```
Developer Console - Google Chrome
orgfarm-7fd2664f10-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File - Edit - Debug - Test - Workspace - Help - < >
test.apxt * X testHandler.apxc * X MonthlyEmailScheduler.apxc * X
  Code Coverage: None -
       trigger test on Tenant__c (before insert)
  1
  2
  3
            if(trigger.isInsert && trigger.isBefore){
  5 ▼
  6
                 testHandler.preventInsert(trigger.new);
  8
  9
  10
  11
```

```
Developer Console - Google Chrome
orgfarm 7fd2664f10 dev ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File - Edit - Debug - Test - Workspace - Help - < >
test.apxt * * testHandler.apxc * * MonthlyEmailScheduler.apxc * *
 Code Coverage: None - API Version: 64 -
                                                                                                                                                                      G
 1 * public class testHandler {
 2
         public static void preventInsert(List<Tenant_c> newlist) {
 3 ▼
              Set<Id> existingPropertyIds = new Set<Id>();
 5
 6
              for (Tenant c existingTenant : [SELECT Id, Property c FROM Tenant c WHERE Property c != null]) {
 7 🕶
 8
                  existingPropertyIds.add(existingTenant.Property_c);
 9
10
              }
11
12
13
              for (Tenant_c newTenant : newlist) {
14 *
15
16
17
                  if (newTenant.Property_c != null && existingPropertyIds.contains(newTenant.Property_c)) {
18 🕶
19
                       newTenant.addError('A tenant can have only one property');
20
21
22
23
24
25
```

```
orgfarm-7fd2664f10-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
e • Edit • Debug • Test • Workspace • Help • < >
est.apxt * * testHandler.apxc * * MonthlyFmailScheduler.apxc * *
Code Coverage: None . API Version 64 .
1 *; lobal class MonthlyEmailScheduler implements Schedulable {
3 ▼
        global void execute(SchedulableContext sc) {
4
            Integer currentDay = Date.today().day();
5
6
            if (currentDay == 1) {
7 •
8
                sendMonthlyEmails();
9
10
11
12
13
14
15
        public static void sendMonthlyEmails() {
16 •
17
            List<Tenant_c> tenants = [SELECT Id, Email_c FROM Tenant_c];
18
19
            for (Tenant_c tenant : tenants) {
20 ▼
21
                 String recipientEmail = tenant.Email c;
22
23
                 String emailContent = 'I trust this email finds you well. I am writing to remind you that the monthly rent is due Your timely payment
24
```

```
String emailSubject = 'Reminder: Monthly Rent Payment Due';

Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();

email.setToAddresses(new String[]{recipientEmail});

email.setSubject(emailSubject);

email.setPlainTextBody(emailContent);

Messaging.sendEmail(new Messaging.SingleEmailMessage[]{email});

}
```