

Get all the support you need

ICT Trainings offers lots of support to help you prepare for your future.

Find a wide range of support, including official ICT preparation materials –
www.ictrainings.com/

Find quizzes and learning tips on our Facebook page –
www.facebook.com/ICTtrainingsLahore

Register now

You can take the course in a class or online on computer.

Step 1: Visit ICT Trainings Centre

Choose from over 50 courses.

Step 2: Choose from over 50 Courses

Find out about registering, costs and preparation courses.

Step 3: Choose your starting date and time

We have dates every month – and you can register just one week before taking your course.

Find out more at

www.ict-trainings.com

Show you are serious about your career

Join millions of people that have selected IT and using in official/daily chores.

ICT Certificates to show they have the work-focused

IT and Telecom skills for career success. These high-quality international courses are developed by Industry leading professionals

Boost your employability – get the IT skills you need to access the best jobs.

Excel in business-related studies with proven IT skills.

Recognized by top local & multinationals as a quality delivering educational institution.

Contact us

ICT Trainings Lahore

349 H/3, Johar Town
Lahore
Punjab
Pakistan
Cell: +92 321 3499131
LandLine: +92 42 35951406-7



Find us:

website: ict-trainings.com

<http://facebook.com/ICTtrainingsLahore>

email: info@ict-trainings.com

www.ict-trainings.com/curriculum/csharp-leaflet.pdf

ICT develop and produce the most valuable range of qualifications for learners of IT and Telecom.

Over 20000 students trained in Two Decades.

Universities, employers, government, ministries and other organizations recognize us as valuable partner.

ICT Trainings Institute – Engineering Your Career

C Sharp



Course Guide

C# (pronounced C sharp) is a multi-paradigm programming language encompassing strong typing, imperative, declarative, functional, generic, object-oriented (class-based), and component-oriented programming disciplines.



Microsoft is globally accepted exam

The Microsoft exams are trusted and accepted by leading employers, educational institutions and governments worldwide.

For a full list of organizations using these market-leading exams, go to www.microsoft.com/learning

C Sharp Course Contents

C Sharp training curriculum follows the Microsoft Programming C Sharp 70-483 is designed to meet the requirements of next generation programming needs.

Manage program flow (25%)

Implement multithreading and asynchronous processing

- Use the Task Parallel library (ParallelFor, Plinq, Tasks); create continuation tasks; spawn threads by using ThreadPool; unblock the UI; use async and await keywords; manage data by using concurrent collections

Manage multithreading

- Synchronize resources; implement locking; cancel a long-running task; implement thread-safe methods to handle race conditions

Implement program flow

- Iterate across collection and array items; program decisions by using switch statements, if/then, and operators; evaluate expressions

Create and implement events and callbacks

- Create event handlers; subscribe to and unsubscribe from events; use built-in delegate types to create events; create delegates; lambda expressions; anonymous methods

Implement exception handling

- Handle exception types (SQL exceptions, network exceptions, communication exceptions, network timeout exceptions); catch typed vs. base exceptions; implement try-catch-finally blocks; throw exceptions; determine when to rethrow vs. throw; create custom exceptions

Create and use types (24%)

Create types

- Create value types (structs, enum), reference types, generic types, constructors, static variables, methods, classes, extension methods, optional and named parameters, and indexed properties; create overloaded and overridden methods

Consume types

- Box or unbox to convert between value types; cast types; convert types; handle dynamic types; ensure interoperability with unmanaged code, for example, dynamic keyword

Enforce encapsulation

- Enforce encapsulation by using properties, by using accessors (public, private, protected), and by using explicit interface implementation

Create and implement a class hierarchy

- Design and implement an interface; inherit from a base class; create and implement classes based on the IComparable, IEnumerable, IDisposable, and IUnknown interfaces

Find, execute, and create types at runtime by using reflection

- Create and apply attributes; read attributes; generate code at runtime by using CodeDom and lambda expressions; use types from the System.Reflection namespace (Assembly, PropertyInfo, MethodInfo, Type)

Manage the object life cycle

- Manage unmanaged resources; implement IDisposable, including interaction with finalization; manage IDisposable by using the Using statement; manage finalization and garbage collection

Manipulate strings

- Manipulate strings by using the StringBuilder, StringWriter, and StringReader classes; search strings; enumerate string methods; format strings

Debug applications and implement security (25%)

Validate application input

- Validate JSON data; data collection types; manage data integrity; evaluate a regular expression to validate the input format; use built-in functions to validate data type and content out of scope: writing regular expressions

Perform symmetric and asymmetric encryption

- Choose an appropriate encryption algorithm; manage and create certificates; implement key management; implement the System.Security namespace; hashing data; encrypt streams

Manage assemblies

- Version assemblies; sign assemblies using strong names; implement side-by-side hosting; put an assembly in the global assembly cache; create a WinMD assembly

Debug an application

- Create and manage compiler directives; choose an appropriate build type; manage programming database files and symbols

Implement diagnostics in an application

- Implement logging and tracing; profiling applications; create and monitor performance counters; write to the event log

Implement data access (26%)

Perform I/O operations

- Read and write files and streams; read and write from the network by using classes in the System.Net namespace; implement asynchronous I/O operations

Consume data

- Retrieve data from a database; update data in a database; consume JSON and XML data; retrieve data by using web services

Query and manipulate data and objects by using LINQ

- Query data by using operators (projection, join, group, take, skip, aggregate); create method-based LINQ queries; query data by using query comprehension syntax; select data by using anonymous types; force execution of a query; read, filter, create, and modify data structures by using LINQ to XML

Serialize and deserialize data

- Serialize and deserialize data by using binary serialization, custom serialization, XML Serializer, JSON Serializer, and Data Contract Serializer

Store data in and retrieve data from collections

- Store and retrieve data by using dictionaries, arrays, lists, sets, and queues; choose a collection type; initialize a collection; add and remove items from a collection; use typed vs. non-typed collections; implement custom collections; implement collection interfaces

Practical Approach

- A real example will be given throughout the lectures, starting from design, analysis, implementation and maintenance.

www.ict-trainings.com/curriculum/csharp-leaflet.pdf