Apex Cheat Sheet

by bharatswati via cheatography.com/125832/cs/36633/

Apex Application

Create Web Service by integrating other systems
Create email services for an email blast or Email Setup
Perform Complex Validation over multiple objects at the same time and also custom validation implementation.
Create complex business processes that are not supported by existing workflow functionality or Flows
Create custom transactional logic
Perform some Logic when a record is modified.

Working Structure of Apex
Flow of Action
Developer Action
End User Action

Understanding the Apex Syntax

Variable Decalation
SOQL Query
Loop Statement
Flow Control Statement
DML Statement

Apex

What is Apex

Apex is a strongly typed, object-oriented Programming language that allows developers to execute the flow and transaction control statements on the force.com platform server in conjunction with calls to the Froce.com API.

Features of Apex as a language
Integrated
Java-like syntax and easy to use
Strongly Integrated with Data
Strongly Typed
Multitenant Environent
Upgrades Automatically

More String Examples

String Examples

public Boolean equalsIgnoreCase(String stringtocompare)

String myString1 = 'MySTRING';
String myString2 = 'MyString';
Boolean result = myString2.equalsIgnoreCase(myString1);
System.debug('Value of equal will be true as they are same and result is : ' +result);

Apex - String

String in Apex: is any set of character with no character limit.
String companyName = 'ABC';

String Methods
Syntax
public Boolean contains(String substring)

Example
String myProductName1 = 'HCL' ;
String myProductName2 = 'NAHCL' ;
Boolean result = myProductName2.contains(myProductName1);
System.debug('O/P will be true as it contains the string and Output is :' +result) ;

Apex - Variables

Declaring Variables
String StrName = 'My String';
Integer myInterger = 1;
Boolean mtBoolean = true;

Apex Variables are case-Insensitive(you cannot declare a variable two times)
Integer m = 100;
For (Integer I = 0; I < 0; I++){
integer m = 1;
Scope of Variables
List<string> Products = new List<strings>();

Understanding the Data Types

Apex Supports the Followings data types
Primitive(Integer, Double, Long, Date, Datetime, String, ID, or Boolean)
Collections(Lists, Sets and Maps)
sObject
Enums
Classes, Objects and Interfaces

Apex - Enviroment

Apex Code Development Tools
Force.com Developer Console
Froce.com IDE
Code Editor in the Salesforce use Interface

Apex - Arrays

Syntax
<String> [ ] arrayOfProducts = new List<String>();
Ex:
String [ ] arrayofproduts = new List <String>();
//Adding elements in arrays
arrayOfProducts.add('HCL');
arrrayOfProducts.add('H2SO4');

For (Integer i=0; I<arrayOfProducts.size( ); i++){
System.debug('Value In Array :' +arrayOfProducts[I]);}

Accessing array element by using index

Apex - Constants

Constants are variables which donot change their values
Ex:

public class CustomerOperationClass{
Static final Double regularCutomerDiscount = 0.1;

Apex - Loops

Loop Type & Description
For loop : This loop performs a set of statements for each item In a set of records.
SOQL for Loop: Execute a sequence of statements directly over the returned set of SOQL query.
Java-Like for Loop : Execute a sequence of statements in traditional Java-like syntax

Conti...

While loop: Repeats a statement or group of statements while a given condition is true. it tests the condition before executing the loop body.
do...while loop
Like a while statement, except that it tests the condition at the end of the loop body.

Apex - Collections

Lists
Ex:
List<String> ListOfCities = new List<String>();
System.debug('Value of ListOfCities :' +ListOfCities);

List <account> AccountToDelete = new List<account> ();
System.debug('Value Of ListOfCities' +AccountToDelete);

There is a limitation on heap size.

Methods For Lists:
size()
add()
get()
clear()
set()

Sets and Method for Sets

Pliblic Boolean equalsIgnore-Case(String stringtocompate)
String MyString1 = 'MySTRING';
String MyString2 = 'MyString';
Boolean result = myString2.eq-ualsIgonoreCase(myString1);
Sytem.debug(' your message :' +result);

Products.add(HCL);
}

Primitive Data Types

Interger
Interger X = 1000;
System.debug(' value of x Variables: ' +x);

Boolean
Boolean x;
x = true;
System.debug(' ' +x);

Date
Date x = date,today();

Long/String
Long x = 231321265465444-5643;

Object
Account x = new Account (Name ='Test AC');

Static Double finalPrice = 0;
public static Double provideDi-scount (Integer price){
// calculate the Discount
finalPrice = price - price * rebularCustomerDiscount;
return finalPrice;
}
}
OUTPUT
Double finalPrice = CustomerO-perationClass.provideDiscount(-100);
System.debug('FinalPrice :' +finalPrice);

Apex - Decision Making

Decision-making structures require that the programmer specify one or more conditions to be evaluated.

Set<string> ProductSet = new Set<string>{'Phenol', 'Benzene', 'H2SO4'};
ProdcctSet.add('HCL');
System.debug('Set with New Value :' +ProductSet);

ProductSet.remove('HCL');
ProductSet.contains('HCL');

Maps

Map<string, string> ProductCo-deToProductName = New Map<string, string> {'1000' =>'HCL', '1001' => 'H2SO4'};