

Salesforce Developer

Apex programming O & A

180

Apex what is Apex?

→ i) Apex is object oriented programming used in Salesforce.

ii) Strongly typed as it validates references to objects at compile time.

iii) Integrated with the database.

iv) Enables developers to add business logics.

v) We can call apex through webservice requests and triggers on objects.

181

Data types in Apex. What are they?

→ i) primitive → number, string, Boolean, Integer, Double, String, Long, Date, Time, Boolean and more.

ii) Object → primitive plus objects.

↳ primitive as an generic Object or as a specific object like Account.

iii) Collection → list, set, map.

iv) Type list of values, also known as enum, string, and so on.

v) User-defined Apex classes.

vi) System Standard Apex classes.

182

Collections in Apex?

→ we have three types of collection in Apex:

i) List

ii) Set

iii) Map

### 133) List in Apex

- ① An ordered collection of elements.
- ② Each element of list has an index (index for identification).
- ③ Index position of first element is always 0.
- ④ List can be nested and even multidimensional.
- ⑤ Elements can be of any datatype:
  - ① Primitive types, collections, objects.
  - ② User-Defined types, Built-in Types.

### 134) List Examples in Apex

- ① List<String> myList = new List<String>();
- ② List<Account> accList = new List<Account>();
- ③ List<Integer> myList = new List<Integer>();  
myList.add(20);  
myList.add(1, 30);  
myList.get(0);
- ④ List<Account> accList = [SELECT ID,  
Name FROM Account];

### 135) Set in Apex

- ① An unordered collection of elements.
- ② It doesn't contain duplicate elements.
- ③ Sets can contain Collection that can be nested within one another.
- ④ Elements can be of any datatype:
  - ① Primitive types, collections, objects.
  - ② User-Defined types, Built-in Types.

138)

## Set Example in Apex ?

① Set<String> stringSet = new Set<String>();

→ ② Set<Id> accIdSet = new Set<Id>();

③ Set<Integer> intSet = new Set<Integer>();

intSet.add(10);

intSet.add(30);

intSet.remove(20);

Boolean b = intSet.contains(30);

Integer s = intSet.size();

Boolean b1 = intSet.isEmpty();

137)

## Map in Apex ?

① A map is a collection of key-value pairing.

② Keys are always unique & having a value associated.

③ No/Value can't be duplicates.

④ Adding a map entry with an existing key overrides the existing entry with new.

⑤ Map key and values can contain any collection and can contain nested collections.

⑥ Keys and values can be of any data type including primitive types, collections, objects, User-Defined Types, Built-in Types.

→ ⑦ primitive types, collections, objects, User-Defined Types, Built-in Types.

138)

## Map Examples in Apex

→ ① Map<String, String> strToStrMap = new Map<String, String>();

- ② Map< Integer, String > intToStrMap = new  
 Map< Integer, String >();  
 ① intToStrMap.put(1, 'First');  
 ② intToStrMap.put(2, 'Second');  
 ③ Boolean b = intToStrMap.containsKey(1);  
 ④ String value = intToStrMap.get(2);  
 ⑤ Set< Integer > s = intToStrMap.keySet();  
 ⑥ Map< ID, Account > idToAccountMap = new  
 Map< ID, Account >();  
 ⑦ List< Account > accList = [SELECT ID, Name  
 FROM Account];  
 ⑧ Map< ID, Account > idToAccountMap = new  
 Map< ID, Account >(accList);  
 ⑨ Map< Account, List< Contact > > accToContMap =  
 new Map< Account, List< Contact > >();

139) 02 for loop and 1 for each loop?

for loop, 2nd for loop, 1st for loop?  
 → ① for (Integer i=0; i<5; i++) {  
     // write some code here  
 for each

List< Account > accList = [SELECT ID, FROM  
 Account LIMIT 5];

for (Account acc : accList) {  
 // write some code here

// write some code here

{  
 // write some code here

    // write some code here  
 }  
 // write some code here

- 140) what is SOQL ?
- (i) Salesforce Object Query Language
  - (ii) Read Records from Salesforce
  - (iii) Similar to the standard SQL but is customized for the lightning platform.
  - (iv) SOQL can be embedded in Apex code.

Example → `list<Account> acList = [SELECT id FROM Account Limit 5];`

- 141) what is SOSL ?
- (i) Salesforce Object Search Language
  - (ii) used to perform text searching records
  - (iii) Use SOSL to search fields across multiple objects.
  - (iv) Use SOQL to retrieve records for a single object whereas use SOSL to search fields across multiple objects.

Syntax: `Find 'SearchQuery' [In SearchGroup] [RETURNING ObjectsAndFields];`

~~does not~~

- 142) what is Developer Console?
- (i) Developer Console is an integrated development environment (IDE)
  - (ii) Here you can create and edit Apex classes, Triggers, Aura Components, VF, pages etc.
  - (iii) Generate logs for debugging and analyze them.

(iv) Test apex code to ensure that it is error free.

(v) Write and execute SOQL and SOSL queries to find, create and update the records in the org.

Q3) What is Query Editor in Developer Console? How does it help to execute SOQL and SOSL?

Q4) What is Execute Anonymous Window?

i) Helps to execute apex code without creating many classes.

ii) You can execute all lines of code at once or together or select particular lines for execution as well.

iii) You also get an option to open logs just after completion of execution.

iv) You cannot save code through Anonymous window.

Q5) What is System.debug() method?

i) Display results in the debug log.

ii) Use System.debug() in the code for debugging purpose.

iii) Once your code is tested and error free then remove all the System.debug() as it is a bad practice.

Example →

```
System.debug('Test Debug');
```

146) what is DML?

- ① DML is used to insert, update & delete & undelete records.
- ② Used upsert to either insert or update a record.
- ③ Use merge when duplicate leads, contacts and accounts are there in to one record, others are deleted and related record are re-parented.
- ④ Always perform DML in bulk.

147) what is Governor's limit?

- ① An apex running in a multitenant environment. So Apex runtime engine strictly enforces limits.
- ② This is because whenever Apex code runs it consumes shared resources.
- ③ if Some apex code exceeds a limit then associated governor issuing a running exception that connot be handled.
- ④ Three limits counts for each Apex transaction.
- ⑤ For Batch Apex, these limit are granted for each execution of a batch without records in the execute method.

if (updateTest == true) update master;

(#)	Description	Limit	Hours Limit
①	Total number of SOQL queries issued	100	200
②	Total number of record entities created by SOQL queries	50,000	50,000
③	Total number of SOSL queries issued	2000	20
④	Total number of records retrieved by a single SOSL query	2000	2000
⑤	Total number of DM2 statements issued	150	150
⑥	Total number of stack depth for any Apex invocation that's recursively triggered by trigger code to insert, update or delete statement	16	16
⑦	Total number of callsouts (HTTP requests or web service calls) in a transaction	100	100
⑧	Maximum cumulative timeout for all callouts (HTTP requests or web service calls) in a transaction	120 second	120 second
⑨	Maximum number of methods with the future annotation in a class per apex invocation	50	50 in batch and future context; 1 in queueable context

(10)	Maximum number of Apex Jobs added to the queue with Transaction System	50	11:
(11)	Total heap size maximum	6 MB	12 MB
(12)	maximum CPU time on the Salesforce Servers to complete	10000 milliseconds	60000 milliseconds
(13)	maximum execution time for each Apex transaction	10 seconds minutes	10 minutes
(14)	Total number of sendEmail method allowed to return	250000	1000000

- Ques) what is Database package?
- (1) Apex contains static bulkified Database class. (figures limit 100000)
  - (2) Database package provides methods which can perform DML operation.
  - (3) Database package methods are static and called through the name of class.
  - (i) Database.insert(); to add new entry
  - (ii) Database.update(); update
  - (iii) Database.upsert(); update
  - (iv) Database.delete(); delete
  - (v) Database.delete(); minimum
  - (vi) Database.insert(); no note

- Ques) more on Database package;
- (1) Database.insert() method have an option for opportunity parameter. This parameter allows you to specify

whether the operation should partially succeed or fail.

- (3) When this parameter is set to false, if errors occur on a majority (50%) of records, then the successful records will be committed and errors will be returned for the failed records.

- (4) No exceptions are thrown with this partial success option.

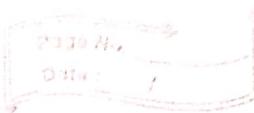
- (5) This feature is not available with DML statements.

- (6) The Database methods returning DPL results objects containing success or failure information for each record.

Database.SuccessResults[] = Database.insert(RecordList, false);

Q1: What's Database class?

- (1) Use DML statements if you want to throw error through exceptions and handle them with try & catch block.
- (2) Use Database methods if you want to allow partial success of a bulk operation. In this way, successful records will be committed and errors will be returned for the failed records.
- (3) Database methods can also throw exception similar to DML statements.



(152)

What is Exception & its condition.

① Exception is runtime error.

Example: i) Dml Statement Exceptions

② If a DML operation fails, it returns an exception of type DMLExceptions.

③ We can catch exceptions in our code to handle errors condition.

try {

// This causes an exception as Name field ( required ) is not provided

Account acc = new Account();

insert acc;

Catch(DMLException e) {

System.debug('A DML exception has occurred: '+e.getMessage());

{

(153)

Parent (→ child relationship) query

① Query is applied on parent record with query restricted child records of former parent relationship.

Example: To get all contacts under account

② SELECT id, name, (SELECT id, first\_name  
FROM Contacts) FROM Account LIMIT 5

↳ Go to Account → Contact tab → Contacts

③ If relationship field is custom field with which two objects are linked, then we have to append it as well with child relationship name option.

④ If relationship field is custom field with which two objects are linked, then we have to append it as well with child relationship name option.



child - parent SOQL ?

- (15) → ① Query is applied on child object which queries parent details as well

② Example

```
SELECT Id, FirstName, Account.Name,  
Employee__r.Salary__c FROM Contact  
LIMIT 5
```

- ③ Now, Account is representing standard relationship whereas Employee is representing custom relationship. So is custom relationship applied to

(15) Debug Log ?

- ① A debug log can record operations, system processes, and errors that occurs when executing a transaction non-running unit tests. Debug logs can contain information about:

- ① Database changes
- ② HTTP callouts
- ③ Apex Errors
- ④ Resources used by apex
- ⑤ Automated workflow processing.