
salesforce

TRAILHEADX
SALESFORCE DEVELOPER CONFERENCE

Intro to Apex

For Developers

Leah McGowan-Hare

@LeahBMH

Leah McGowan-Hare

Director,
Trailhead for Technology and Products

Forward-Looking Statements

Statement under the Private Securities Litigation Reform Act of 1995:

This presentation may contain forward-looking statements that involve risks, uncertainties, and assumptions. If any such uncertainties materialize or if any of the assumptions proves incorrect, the results of salesforce.com, inc. could differ materially from the results expressed or implied by the forward-looking statements we make. All statements other than statements of historical fact could be deemed forward-looking, including any projections of product or service availability, subscriber growth, earnings, revenues, or other financial items and any statements regarding strategies or plans of management for future operations, statements of belief, any statements concerning new, planned, or upgraded services or technology developments and customer contracts or use of our services.

The risks and uncertainties referred to above include – but are not limited to – risks associated with developing and delivering new functionality for our service, new products and services, our new business model, our past operating losses, possible fluctuations in our operating results and rate of growth, interruptions or delays in our Web hosting, breach of our security measures, the outcome of any litigation, risks associated with completed and any possible mergers and acquisitions, the immature market in which we operate, our relatively limited operating history, our ability to expand, retain, and motivate our employees and manage our growth, new releases of our service and successful customer deployment, our limited history reselling non-salesforce.com products, and utilization and selling to larger enterprise customers. Further information on potential factors that could affect the financial results of salesforce.com, inc. is included in our annual report on Form 10-K for the most recent fiscal year and in our quarterly report on Form 10-Q for the most recent fiscal quarter. These documents and others containing important disclosures are available on the SEC Filings section of the Investor Information section of our Web site.

Any unreleased services or features referenced in this or other presentations, press releases or public statements are not currently available and may not be delivered on time or at all. Customers who purchase our services should make the purchase decisions based upon features that are currently available. Salesforce.com, inc. assumes no obligation and does not intend to update these forward-looking statements.

Go Social!


@SalesforceDevs


Salesforce Developers


Salesforce Developers

Salesforce Developers


+Salesforce Developers

Expectations

You are an experienced developer

You are curious about Apex but no experience

You are ready to learn something

You are ready to build something, too.

It is a 90 minute workshop: you will not leave here an expert.

Agenda

Platform Overview
Writing Apex Classes
Accessing Data using SOQL
Writing Triggers

Extra Credit:
Writing Visualforce Pages
Writing Controller Extensions
Using the REST APIs
Unit Testing
Batching and Scheduling

How Will We Do It?

bit.ly/tdx-conf-app

Build a Conference Management App

Build a simple event management app with Apex and Visualforce.

Beginner | Developer

Project

Free Developer Environment

http://developer.salesforce.com/signup

App Cloud Makes Building Apps 70% Faster

The fastest path from idea to App

Source: IDC White Paper, sponsored by Salesforce.com, Salesforce Platform: Accelerate App Dev with Huge ROI, Doc #246505, Feb 2014

App Cloud Gives You Tools for Building Any App

Full spectrum of capabilities from enterprise control to elastic flexibility

Two Approaches to Development

Declarative Approach		Programmatic Approach
Page Layouts Record Types Lightning App Builder	User Interface	Visualforce Pages Visualforce Components Lightning Components
Formula Fields Validation Rules Workflows and Approvals Process Builder	Business Logic	Apex Controllers Apex Triggers
Custom Objects Custom Fields Relationships	Data Model	Metadata API REST API Bulk API

The Conference App

What we'll build…

Schema for sessions and speakers (to be installed via package)

Automatically send confirmation emails

Perform validation against other data

Customized user interface with Visualforce Pages

Upload speaker pictures

Flickr integration (Apex) to show conference pictures

Lab 1: Setup Your Developer Org

- Create org (if not already done)

  developer.salesforce.com/signup
- Install Package

  bit.ly/tdx-16-apex-test-1
- Verify Steps 1 and 2 of Project

  bit.ly/tdx-conf-app

Writing Apex Classes

What is Apex?

Force.com-native programming language

Object-oriented (Classes, Interfaces, Inheritance)

Tenant Secure

Syntactically similar to Java and C#

Compiles to Java bytecode

Strongly typed

I've Heard Apex is a Bit Funny

Not a stand-alone language

Compiled only on the server…no local compiler

Governor limits

Apex Syntax Constructs

Primitive data types

Flow control (if, for, while, …)

Exception handling

Collections: List, Set, Map

Case insensitive

Single-quotes for strings: 'Joe'

Lines terminated with semicolons

Apex Language Features

Id data type

SObject class

Built-in support for data access

Built-in test framework

Apex Class

```
public class MortgageCalculator {

    public Double amount { get; set; }
    public Double rate { get; set; }
    public Integer years { get; set; }

    public Double calculateMonthlyPayment() {
        Integer months = years * 12;
        Double monthlyRate = rate / (12 * 100);
        return amount * (monthlyRate/
            (1 - Math.pow(1 + monthlyRate, -months)));
    }

}
```

Development Tools

Developer Console

Force.com IDE (Eclipse plugin)

Cloud9

MavensMate (Sublime/Atom/other plugin)

The Welkin Suite

JetBrains (IntelliJ plugin)

Force CLI

…and more

Developer Console

Browser Based IDE

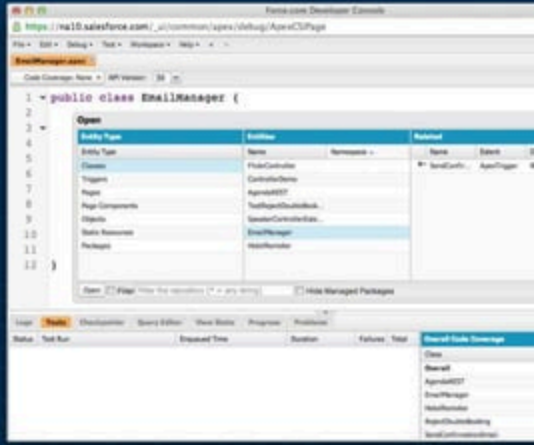
Create Classes, Triggers, Pages

Execute Apex Anonymously

Execute SOQL Queries

Run Unit Tests

Review Debug Logs



Lab 2: Creating an Apex Class

Create the EmailManager class

Send emails from the developer console

Accessing Data with SOQL and DML

What's SOQL?

Salesforce Object Query language

Similar to SQL

Streamlined syntax to traverse object relationships

Built into Apex

Read Only (We have different syntax to change data)

Basic Select Statement

```
SELECT Id, Name, Phone
    FROM Contact
```

Filters

```
SELECT Id, Name, Phone
    FROM Contact
    WHERE MailingCountry = 'Spain'
```

Boolean Operators

```
SELECT Id, Name, Phone
    FROM Contact
    WHERE MailingCountry = 'Spain'
    AND Name LIKE '%rosa%'
```

Sort Order

```
SELECT Id, Name, Phone
    FROM Contact
    WHERE MailingCountry = 'Spain'
    AND Name LIKE '%rosa%'
    ORDER BY Name
```

Limit Rows Returned

```
SELECT Id, Name, Phone
    FROM Contact
    WHERE MailingCountry = 'Spain'
    AND Name LIKE '%rosa%'
    ORDER BY Name
    LIMIT 50
```

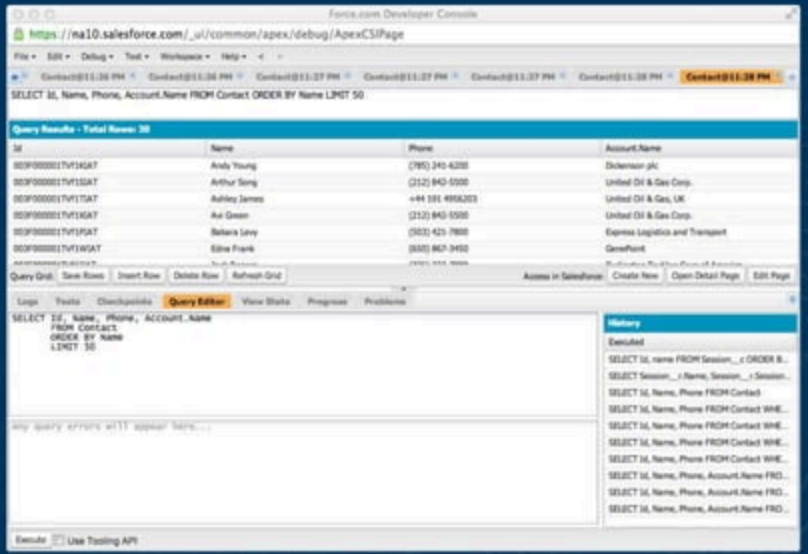
Query Including Parent Data

```
SELECT Id, Name, Phone, Account.Name
    FROM Contact
    WHERE MailingCountry = 'Spain'
    AND Name LIKE '%rosa%'
    ORDER BY Name
    LIMIT 50
```

Query Including Child Data

```
SELECT Id, Name, Phone, Account.Name
      (SELECT FirstName, LastName, Phone
            FROM Contacts)
      FROM Account
            ...
```

Demo: Executing SOQL in Developer Console



Inline SOQL in Apex: Handling Result

```
List<Session__c> sessions = [SELECT Name, Type__c
                             FROM Session__c];
```

Type check on compile

Returns List, sObject, or Integer

```
Integer i = [SELECT Count() FROM Session__c];
```

Inline SOQL in Apex: Filter Variable

```
String level = 'Advanced';

List<Session__c> sessions = [SELECT Name, Level__c
                             FROM Session__c
                             WHERE Level__c = :level];
```

Inline SOQL in Apex: List Filter Variable

```
List<String> levels = new List<String>();
levels.add('Intermediate');
levels.add('Advanced');

List<Session__c> sessions = [SELECT Name, Level__c
                             FROM Session__c
                             WHERE Level__c IN :levels];
```

Inline SOQL in Apex: As Iterator

```
for (Speaker__c s : [select email__c from Speaker__c])
    {
  System.debug(s.email__c);
   //...or maybe do something meaningful...
    }
```

Changing Data: DML

Data Manipulation Language

Syntax to create, update, delete records

Keyword syntax:

```
insert myRecord;
```

Static method of Database class:

```
Database.insert(myRecord);
```

insert

```
Session__c session = new Session__c();

session.name = 'Apex 101';
session.level__c = 'Beginner';

insert session;
```

insert

```
Session__c session = new Session__c(
    name = 'Apex 201',
    level__c = 'Intermediate'
);
insert session;
```

update

```
String oldName = 'Apex 101';
String newName = 'Apex for Beginners';

Session__c session = [SELECT Id, Name FROM Session__c
                      WHERE Name=:oldName];
session.name = newName;
update session;
```

delete

```
String name = 'Testing 501';

Session__c session = [SELECT Name FROM Session__c
                                        WHERE Name=:name];
delete session;
```

Lab 3: Accessing Data using SOQL and DML

Execute SOQL statements in the Query Editor

Execute DML statements in the Anonymous Window

What Else Should I Know?

- SOQL and DML Governor Limits
- Aggregate SOQL queries
- Dynamic SOQL using Database.query
- QueryLocator for large query results
- Full text search with SOSL
- Other DML: Upsert, Undelete, SavePoint/Rollback, etc.

Writing Triggers

What's a Trigger?

Apex code executed on database events

Before or after:

Insert

Update

Delete

Undelete

Before or After?

Before

Update fields on this record en route to database

Example: set default/calculated values

After

Create related records, access DB-generated values, async callouts

Example: Send speaker confirmation email

Think Bulkification

Trigger API is designed for batch operations

Data Import, Bulk API, REST composite/tree endpoint, etc.

Triggers work on collections of records, not single records

Context variables provide access to data:

Trigger.old and Trigger.new (List<SObject>)

Trigger.oldMap and Trigger.newMap (Map<Id,SObject>)

Example 1: DML and Loops

```
trigger WelcomeKit on Account (after insert) {
  List<Case> myCases = new List<Case>();

  for (Account account : Trigger.new) {
    Case welcomeCase = new Case();
    welcomeCase.Subject = 'Mail Welcome Kit';
    welcomeCase.AccountId = account.Id;
    myCases.add(welcomeCase);
  }
  insert myCases;
}
```

List Iterator

DML Outside Loop

Example 2: Check for Update

```
Trigger on Account (before update) {
    for (Account acc: Trigger.New) {
        // Compare new value with old value
        if (acc.Rating != Trigger.oldMap.get(acc.Id).Rating) {
            // Your Logic
        }
    }
}
```

Remember This?

Declarative vs Trigger

	Process Builder/Workflow	Trigger
Created with	Clicks	Code
What can it do	• Update field • Send email • Create records • Post to Chatter • Launch flow (flow trigger)	~ Anything (e.g. create/delete records, REST callout, etc.)
Cross-object field updates	Limited (detail -> master)	Any

Lab 4: Writing Triggers

Write the SendConfirmationEmail trigger

Write the RejectDoubleBooking trigger

What Else Should I Know?

- Order of Execution
- Batch Size Limits for Trigger Execution
- Transaction Persistent Static Variables
- Recursive Trigger Control
- Asynchronous Apex
- Trigger Patterns
- Apex Tests

thank you


salesforce


Welcome
Toulouse Salesforce
Developer Group
salesforce
Ligthning Web Component

Batch Apex

Dave Helgerson

davehelgerson.com

@davehelgerson

Relax

Effortlessly manage your Batch and Scheduled Apex processes

Zach McElrath, Skuid, Lead Developer

@zachelrath

salesforce

SELL. SERVICE. MARKET. SUCCEED.

TRAILHEADX

SALESFORCE DEVELOPER CONFERENCE

Exploring the Salesforce REST API

Jay Hurst

Director, Product Management

@ejhurst

Building Strong Foundations

Apex Enterprise Patterns

Andrew Fawcett, FinancialForce.com, CTO
@andyinthecloud

SELL. SERVICE. MARKET. SUCCEED.


salesforce
Trailblazer
COMMUNITY GROUPS
Introducing Lightning
Web Components
Roy Gilad,
Founder, SND Solutions

Salesforce Customization Vs Configuration

cloud.analogy
info@cloudanalogy.com
+1(415)830-3899

From Sandbox To Production

An Introduction To Salesforce Release Management

Seth Tager, Salesforce, Lead Member of Technical Staff
John Vogt, Salesforce, Senior Product Manager

SELL. SERVICE. MARKET. SUCCEED.