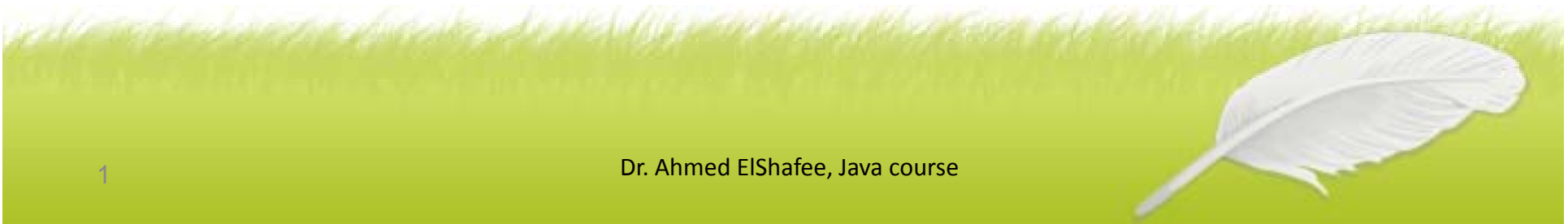




How to Become a Programmer

Dr. Ahmed ElShafee



Agenda

1. Our agreement
2. Steps
3. what type of programmer you are/want to be?

Our agreement

- It may or may not be true that you can learn a programming language in 7 or 21 days, and it may or may not depend on what book you read or what course you take.
- However, it does take a lot longer than that to gain experience in how to use a language most effectively.
- It is a cumulative process that builds up your skills day after day and year after year.
- Programming can also be fun and rewarding (mentally, spiritually and financially).



-
- This course does not promise to give a magical easy way to becoming a programmer, but you'll get a general outline of how to become a programmer in one of the most popular programming language



Steps

Step1: Prepare yourself for becoming a programmer:

1. Take an introductory course in logic, discrete mathematics or both.
2. Learn at least one of the following program conception techniques. As you advance in programming you may find them useful.
 - * Flow Charting (globally considered the easiest to learn)
 - * Pseudo-Code
 - * Unified Modeling Language (UML)
 - * Object Relational Mapping (ORM)

3. Learn database concepts such as tables, views/queries and procedures.

You can use any simple database package to do this, such as MS access, Fox Pro, Paradox.

MySQL is a particularly good database to learn because it's free, commonly used, and databases are commonly accessed with SQL queries.

4. Learn about programming paradigms (an introduction is enough for now), the most important being procedural, object oriented, functional, logic and declarative programming.

Step2: Decide what type of programmer you are/want to be.

Programmers generally fall under one of the following categories:

- Web Programmer
- Desktop Application Programmer
 - Operating System (OS) Oriented (tied to a single operating system or set of operating systems)
 - Platform Independent
- Distributed Applications Programmer
- Library/Platform/Framework/Core Programmer

-
- System Programmer
 - Kernel Programmer
 - Driver Programmer
 - Compiler Programmer
 - Programming Scientist

Step 3: Learn the technologies and programming languages
related to your programming field of choice.

The following sections break down the tasks for different types of programming.

Web Programmers

- Web applications are software components designed to work on top of the internet architecture.
- This means that the applications is accessed through a web browser software such as Firefox or Internet Explorer.
- Many applications become web based application (DBs, ERPs,...)



Steps

1. Get familiar with website structuring (creating conceptual web diagrams, site-maps, and navigation structures).
2. Learn the HTML and CSS languages,
3. Create simple static websites until you are familiar with and comfortable around HTML
4. Learn a client-side scripting language
5. Learn at least one server side programming language.

Desktop Application Programmers

- Desktop applications do not require a browser and work on their own.
- The browser program itself is a desktop application

Steps:

- Learn procedural programming languages.
- Learn object oriented programming languages.
- Learn visual programming concepts using one of the languages you learned
- Learn at least one advanced modeling technique such as UML or ORM.

Distributed Applications Programmers

- Distributed applications use services and serviced components to *distribute* the workload of a big computing task
- Distributed application programming is considered by many to be one of the hardest to learn and requires diverse knowledge in computer and communication technologies.

Steps

- Understand Networking protocols, architectures and devices.
- prepare yourself for desktop application programming
- Study distributed applications design and architectures
- Learn about building serviced components and services using your programming language of choice.

-
- Learn one or more of the following technologies.
 - Common Object Request Broker Architecture (CORBA)
 - Distributed Component Object Model (DCOM)
 - .NET Remoting
 - XML Web Services

Library/Platform/Framework/Core Programmers

- Core programmers are merely advanced programmers who made the transfer from programming applications to programming code units to be used by other programmers.

System Programmers

- systems programming aims to produce software which provides services to the computer hardware (e.g. disk defragmenter).
- It requires a greater degree of hardware awareness.

Programming Scientists

- Programming scientists are very advanced programmers, who instead of working on developing applications, work on developing computing technologies such as encryption, programming languages and data mining algorithms.
- This level is seldom achieved without academic study and dedication.



Thanks,
See you next Lecture, isA

