Enter Apex Code ▴ X

```
1    batch apex class - async apex
2
3    hit governor limits
4    1cr record, no of batch.
5
6    each batch will get a frsh set of governor limits
7
8    1000 records - it has a fresh set of governor limit    I
9    2nd 1000 records (1001 to 2000) -
10
```

☐ Open Log   Execute   Execute Highlighted

File ▾   Edit ▾   Debug ▾   Test ▾   Workspace ▾   Help ▾   ‹   ›

myfirstclass.apxc ✕   **ApplicationBatchTest.apxc** ✕

Code Coverage: None ▾   API Version   53   ✕

```
 1  ▾  public class ApplicationBatchTest implements Database.Batchable<sobject> {
 2
 3         //start(), execute(). finish()
 4         public Integer totalFroms = 0;
 5         public Integer totalConverted = 0;
 6  ▾      public Database.QueryLocator start(Database.BatchableContext bc){
 7             // gather the data for you
 8             String applicationQuery = 'select id, name, Ready_to_Join__c from Application_Form__c';
 9             return Database.getQueryLocator(applicationQuery);
10         }
11  ▾      public void execute(Database.BatchableContext bc, List<Application_Form__c> formList){
12             // process the data
13             for(Application_Form__c af : formList){
14             }
15         }
16  ▾      public void finish(Database.BatchableContext bc){
17             // emails ,                                                    |
18         }
19      }
20  }
```

Logs   Tests   Checkpoints   Query Editor   View State   Progress   Problems

User

Application   Operation   Time ▾   Status   Read

User

implements Database.Batchable<sobject> {

}

}

~~~~/~~~~~~ ~~~~~~~~~ ~~~~~~~~~ ~~~~

Apex Code   ▲ X

// application form.
at the end of business hr, they basically want to trigger an email which will
no of total application form, no of application got converted to student.

⬚ Open Log   Execute   Execute Highlighted

Edit MIT-CCU

Owner
⬤ Bommineni Pravalika

*College Name
MIT-CCU ▾
View all dependencies

*college fee
$1,100,000.00 ▾

College Email
mit@ccu.com ▾
View all dependencies

Hostel Fee
$150,000.00

No of Seats
5000-8000 ▾
View all dependencies

*Record Info
MIT-CCU

Last Modified By
⬤ Bommineni Pravalika, 1/11/2022, 3:24 AM

Created By
⬤ Bommineni Pravalika, 1/11/2022, 3:24 AM

Cancel   Save & New   Save

```
1  // database.executeBatch(instance-of batch class, size of batch);
2
3  |
```

Enter Apex Code

```
1  // Database.executeBatch(instance-of batch class, size of batch);
2
3  ApplicationBatchTest abt = new ApplicationBatchTest();
4  Database.executeBatch(abt, 400); // 200 to 2000
```

Open Log   Execute   Execute Highlighted

```
1 ▾ public class ApplicationSchedule Implements Schedulable{
2
3 ▾     public void execute(SchedulableContext  sc){
4           ApplicationBatchTest abt = new ApplicationBatchTest();
5           Database.executeBatch(abt,400); //200 to 2000
6       }
7   }
```

⚙ Apex Classes

Apex Classes

Apex Code is an object oriented programming language that allows developers to develop on-demand business applications on the Lightning Platform

> **Percent of Apex Used: 0.05%**
> You are currently using 3,061 characters of Apex Code (excluding comments and @isTest annotated classes) in your organization, out of an allowed limit of 6,000,000 characters. Note that the amount i
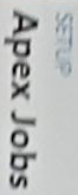> Apex Classes and Triggers defined in your organization.

Estimate your organization's code coverage ⓘ

Compile all classes ⓘ

View: All ▾ Create New View

A B C D E F G H I J K L M N O P Q R S T U V

| | | Developer Console | New | Generate from WSDL | Run All Tests | Schedule Apex |

Action	Name ↑	Namespace Prefix	Api Version	Status	Size Without Comments	Last Modified By
Edit	Del	Security	ApplicationBatchTest		53.0	Active	622	Bommineni Pravalika 1/13/2022, 2:32 AM
Edit	Del	Security	ApplicationSchedule		53.0	Active	221	Bommineni Pravalika 1/13/2022, 3:08 AM
Edit	Del	Security	College_c		53.0	Active	27	Bommineni Pravalika 1/13/2022, 3:08 AM
Edit	Security	🔺 DashboardPalCtrl	Dashboard_Pal	41.0	Active		Bommineni Pravalika 1/12/2022, 4:23 AM
Edit	🔺 DashboardPalTest	Dashboard_Pal	41.0	Active	3,252	Bommineni Pravalika 10/28/2021, 11:15 PM
Edit	Del	Security	database		41.0	Active	1,206	Bommineni Pravalika 10/28/2021, 11:15 PM
Edit	Del	Security	datatable		53.0	Active	26	Bommineni Pravalika 1/17/2022, 6:47 AM
Edit	Del	Security	formhelper1		53.0	Active	27	Bommineni Pravalika 1/17/2022, 6:50 AM
Edit	Del	formhelpertest		53.0	Active	973	Bommineni Pravalika 1/13/2022, 7:13 AM
Edit	Del			53.0	Active	690	Bommineni Pravalika 1/13/2022, 7:42 AM
			53.0	Active	33	Bommineni Pravalika

Click here to go to the new batch jobs page

Apex Jobs

Monitor the status of all Apex jobs, and optionally, abort jobs that are in progress.

View: All ▾ Create New View

Action	Submitted Date ↑	Job Type	Status	Status Detail	Total Batches	Batches Processed	Failures	Submitted By	Completion Date	Apex Class	Apex Method	Apex Job ID
	1/21/2022 5:23 AM	Future	Completed		0	0	0	Pravalika Remmanana	1/21/2022 5:23 AM	mrdrrcnm		brf
	1/16/2022 11:10 PM	Batch Apex	Completed		1	1	0	Pravalika Remmanana	1/16/2022 11:10 PM	Annbatchmethod.Test		Pnmpnmbplnf
	1/16/2022 7:59 AM	Future	Completed		0	0	0	Pravalika Remmanana	1/16/2022 7:59 AM	mrdrrcnm		brf
	1/13/2022 3:32 AM	Scheduled Apex	Queued		0	0	0	Pravalika Remmanana		AvabcabbbSchedb		ktnwpnwmhp

Apex Flex Queue

Monitor the status of Apex jobs in the Apex flex queue, and reorder them to control which jobs are processed first

View: All ▾ Create New View

Position	Submitted Date	Job Type	Submitted By	Apex Class	Apex Method

No records to display.

Object Manager ∨

SETUP

Scheduled Jobs

All Scheduled Jobs

The All Scheduled Jobs page lists all of the jobs scheduled by your users. Multiple job types may display on this page. You can delete scheduled jobs if you have the permission to do so.

View: All Scheduled Jobs ∨   Create New View

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Other All

Action	Job Name ↑	Submitted By	Submitted	Started	Next Scheduled Run	Type
Del	cronJob1	Pravalika_Bommaneni	1/13/2022, 3:32 AM	1/16/2022, 11:10 PM	1/23/2022, 11:10 PM	Scheduled Apex
Del	Metalytics Data Loader Job for Org - 00D5j000001KY6C	User_Integration	10/26/2021, 10:41 PM	1/20/2022, 6:56 PM	1/21/2022, 6:56 PM	Autonomous Data Loader Job

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Other All

File ▾   Edit ▾   Debug ▾   Test ▾   Workspace ▾   Help ▾   < >

myFirstclass.apxc ×   ApplicationBatchTest.apxc ×   ApplicationSchedule.apxc ×   **ApplicationFormTrigger.apxt** ×

Code Coverage: None ▾   API Version   53 ˅

```
1 ▾  trigger ApplicationFormTrigger on Application_Form__c (before delete) {
2
3        // context variables : trigger.new, trigger.old, trigger.newMap, trigger.oldMap
4 ▾      /* object - Application_Form__c
5        trigger.new, trigger.old - list of Application_Form__c
6        trigger.new or trigger.old = List<Application_Form__c>
7        trigger.newMap, trigger.oldMap = Map<id, Application_Form__c>
8
9        when you do a before insert, you dont have trigger.old .
10       when you do after insert operation, both trigger.old and trigger.new is same.
11
12       in edit operation, your trigger.old is different then trigger .new
13
14
15       we will be able to get our previous values. */
16
17
18 ▾    if(Trigger.isUpdate && Trigger.isBefore){
19
20       formHelper1.getoldNewValues(trigger.oldMap, trigger.new);
21
22       }
23
24 ▾   if(Trigger isDelete && Trigger isBefore){
```

Logs   Tests   Checkpoints   **Query Editor**   View State   Progress   Problems

Guardian_Name__c, Date_of_Birth__c from Application_Form__c

```
1    • public class formhelper1 {
2
3    •    public static void getoldNewValues(Map<id, Application_Form__c> formoldMap, List<Application_Form__c
4    •        for(Application_Form__c fm : formlist){
5
6
7            Application_Form__c oldFormRecord = formoldMap.get(fm.id);
8            System.debug(' old value : '+oldFormRecord.Student_Name__c);
9            System.debug(' new values :'+ fm.Student_Name__c+ 'old values : '+formoldMap.get(fm.id).Student_N
10
11   •        if(fm.Student_Name__c != formoldMap.get(fm.id).Student_Name__c){
12               // do some operation
13
14
15        }
16   •  public static void preventRecordDeletion(List<Application_Form__c> formlist){
17
18            Set<id> formidSet = new Set<id>();
19   •        for(Application_Form__c fm : formlist){
20               formidSet.add(fm.id);
21
22   •        if(formidSet != null && formidSet.size() > 0){
23            List<Student__c> stdlist = [select id from Student__c where Application_Form__c IN :formidSe
```