**PostgreSql Databases Backup Automation**

1. **Tables**
   1. **DBBackupInfo**

New Table

**This table is used for taking the details to run the backup tool. Details used are as below-**

Schema Details as below-

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Nullable** | **Usage** |
| Id | Int | No | Auto Increment |
| DatabaseServer | Text | No | Input from frontend |
| DatabaseUser | Text | No | Input from frontend |
| DatabasePassword | Text | No | Input from frontend |
| DatabaseName | Text | No | Input from frontend |
| DatabasePort | Text | No | Input from frontend |
| WantBackup | Boolean | No | It indicates whether the database for the particular row needs to be taken or not. Default value is True. |

New Table

* 1. **PostgreBackupLogs**

**This table stores all the logs of backup done by tool.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Nullable** | **Usage** |
| Id | Int | No | Auto Increment |
| DatabaseServer | Int | No |  |
| DatabaseUserName | Text | No |  |
| DatabasePassword | Text | No |  |
| DatabaseName | Text | No |  |
| DatabasePort | Text | No |  |
| DBBackupInfoId | Int | No | Parent Id of DBBackupInfo |
| NoOfBackupFiles | Text | No | No of files taken for backup |
| Category | Text | No | Category denotes Daily or Weekly.  Daily means it is as per the Daily job and Weekly means it is as per Weekly job |
| CreatedOn | DateTime | No | Log CreatedOn |

* 1. **ExceptionLogs**

**This table contains the exception logs if any.**

1. **Settings in Json file**

Provide the values in appSettings.json file of tool to proceed.

"ConnectionStrings": {

"DefaultConnection": “”,

Provide the connections string of database which contains the table DBBackupInfo table from which the records need to be read for backup.

},

"ScheduleSettings": {

"Daily": "true", -- It indicates files need to be moved to Daily folder. Make it true only if the scheduling job is running it on daily basis.

"Weekly": "true" -- It indicates files need to be moved to Weekly folder. Make it true only if the scheduling job is running it on weekly basis.

},

"GenericStorageSettings": {

"AccessKey": "", Provide the Access Key(key1) of Destination Storage Account over which the backup needs to be taken. Make sure the account is under the same subscription.

"ConnectionString": "",

Provide the Connection String(key1) of Destination Storage Account over which the backup needs to be taken. Make sure the account is under the same subscription.

"ContainerName": "Backup"

Name of container under the destination storage account over which backup needs to be taken.

},

"BackupSettings": {

"BackupPath": "C:\\pgbackup\\Postgre\\",

Path where the Backup of Databases need to be place. This should be shared drive which should be accessible for all servers from which databases need to be taken backup.

"Extension": ".backup"

Extension of backup file.

},

1. **How Tool Works?**
2. Provide all the necessary details in appSettings.json file.
3. Tool will first read the data from Database table- DBBackupInfo to fetch the data for which WantBackup is made true. Looping for all databases will start.
4. For all those records, first over the Destination Storage Account as per the json file containers will be checked whether there is container or not. If there will be no container with the name then a container will be created over the destination storage account.
5. For the container, tool will delete the all the files as per the category under which the tool is running which is Daily or Weekly. If Daily is true then all the files of particular container will be deleted and same goes for Weekly.
6. For each database of each server the backup of all the files will be done over the same container mentioned in json file over destination storage account as per the category under which the tool is running which is Daily or Weekly. If Daily is true then all the files of particular container and same goes for Weekly. This category is decided under json file.
7. Path for Backup will be like this-

Generic Storage Account-> Contianer Name(Backup) -> Server -> DBName-Daily -> Files

Generic Storage Account-> Contianer Name(Backup) -> Server -> DBName-Weekly -> Files

1. After Backup the Backup logs will be stored in PostgreBackupLogs table.