

Exception Class and Built-In Exceptio

An exception denotes an error that disrupts the normal flow of code execution. You can use , built-in exceptions or create custom exceptions. All exceptions have common methods.

All exceptions support built-in methods for returning the error message and exception type. addition to the standard exception class, there are several different types of exceptions:

The following are exceptions in the system namespace.

| Exception | Description |
|---------------------------|--|
| AssertException | A System.assert failure that halts code execution. Optionally contains the custom message specified in the last (msg) argu to the assert() method. |
| AuraException | Legacy Aura-related exception. Use System.AuraHandledExcep instead. |
| AuraHandledException | Returns a custom error message to a JavaScript controller. See Returning Errors from an Apex Server-Side Controller . |
| AsyncException | Any problem with an asynchronous operation, such as failing t enqueue an asynchronous call. |
| BigObjectException | Any problem with big object records, such as connection time during attempts to access or insert big object records. |
| CalloutException | Any problem with a Web service operation, such as failing to r a callout to an external system. |
| DataWeaveScriptException | Any run-time script errors that occur within DataWeave in Ape |
| DmlException | Any problem with a DML statement, such as an insert stater missing a required field on a record. |
| DuplicateMessageException | Attempt to enqueue job with duplicate queueable signature |
| EmailException | Any problem with email, such as failure to deliver. For more information, see Outbound Email . |
| ExternalObjectException | Any problem with external object records, such as connection timeouts during attempts to access the data that's stored on external systems. |
| FatalCursorException | Any problem with Apex cursors in a transaction. |
| FinalException | Any attempt to mutate a read-only collection or record such a sObject in an after-update trigger, or a final variable. This exce causes execution to halt. |
| FlowException | Any problem with starting flow interviews from Apex. For exar if an active version of the flow can't be found or it can't be sta from Apex. |



| | |
|--------------------------------|--|
| IllegalArgumentException | An illegal argument was provided to a method call. For example, a method that requires a non-null argument throws this exception if a null value is passed into the method. |
| InvalidHeaderException | An illegal header argument was provided to an Apex REST call. For example, a call to the <code>RestResponse.addHeader(name, value)</code> method throws this exception if the header name is <code>cookie</code> . |
| InvalidParameterValueException | <p>This exception is used with both Visualforce pages and Salesforce Functions.</p> <p>Visualforce</p> <p>The exception is thrown when an invalid parameter is supplied for a method, or any problem is encountered with a URL used with Visualforce pages. For more information on Visualforce, see the Visualforce Developer's Guide.</p> <p>Salesforce Functions</p> <p>The exception is thrown when the <code>functionName</code> parameter to <code>Function.get()</code> doesn't have the correct project name.function name format. For more information on Salesforce functions, see Function.get().</p> |
| LimitException | A governor limit has been exceeded. This exception can't be caught. |
| JSONException | Any problem with JSON serialization and deserialization operations. For more information, see the methods of System.JSON , System.JSONParser , and System.JSONGenerator . |
| ListException | Any problem with a list, such as attempting to access an index that is out of bounds. |
| MathException | Any problem with a mathematical operation, such as dividing by zero. |
| NoAccessException | Any problem with unauthorized access, such as trying to access an sObject that the current user doesn't have access to. This exception is used with Visualforce pages. For more information on Visualforce, see the Visualforce Developer's Guide . |
| NoDataFoundException | <p>This exception is used with both Visualforce pages and Salesforce Functions.</p> <p>Visualforce</p> <p>The exception is thrown with data that doesn't exist, such as trying to access an sObject that has been deleted. For more information on Visualforce, see the Visualforce Developer's Guide.</p> <p>Salesforce Functions</p> <p>The exception is thrown when the project or function name provided in the <code>functionName</code> parameter to the <code>Function.get()</code> method can't be found. For more information on Salesforce functions, see Function.get().</p> |
| NoSuchElementException | This exception is thrown if you try to access items that are out of the bounds of a list. This exception is used by the Iterator <code>next</code> method. For example, if <code>iterator.hasNext() == false</code> and you call <code>iterator.next()</code> , this exception is thrown. This exception |



| | |
|--------------------------|--|
| NullPointerException | Any problem with dereferencing null, such as in the following code: <div><pre>String s; s.toLowerCase(); // Since s is null, this call // a NullPointerException</pre></div> |
| QueryException | Any problem with SOQL queries, such as assigning a query that returns no records or more than one record to a singleton sObject variable. |
| RequiredFeatureMissing | A Chatter feature is required for code that has been deployed to an organization that doesn't have Chatter enabled. |
| SearchException | Any problem with SOSL queries executed with SOAP API search call, for example, when the searchString parameter contains fewer than two characters. For more information, see the SOA Developer Guide . |
| SecurityException | Any problem with static methods in the Crypto utility class. For more information, see Crypto Class . |
| SerializationException | Any problem with the serialization of data. This exception is used with Visualforce pages. For more information on Visualforce, see the Visualforce Developer's Guide . |
| SObjectException | Any problem with sObject records, such as attempting to change a field in an update statement that can only be changed during insert. |
| StringException | Any problem with Strings, such as a String that is exceeding its heap size. |
| TransientCursorException | A transient problem with an Apex cursor transaction. The failed transaction can be retried. |
| TypeException | Any problem with type conversions, such as attempting to convert the String 'a' to an Integer using the valueOf method. |
| UnexpectedException | A non-recoverable internal error within Salesforce has occurred. This exception causes execution to halt. If necessary, contact Salesforce Customer Support for more information. |
| VisualforceException | Any problem with a Visualforce page. For more information on Visualforce, see the Visualforce Developer's Guide . |
| XmlException | Any problem with the XmlStream classes, such as failing to read or write XML. |

The following is an example using the DmlException exception:

```
Account[] accts = new Account[]{new Account(billingcity = 'San Jose')};  
try {  
    insert accts;  
} catch (System.DmlException e) {  
    for (Integer i = 0; i < e.getNumDml(); i++) {
```



For exceptions in other namespaces, see:

- [Cache Exceptions](#)
- [Canvas Exceptions](#)
- [Compression Exceptions](#)
- [ConnectApi Exceptions](#)
- [DataSource Exceptions](#)
- [Reports Exceptions](#)
- [Site Exceptions](#)

Common Exception Methods

Exception methods are all called by and operate on an instance of an exception. This table describes all instance exception methods. All types of exceptions have these methods in common.

| Name | Arguments | Return Type | Description |
|---------------------|------------------------|-------------|--|
| getCause | | Exception | Returns the cause of the exception as an exception object. |
| getLineNumber | | Integer | Returns the line number from where the exception was thrown. |
| getMessage | | String | Returns the error message that displays for the user. |
| getStackTraceString | | String | Returns the stack trace of a thrown exception as a string. |
| getTypeName | | String | Returns the type of exception, such as DmlException, ListException, MathException and so on. |
| initCause | Exception <i>cause</i> | Void | Sets the cause for this exception, if one has already been set. |
| setMessage | String <i>s</i> | Void | Sets the error message that displays for the user. |

DMLException and EmailException Methods

In addition to the common exception methods, DMLException and EmailException have the following methods:

| Name | Arguments | Return Type | Description |
|------------------|------------------|-------------|---|
| getDmlFieldNames | Integer <i>i</i> | String [] | Returns the names of the field or fields that caused the error described by the failed row. |



| | | | |
|------------------|-------------|-----------------------------------|---|
| | | | described by the i^{th} failed row. For more information on field tokens, see Dynamic Apex . |
| getDmlId | Integer i | String | Returns the ID of the failed record that caused the error described by the i^{th} failed row. |
| getDmlIndex | Integer i | Integer | Returns the original row position of i^{th} failed row. |
| getDmlMessage | Integer i | String | Returns the user message for the i^{th} failed row. |
| getDmlStatusCode | Integer i | String | Deprecated. Use getDmlType instead. Returns the Apex failure code for the failed row. |
| getDmlType | Integer i | System.StatusCode | Returns the value of the System.StatusCode enum. For example: <div><pre>try { insert new Account(); } catch (System.DmlException ex) { System.assertEquals(System.StatusCode.REQUIRED_FIELD_MISSING, ex.getDmlType(0)); }</pre></div> |
| getNumDml | | Integer | Returns the number of failed rows from the DML exceptions. |

QueryException Method

In addition to the common exception methods, QueryException has this method.

| Name | Arguments | Return Type | Description |
|-----------------------|-----------|---|--|
| getInaccessibleFields | | Map<String,Set<String>> | Returns a map in which each key is an sObjectType and the corresponding value is the set of inaccessible field names in the fully qualified format (Namespace__FieldName). Use this method to determine the cause of the QueryException. The returned map contains data only if the method that threw the QueryException is running in user mode (as opposed to the default system mode). |



have field-level security access to the Contact.Email and Account.Website fields.

```
try {
    List<Account>
} catch (QueryException) {
    // Handle in
    Map<String,
    Set<String>
    Set<String>
}
```

DID THIS ARTICLE SOLVE YOUR ISSUE?
Let us know so we can improve!

[Share your feedback](#)



DEVELOPER CENTERS

- [Heroku](#)
- [MuleSoft](#)
- [Tableau](#)
- [Commerce Cloud](#)
- [Lightning Design System](#)
- [Einstein](#)
- [Quip](#)

POPULAR RESOURCES

- [Documentation](#)
- [Component Library](#)
- [APIs](#)
- [Trailhead](#)
- [Sample Apps](#)
- [Podcasts](#)
- [AppExchange](#)

COMMUNITY

- [Trailblazer Connect](#)
- [Events and Webinars](#)
- [Partner Community](#)
- [Blog](#)
- [Salesforce Admins](#)
- [Salesforce Architects](#)

© Copyright 2025 Salesforce, Inc. [All rights reserved.](#) Various trademarks held by their respective owners. Salesforce, Inc. Salesforce Tower, 415 Mission Street, 3rd Floor, San Francisco, CA 94105, United States

[Privacy Information](#) [Terms of Service](#) [Legal](#) [Use of Cookies](#) [Trust](#) [Cookie Preferences](#)

[Your Privacy Choices](#) [Responsible Disclosure](#) [Contact](#)