

Software Development

About the Course :-

Software developers are in high demand in the current job market, and computer programming is a prerequisite skill for success in this field.

Start your journey toward becoming a professional software developer by learning Java, one of the industry's most commonly used programming languages.

This course, part of the CS Essentials for Software Development Professional Certificate program, will quickly cover Java syntax and keywords and then explore features of object-oriented programming including encapsulation, inheritance, and polymorphism. You will learn how to apply these concepts to programmatic problem solving by investigating class modeling techniques and relationships such as aggregation, realization, and generalization.

In addition to programming, you will learn about software testing techniques that help us find problems in our code, and you will use modern development environments and tools for tasks like debugging and unit testing. We will introduce Eclipse, the eclipse debugger and Junit (a unit testing framework).

After completing this course, you will be able to design, develop, and test large applications in Java and understand and apply core principles of professional software development.

Career Prospect:-

There are different levels of software engineering and promotion is usually dependent upon both your ability and experience.

An entry-level post typically involves working under supervision, formulating the scope of, and objectives, for systems and designing code. After around three years, you may progress to preparing detailed specifications from which programs may be written and be competent to work at the highest technical level.

A senior software engineer often supervises the activities of a team of developers for a large project or several smaller projects. This usually requires a minimum of five years' experience.

Progression is mainly into management via team leadership and project management roles, or to designer/consultant via technical specialisation. Transfer between organisations for advancement is often possible.

Software engineers are specialists and may remain within one organisation, and frequently in one role, for much of their working lives. However, others expand and develop their roles to the extent that software engineering becomes part of a much wider role in positions such as technical writing, IT training and education. They can also move on to the role of software architect.

Both BCS and IAP provide up-to-date information and advice on career development.

Course Content :-

Course Outline

Lesson 1: Introduction to Programming 1

Objective Domain Matrix 1

Key Terms 1

Understanding Computer Programming 2

Introducing Algorithms 2

Introducing C# 4

Understanding Decision Structures 11

The If Statement 11

The If-Else Statement 13

The Switch Statement 15

Understanding Repetition Structures 17

Understanding the While Loop 17

Understanding the Do-While Loop 19

Understanding the For Loop 20

Understanding the Foreach Loop 21

Understanding Recursion 22

Understanding Exception Handling 23

Handling Exceptions 24

Using Try-Catch-Finally 25

Skill Summary 26

Knowledge Assessment 27

Competency Assessment 30

Proficiency Assessment 31

Lesson 2: Introduction to Object-Oriented Programming 32

Objective Domain Matrix 32

Key Terms 32

Understanding Objects 33

Thinking in an Object-Oriented Way 33

Understanding Classes 33

Understanding Values and References 44

Understanding Structs 44
Understanding Memory Allocation 45
Understanding Encapsulation 47
Understanding Access Modifiers 48
Understanding Inheritance 48
Understanding Abstract and Sealed Classes 50
Inheriting from the Object Class 51
Casting between Types 52
Understanding Polymorphism 53
Understanding the Override and New Keywords 55
Understanding Interfaces 56
Skill Summary 58
Knowledge Assessment 59
Competency Assessment 64
Proficiency Assessment 64

Lesson 3: Understanding General Software Development 65

Objective Domain Matrix 65
Key Terms 65
Understanding Application Lifecycle Management 66
Understanding Requirements Analysis 66
Understanding the Design Process 67
Understanding Software Development 67
Understanding Software Testing 68
Understanding Release Management 68
Understanding Testing 68
Understanding Testing Methods 69
Understanding Testing Levels 69
Understanding Data Structures 70
Understanding Arrays 70
Understanding Queues 72
Understanding Stacks 73
Linked Lists 74
Understanding Sorting Algorithms 76
Understanding BubbleSort 76
Understanding QuickSort 79
Skill Summary 81
Knowledge Assessment 81
Competency Assessment 83
Proficiency Assessment 84

Lesson 4: Understanding Web Applications 85

Objective Domain Matrix 85
Key Terms 85
Understanding Web Page Development 85
Understanding HTML 86
Understanding Cascading Style Sheets 88
Understanding JavaScript 92
Understanding Client-Side vs. Server-Side Programming 94
Understanding ASP.NET Application
Development 95
Understanding ASP.NET Page Life Cycle and Event
Model 96
Understanding State Management 99
Understanding IIS Web Hosting 104
Understanding Internet Information Services 105
Creating Virtual Directories and Web Sites 105
Deploying Web Applications 106
Understanding Web Services Development 107
Introducing SOAP 107
Introducing WSDL 108
Creating Web Services 108
Consuming Web Services 112
Skill Summary 115
Knowledge Assessment 115
Competency Assessment 118
Proficiency Assessment 119

Lesson 5: Understanding Desktop Applications 120

Objective Domain Matrix 120
Key Terms 120
Understanding Windows Forms Applications 120
Designing a Windows Form 121
Understanding the Windows Form Event Model 123
Using Visual Inheritance 123
Understanding Multiple Document Interface (MDI) Applications 126
Understanding Console-Based Applications 129
Working with Command-Line Parameters 130
Understanding Windows Services 131
Creating a Windows Service 132
Skill Summary 137
Knowledge Assessment 138
Competency Assessment 140
Proficiency Assessment 141

Lesson 6: Understanding Databases 142

Objective Domain Matrix 142

Key Terms 142

Understanding Relational Database Concepts 142

Understanding Databases 143

Understanding Relational Database Concepts 143

Understanding Relational Database Design 144

Understanding Entity-Relationship Diagrams 144

Understanding Data Normalization 146

Understanding Database Query Methods 149

Working with SQL Queries 150

Working with Stored Procedures 159

Understanding Database Connection
Methods 164

Working with Flat Files 164

Working with XML 167

Working with DataSet 170

Skill Summary 173

Knowledge Assessment 173

Competency Assessment 176

Proficiency Assessment 176

Salary Offered :-

Software Developer Salary. A Software Developer earns an average salary of **Rs390,970 per year**. People in this job generally don't have more than 10 years' experience. Skills that are associated with high pay for this job are C++ Programming Language and Java/J2EE.