an object into a string
- Transforming a string into an object
- Manipulating an object

Contents

- WebSite W3Schools: JSON (https://www.w3schools.com/js/js_json_intro.asp)
- WebSite MDN Web Docs: JSON (https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/JSON)
- WebSite MDN Web Docs: Working with JSON (https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Objects/JSON)
- YouTube Web Dev Simplified: Learn JSON in 10 Minutes (https://www.youtube.com/watch?v=iiADhChRriM)
- YouTube Traversy Media: JSON Crash Course (https://www.youtube.com/watch?v=wI1CWzNtE-M)

Command Line - Fundamentals:

- CLI is a command line program that accepts text input to execute operating system functions.
- Knowing the most important commands

Contents

- `WebSite` W3Schools: What is Command Line Interface (CLI)? (https://www.w3schools.com/whatis/whatis_cli.asp)
- `WebSite` Microsoft Docs: Using command line arguments for Windows Terminal (https://docs.microsoft.com/en-us/windows/terminal/command-line-arguments?tabs=windows)
- `Article` Advanced CLI: Commands You Should Know as a Developer (https://betterprogramming.pub/advanced-cli-commands-you-should-know-as-a-developer-7bc48c752a5e)
- `YouTube` freeCodeCamp.org: Command Line Crash Course (https://www.youtube.com/watch?v=yz7nYlnXLfE)
- `YouTube` Traversy Media: Command Line Crash Course For Beginners - Terminal Commands (https://www.youtube.com/watch?v=uwAqEzhyjtw)

Cloud - Fundamentals:

- Cloud, or cloud computing, is the distribution of computing services over the Internet using a pay-as-you-go pricing model. A cloud is composed of various computing resources, ranging from the computers themselves (or instances, in cloud terminology) to networks, storage, databases, and everything around them. In other words, everything that is normally needed to set up the equivalent of a server room, or even a complete data center, will be ready to use, configured, and run.
- Knowing the difference between IaaS, PaaS and SaaS
- Knowing the largest cloud providers
- Specializing in a specific provider of your choice

Contents

- `WebSite` Microsoft Azure: What is cloud computing? (https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-cloud-computing/)
- `WebSite` Amazon AWS: What is cloud computing? (https://aws.amazon.com/en/what-is-cloud-computing/)

- `Article` A beginner's guide to the basics of what cloud computing is about (https://scientya.com/a-beginners-guide-to-the-basics-of-what-cloud-computing-is-about-e8b3b7f25a30/)
- `Article` Cloud Computing for Beginners (https://medium.com/hackernoon/cloud-computing-for-beginners-85d168959afb/)
- `Article` What are Cloud Computing Services [IaaS, CaaS, PaaS, FaaS, SaaS] (https://medium.com/@nnilesh7756/what-are-cloud-computing-services-iaas-caas-paas-faas-saas-ac0f6022d36e)
- `YouTube` Simplilearn: Cloud Computing Tutorial for Beginners (https://www.youtube.com/watch?v=RWgW-CgdIk0)
- `YouTube` Amazon Web Services: What is Cloud Computing? - Amazon Web Services (https://www.youtube.com/watch?v=mxT233EdY5c)
- `YouTube` Ecourse Review: Cloud Computing Services Models - IaaS PaaS SaaS Explained (https://www.youtube.com/watch?v=36zducUX16w/)

SQL - Fundamentals:

- Structured Query Language (SQL) is a standardized programming language that is used to manage relational databases and perform various operations on the data in them.
- Knowing the most common SQL commands
- Using SELECT to query a table
- Using INSERT to insert data into a table
- Using UPDATE to update a table
- Using DELETE to remove data from a table
- Using JOIN to connect data from multiple tables
- Knowing the clauses (FROM, ORDER BY, etc.)

Contents

- `WebSite` W3Schools: Introduction to SQL (https://www.w3schools.com/sql/sql_intro.asp)

- `WebSite` Amazon AWS: What is SQL? (https://aws.amazon.com/what-is/sql/?nc1=h_ls)
- `Article` From Basic to Intermediate SQL in 10 Minutes (https://medium.com/analytics-vidhya/from-basic-to-intermediate-sql-in-10-minutes-42b960ed6f9e)
- `Article` SQL for Beginners 2022 — A Practical SQL Guide (https://medium.com/@besimum/sql-for-beginners-2022-a-practical-sql-guide-1de6adb43d52)
- `YouTube` Kevin Stratvert: SQL for Beginners Tutorial (https://www.youtube.com/watch?v=h0nxCDiD-zg)
- `YouTube` Fireship: Learn SQL in 23 Easy Steps (https://www.youtube.com/watch?v=Cz3WcZLRaWc)
- `YouTube` freeCodeCamp.org: SQL Tutorial - Full Database Course for Beginners (https://www.youtube.com/watch?v=HXV3zeQKqGY)

Auxiliary Skill: Good practices

SOLID:

- SOLID has five principles that are considered best practices in software development that help programmers write cleaner code by separating responsibilities, reducing docking, easing refactoring, and encouraging code reuse.

Contents

- `Article` The S.O.L.I.D Principles in Pictures (https://medium.com/backticks-tildes/the-s-o-l-i-d-principles-in-pictures-b34ce2f1e898)
- `Article` SOLID Principles every Developer Should Know (https://blog.bitsrc.io/solid-principles-every-developer-should-know-b3bfa96bb688)
- `YouTube` Fireship: Solid in 100 Seconds (https://www.youtube.com/watch?v=q0BGgQJcp7w)

- YouTube A Dev' Story: SOLID Design Principles Explained in a Nutshell (https://www.youtube.com/watch?v=69sfWNzxTMc)
- YouTube Web Dev Simplified: Single Responsibility Principle Explained - SOLID Design Principles (https://www.youtube.com/watch?v=UQqY3_6Epbg)

Design Patterns:

- In software engineering, a Design Pattern is a general, reusable solution to a commonly occurring problem within a given context in software design. It is a description or template for how to solve a problem that can be used in many different situations. Design Patterns are formalized best practices that the programmer can use to solve common problems when designing an application or system.
- Getting familiarized with and applying the main Design Patterns

Contents

- WebSite What's a design pattern? (https://refactoring.guru/design-patterns/what-is-pattern)
- Article Design Patterns — Introduction (https://henriquesd.medium.com/design-patterns-introduction-220f811db857)
- Article What Are Design Patterns and Do I Need Them? (https://www.developer.com/design/what-are-design-patterns-and-do-i-need-them/)
- Article The 3 Types of Design Patterns All Developers Should Know (with code examples of each) (https://www.freecodecamp.org/news/the-basic-design-patterns-all-developers-need-to-know/)
- Article 10 Design Patterns every Software Architect and Software Engineer must know (https://ravindraelicherla.medium.com/10-design-patterns-every-software-architect-must-know-b33237bc01c2)
- Article Modern-Day Architecture Design Patterns for Software Professionals (https://betterprogramming.pub/modern-day-architecture-design-patterns-for-software-professionals-9056ee1ed977)

- YouTube Web Dev Simplified: Design Patterns (Playlist) (https://www.youtube.com/playlist?list=PLZlA0Gpn_vH_CthENcPCM0Dww6a5XYC7f)
- YouTube Programming with Mosh: Design Patterns in Plain English (https://www.youtube.com/watch?v=NU_1StN5Tkk)
- YouTube Fireship: 10 Design Patterns Explained in 10 Minutes (https://www.youtube.com/watch?v=tv-_1er1mWI)
- YouTube Traversy Media: 5 Design Patterns Every Engineer Should Know (https://www.YOUTUBE.com/watch?v=AWOf6Wo6gtg)

Clean Architecture:

- Clean architecture is a way of developing software, such that just by looking at the source code of a program, you should be able to tell what the program does.

Contents

- Article The Clean Architecture—Beginner's Guide (https://betterprogramming.pub/the-clean-architecture-beginners-guide-e4b7058c1165)
- Article Thoughts on Clean Architecture (https://medium.com/android-news/thoughts-on-clean-architecture-b8449d9d02df)
- Article Clean Architecture, the right way (https://medium.com/gdg-vit/clean-architecture-the-right-way-d83b81ecac6)
- Article Why use a Clean Architecture (https://www.mytaskpanel.com/the-5-advantages-of-using-a-clean-architecture-all-you-need-to-know/)
- Article Clean Architecture – Build Software like an Artisan (https://flexiple.com/developers/clean-architecture-build-software-like-an-artisan/)
- Article React Clean Architecture (https://kpiteng.medium.com/react-clean-architecture-e4144a0788b6)
- Article Clean Architecture with Java 11 (https://medium.com/slalom-build/clean-architecture-with-java-11-f78bba431041)

- `Article` A detailed guide on developing Android apps using the Clean Architecture pattern (https://medium.com/@dmilicic/a-detailed-guide-on-developing-android-apps-using-the-clean-architecture-pattern-d38d71e94029)
- `YouTube` CodeOpinion: Clean Architecture Example & Breakdown (https://www.youtube.com/watch?v=Ys_W6MyWOCw)
- `YouTube` Women Who Code: Clean Architecture Concepts (https://www.youtube.com/watch?v=ouBSPdvbzvw)
- `YouTube` CodandoTV(Rods) - Simplifying Clean Architecture +MVVM in your mobile application - COMPLETE GUIDE (https://youtu.be/8ehlZfyN1S0?si=l7-l5l4zsnYLlSjD)

Clean Code:

- Applying simple techniques that aim to make a code easier to write and read
- Refactoring your code to make it clearer

Contents

- `Article` What is Clean Code? (https://garywoodfine.com/what-is-clean-code/)
- `Article` Clean Code Explained – A Practical Introduction to Clean Coding for Beginners (https://www.freecodecamp.org/news/clean-coding-for-beginners/)
- `YouTube` UnityCoin: Clean Code - Uncle Bob - Lesson 1 (https://www.youtube.com/watch?v=7EmboKQH8lM)
- `YouTube` Programming with Mosh: Clean Code - Learn to write clean, maintainable and robust code (https://www.youtube.com/watch?v=5koPpYVa020)

Domain-Driven Design (DDD) Concepts:

- Domain-Driven Design (DDD) is an approach to software design and development that is first informed by business requirements. The program

components (objects, classes, arrays, etc.) indicate the industry, sector, or business domain in which the business operates.

- Modeling domains effectively
- Basing complex projects on domain models
- Getting to know the building blocks of DDD

Contents

- `Article` Domain-Driven Design: Everything You Always Wanted to Know About it, But Were Afraid to Ask (https://medium.com/ssense-tech/domain-driven-design-everything-you-always-wanted-to-know-about-it-but-were-afraid-to-ask-a85e7b74497a)
- `Article` Domain-Driven Design in the era of Microservices (https://medium.com/it-dead-inside/domain-driven-design-in-the-era-of-icroservices-de2be01821ed)
- `YouTube` Amichai Mantinband: Domain-Driven Design in 150 Seconds (https://www.youtube.com/watch?v=8Z5IAkWcnIw)
- `YouTube` Domain-Driven Design Europe: What is DDD? (Eric Evans - DDD Europe 2019) (https://www.youtube.com/watch?v=pMuiVlnGqjk)
