

Salesforce Apex Data Types – Apex Fundamentals

What are Apex Data types?

Apex Data types tells about what type of data (such as sObject, primitive or enum) can be stored and what is the range of the data that can be stored.

Salesforce Apex is the world's first on demand strongly typed programming language developed by Salesforce.com. Apex is an object oriented language similar to C# and [Java](#) that allows to implement complex business requirements and transactions on the force.com platform. In this Salesforce tutorial, we will learn about various Salesforce apex basics like primitive data types, Collections and Enums.

Apex Data types

These are the data types which are predefined by the Apex. Salesforce supports the following Apex data types :

1. Primitive data type.
2. Collections.
3. sObjects.
4. Enums.

Primitive data type.

Primitive data type includes Integer, Double, Long, Date, Date Time, String, ID and Boolean.

- All primitive data types are passed by value, not by reference.
- All Apex variables, whether they are class member variable, are initialized to null. Make sure that we initialize variables to appropriate values before using them.

Boolean

A value that can only be assigned True, False or Null.

Eg :- Boolean isActive="False".

Date

A value that indicates a particular day, date values contain no information about time. Date value must always be created with a system static method.

Eg:- Date myDate=Date.newInstance(2017,09,15).

output : 2017-09-15 00:00:00.

Time and DateTime

These are date types associated with dates and times along with date data type. The time data types stores times(Hours, minutes, seconds and milliseconds). The date data type stores dates(Years, month and day).

Integer, Long, Double and Decimal

To store numeric values in variables, declare variables with one of the numeric data types Integer, Long, Double and Decimal.

Integer

A 32-bit numbers that doesn't include a decimal point. Integer have a minimum value of -2,147,483,648 and a maximum of 2,147,483,648.

Example : Integer i=1;

Object

Object is a instance of the class. This has both state and behaviour. Memory for the data members are allocated only when we create an object.

Syntax

```
Classname  
Objectname = new  
Classname Objectname = new Classname();
```

- Classname – This is the name of the class for which we are creating an object.
- Objectname – It's a reference variable.
- new – It is a keyword which we are allocating the memory.
- Classname() – Constructor.

What is a Class?

A class is a collection of data members and methods.

```
class Student
```

```
class Student
{
Integer no;
String name;
public void getDetails()
{
    System.debug('rollno'+no);
    System.debug('name'+name);
}
}
```

How to define an Apex class ?

To define an Apex class, we must use Access modifiers for the top level class(public or global), optional modifiers such as virtual, abstract, the keywords class followed by class name and optional extensions like AND/OR.

Salesforce Apex Class Syntax:

```
private/public/global[vir
```

```
private/public/global[virtual/abstract/With Sharing/(none)]

class classname [implements InterfaceNameList / (none)][extends classname/(none)]

{

// the body of the class

}
```

Salesforce Apex

‡ What is Salesforce Apex?

Apex Basics

‡ How to Enable Developing Mode in Salesforce?

‡ How to use Salesforce developer Console.

‡ Apex Data types.

‡ Apex - Variables

‡ Apex - Class

‡ Apex - Methods

‡ Apex - Constructors

‡ Apex - Scheduler

‡ What is Salesfore Batch Apex

‡ Batch Apex - Governor Limits

‡ Triggers in Salesforce

‡ Email Messages - Inbound, Outbound

‡ Apex - Interface

‡ Apex - DML statements