Today’s “Friendly Conversation” topic… ~3-5min

Agile Manifesto

“We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.”

Object-Oriented Programming

Session: Week 1 Session 1
Instructor: Eric Pogue

Agenda:

1. Friendly Conversation & Good Natured Banter… let's make sure that everyone gets a good seat where they can hear and speak comfortably
2. Welcome & Introductions*
3. Review Course Syllabus
4. Your Time for Introductions
5. Introduction to Object-Oriented Programming
6. Assignment and Lab… and a first attempt at self-organizing lab ("Scrum") teams
7. Assignment
8. Lab Time… and Report-Out
9. Wrap-up and Final Questions/Comments?

Discussion & Questions welcome at any time… please be present with no phones or email during our discussion time

Welcome & Introductions

This is:

Object-Oriented Programming

TTh 2:00 to 3:15pm CST

AS 104A

And I am:

Eric Pogue

Introduction Topics:

Full and Preferred Name
Family, Home, College background
Programming experience
Likely programming environment
Top two or three things that you would like to get out of this class
Hobby or two
Unique fun fact about yourself

Welcome & Introductions

Full and Preferred Name:

Eric Pogue

Eric, Mr. Pogue, or Professor

Family, Home, College background:

Married with five children, recently relocated from Davenport, IA to Chicago area

Undergraduate in CS and Masters in Business… teaching online/evening for many years

Programming experience:

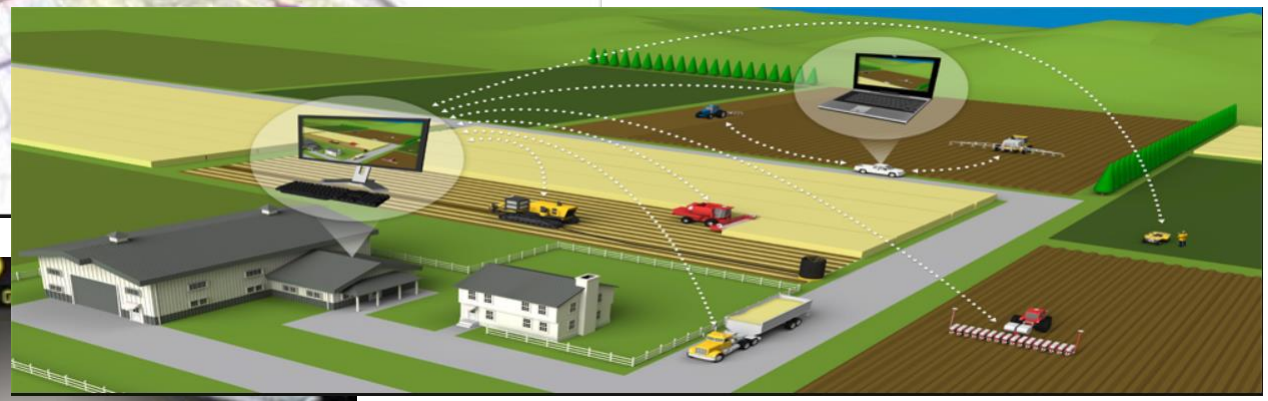
Decades in the industry as a developer, architect, project manager, division manager, and vice president of various software development organizations.

Part of many teams that have delivered products to ten's of millions of customers globally

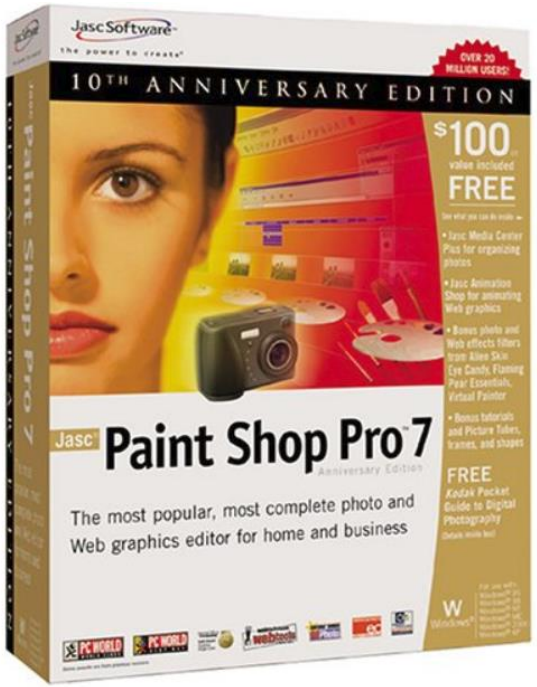
Parsons Technology, Intuit, The Learning Company, Jasc Software, and John Deere

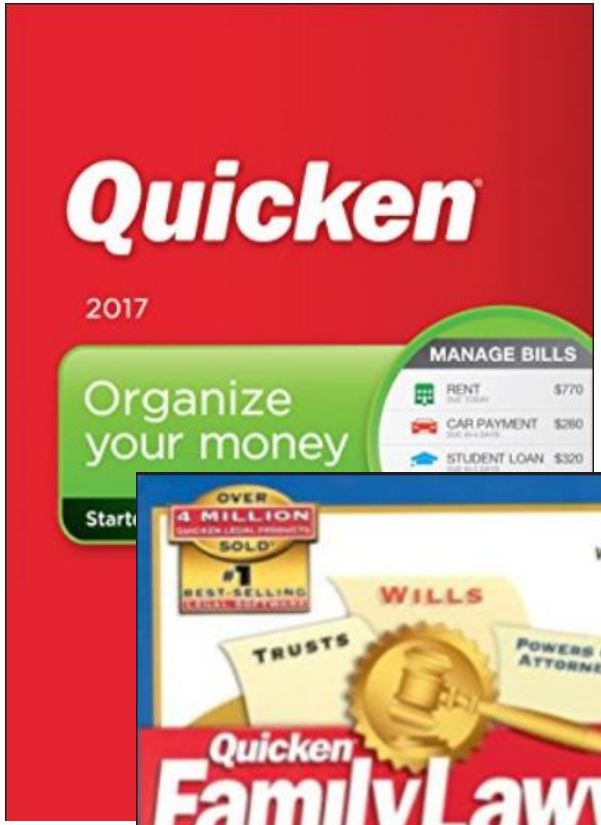
… and currently working on a startup with my oldest son.

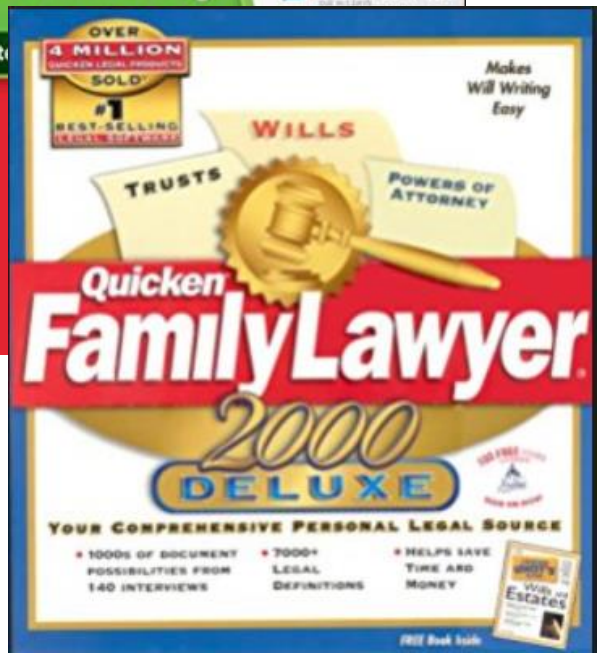

JDLink
Wireless Communications




Maschinen von Kunden oder Vertriebspartnern
Computer
JDLink
Wireless Communications
INTERNET
Service-Netzwerk


Jasc Software
the power to create
OVER 20 MILLION USERS!
10TH ANNIVERSARY EDITION
$100 value included FREE
Jasc Paint Shop Pro 7
Anniversary Edition
The most popular, most complete photo and Web graphics editor for home and business
FREE Kodak Pocket Guide to Digital Photography


Quicken
2017
Organize your money
MANAGE BILLS
RENT $770
CAR PAYMENT $260
STUDENT LOAN $320


OVER 4 MILLION SOLD
#1 BEST-SELLING
Makes Will Writing Easy
WILLS
TRUSTS
POWERS OF ATTORNEY
Quicken
FamilyLawyer
2000
DELUXE
YOUR COMPREHENSIVE PERSONAL LEGAL SOURCE
1000s OF DOCUMENT POSSIBILITIES FROM 140 INTERVIEWS
7000+ LEGAL DEFINITIONS
HELPS SAVE TIME AND MONEY

Welcome & Introductions

Likely programming environment

Personal Laptop, Windows 10, Chrome browser, and Visual Studio Code text editor

Hobbies:

Wilderness Canoeing & Camping (Quetico) and Triathlons

Top two or three things that you would like to get out of this class

- **help each of you be successful in this class**
- **explore software development processes and techniques together and motivate you to look deeper**
- **and for us to find a little enjoyment and fun along the way***
- **… oh yes, and it would be wonderful if I could help you build something that made you proud during the semester**

Fun Fact:

At one point I had the very dubious "honor" or being the most traveled John Deere employee to India with 40+ trips over a 5-6 year period while setting up the 400+ person John Deere Technology Center – India application development organization.

Introductions

Introduction – Please Introduce Yourself

Full and Preferred Name:

-
-

Family, Home, College background:

-
-

Programming experience:

-
-

Please rate yourself in the following technical areas on a scale of 1 to 5

__ Java
__ C++
__ C#
__ Python
__ Android

Hobbies:

-
-

Top two or three things that you would like to get from this class

-
-
-

Fun Fact:

-

Syllabus Overview

Your Turn for Introductions

Introduction – Please Introduce Yourself

Full and Preferred Name:

-
-

Family, Home, College background:

-
-

Programming experience:

-
-

Please rate yourself in the following technical areas on a scale of 1 to 5

__ Java
__ C++
__ C#
__ Python
__ Android

Hobbies:

-
-

Top two or three things that you would like to get from this class

-
-
-

Fun Fact:

-

Object-Oriented Programming

Session: Week 1 Session 1
Instructor: Eric Pogue

LEWIS UNIVERSITY
A Catholic and Lasallian University

Agenda:

1. Friendly Conversation & Good Natured Banter… let's make sure that everyone gets a good seat where they can hear and speak comfortably
2. Welcome & Introductions*
3. Review Course Syllabus
4. Your Time for Introductions
5. Introduction to Object-Oriented Programming
6. Assignment
7. Lab Time… and Lab Report-Out
8. Wrap-up and Final Questions/Comments?

Discussion & Questions welcome at any time… please be present with no phones or email during our discussion time

Introduction to Object-Oriented Programming

Assignment

Prior to class on Thursday:

1. View and reflect on “Introduction to Scrum in 7 Minutes” video [link]
2. Install/implement MS PowerShell [link]
3. Install Git Client [link]
4. Utilize (local) Git Client to create, update, branch, and merge a local project utilizing a beginners level tutorial… one option would be “An Intro to Git and GitHub for Beginners” [link]

Estimate: 3 hours… “Fist of Five” Agile polling

Lab Time ~30min
Lab (“Stand Up”) report out at 3:10

Lab Assignment:

1. Split into “Two-Pizza” sized Lab (“Scrum”) teams (3-4 people per team) ~2min
2. Elect a team leader (“Scrum Master”) ~2mim
3. Give your team a name… a specific type of wild animal ~1min
4. Spend our lab time helping each other accomplish our assignment for Thursday
5. Perform a team “Fist of Five” polling on the assignment and alternatives at 3:08 ~2min
6. Be ready for a <1 minute Lab (“Stand Up”) report-out where the team leader (“Scrum Master”) will report your progress, capacity, and commitment on the assignment items
7. Optionally, continue to assist each other with today’s assignment outside of class

Wrap-up and Final Questions/Comments?

End of Session

Instructor: Eric Pogue

Sparse header

- Sparse body

Dense Header

1. Describe what a thread is and why it can be useful to distribute tasks among multiple threads
2. Review our multi-threaded application development activities
3. Explain why it is important to synchronize threads that need to share data source access
4. Review Object Oriented Programming benefits including the associating Data & Functionality, Encapsulation & Information Hiding, Inheritance, and Polymorphism
5. Review databases, database servers, and the SQL language
6. Understand how databases support (or don't support)work within a Object Oriented Programming environment
7. Understand client-server (two-tier), three-tier, and n-tier architectures
8. Introduce network programming concepts
9. Understand Web Services network programming
10. Develop a middle-tier data server using network programming

- We covered these topics in week 5/6 and week 1 respectively.

Default Header

- Default Body

How to Be a Successful Programmer

Top-ten list on how to be a successful programmer, and successfully complete course project:

1. Start early on project assignments
2. Establish a physical work environment that allows you to focus for extended periods of time
3. Become comfortable with your development environment/tools
4. Read or re-read the project assignment and related materials
5. Suggest changes to the assignment if you feel there is a better (or more entertaining) way
6. Save and deploy working versions of your project regularly… this allows you to experiment without risking what you have already accomplished
7. Ask for help if you are stuck… often simply articulating the problem/question will lead you to finding your own answer
8. Look for similar examples… but write your own code that you understand
9. Come to class and participate in class exercises… ask questions during or after class
10. Review the textbook and review the lecture slides