

BackupDataOutput

public class BackupDataOutput

extends Object

(<https://developer.android.com/reference/java/lang/Object.html>)

java.lang.Object (<https://developer.android.com/reference/java/lang/Object.html>)

↳ android.app.backup.BackupDataOutput

added in API level 8

(<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

Summary: Methods (#pubmethods) | Inherited Methods (#inhmethods) | [Expand All] (#)

Provides the structured interface through which a **BackupAgent**

(<https://developer.android.com/reference/android/app/backup/BackupAgent.html>) commits information to the backup data set, via its **onBackup()**

([https://developer.android.com/reference/android/app/backup/BackupAgent.html#onBackup\(android.os.ParcelFileDescriptor, android.app.backup.BackupDataOutput, android.os.ParcelFileDescriptor\)](https://developer.android.com/reference/android/app/backup/BackupAgent.html#onBackup(android.os.ParcelFileDescriptor, android.app.backup.BackupDataOutput, android.os.ParcelFileDescriptor))) method. Data written for backup is presented as a set of "entities," key/value pairs in which each binary data record "value" is named with a string "key."

To commit a data record to the backup transport, the agent's **onBackup()**

([https://developer.android.com/reference/android/app/backup/BackupAgent.html#onBackup\(android.os.ParcelFileDescriptor, android.app.backup.BackupDataOutput, android.os.ParcelFileDescriptor\)](https://developer.android.com/reference/android/app/backup/BackupAgent.html#onBackup(android.os.ParcelFileDescriptor, android.app.backup.BackupDataOutput, android.os.ParcelFileDescriptor))) method first writes an "entity header" that supplies the key string for the record and the total size of the binary value for the record. After the header has been written, the agent then writes the binary entity value itself. The entity value can be written in multiple chunks if desired, as long as the total count of bytes written matches what was supplied to **writeEntityHeader()**

([https://developer.android.com/reference/android/app/backup/BackupDataOutput.html#writeEntityHeader\(java.lang.String, int\)](https://developer.android.com/reference/android/app/backup/BackupDataOutput.html#writeEntityHeader(java.lang.String, int))).

Entity key strings are considered to be unique within a given application's backup data set. If a backup agent writes a new entity under an existing key string, its value will replace any previous value in the transport's remote data store. You can remove a record entirely from the remote data set by writing a new entity header using the existing record's key, but supplying a negative **dataSize** parameter. When you do so, the agent does not need to call **writeEntityData(byte[], int)**

([https://developer.android.com/reference/android/app/backup/BackupDataOutput.html#writeEntityData\(byte\[\], int\)](https://developer.android.com/reference/android/app/backup/BackupDataOutput.html#writeEntityData(byte[], int))).

Example

Here is an example illustrating a way to back up the value of a String variable called **mStringToBackUp**:

```
static final String MY_STRING_KEY = "storedstring";

public void onBackup(ParcelFileDescriptor oldState, BackupDataOutput data, ParcelFileDescriptor newSta
    throws IOException {
    ...
    byte[] stringBytes = mStringToBackUp.getBytes();
    data.writeEntityHeader(MY_STRING_KEY, stringBytes.length);
    data.writeEntityData(stringBytes, stringBytes.length);
    ...
}
```

See also:

BackupAgent (<https://developer.android.com/reference/android/app/backup/BackupAgent.html>)

Summary

Public methods	
long	<code>getQuota</code> (https://developer.android.com/reference/android/app/backup/BackupDataOutput.html#getQuota())() Returns the quota in bytes for the application's current backup operation.
int	<code>writeEntityData</code> (https://developer.android.com/reference/android/app/backup/BackupDataOutput.html#writeEntityData(byte[],int)) (byte[] data, int size) Write a chunk of data under the current entity to the backup transport.
int	<code>writeEntityHeader</code> (https://developer.android.com/reference/android/app/backup/BackupDataOutput.html#writeEntityHeader(java.lang.String,int))(String (https://developer.android.com/reference/java/lang/String.html) key, int dataSize) Mark the beginning of one record in the backup data stream.

Inherited methods	
▼	(#)From class <code>java.lang.Object</code> (https://developer.android.com/reference/java/lang/Object.html)

Public methods

`getQuota` added in API level 26 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

long getQuota ()

Returns the quota in bytes for the application's current backup operation. The value can vary for each operation.

Returns	
long	

See also:

getQuota() ([https://developer.android.com/reference/android/app/backup/FullBackupDataOutput.html#getQuota\(\)](https://developer.android.com/reference/android/app/backup/FullBackupDataOutput.html#getQuota()))

writeEntityData

added in API level 8 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

```
int writeEntityData (byte[] data,
                    int size)
```

Write a chunk of data under the current entity to the backup transport.

Parameters	
data	byte : A raw data buffer to send
size	int : The number of bytes to be sent in this chunk
Returns	
int	the number of bytes written
Throws	
IOException (https://developer.android.com/reference/java/io/IOException.html)	if the write failed

writeEntityHeader

added in API level 8 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

```
int writeEntityHeader (String (https://developer.android.com/reference/java/lang/String.html) key,
                      int dataSize)
```

Mark the beginning of one record in the backup data stream. This must be called before **writeEntityData(byte[], int)** ([https://developer.android.com/reference/android/app/backup/BackupDataOutput.html#writeEntityData\(byte\[\], int\)](https://developer.android.com/reference/android/app/backup/BackupDataOutput.html#writeEntityData(byte[], int))).

Parameters	
key	String : A string key that uniquely identifies the data record within the application. Keys whose first character is <code>0</code> or higher are not valid.
dataSize	int : The size in bytes of this record's data. Passing a dataSize of -1 indicates that the record under this key should be deleted.
Returns	
int	The number of bytes written to the backup stream
Throws	
IOException (https://developer.android.com/reference/java/io/IOException.html)	if the write failed

