DECLARE @linkedserver as SYSNAME  
DECLARE @sql1 NVARCHAR(MAX)  
DECLARE @sql2 NVARCHAR(MAX)  
DECLARE @path VARCHAR(150)  
DECLARE @bakupfile VARCHAR(200)  
DECLARE @db VARCHAR(255)  
DECLARE @BackUpsql NVARCHAR(MAX)  
  
--Get a linked server information  
SET @linkedserver = (select Name from sys.servers where server\_id <> 0 and is\_linked = 1)  
  
-- Get back up path from database setting  
EXEC master..xp\_instance\_regread @rootkey = 'HKEY\_LOCAL\_MACHINE',    
@key = 'Software\Microsoft\MSSQLServer\MSSQLServer',    
@value\_name = 'BackupDirectory', @path = @path OUTPUT ;    
  
--Added back slash at the end of back up path  
Set @path = @path + '\'  
  
DECLARE AG\_cr CURSOR LOCAL FOR  
  
 -- Get new databases which are not part of an availability replica of in availablity group  
  
  SELECT name FROM sys.databases   
  WHERE replica\_id IS NULL   
  AND database\_id > 4 and   
  source\_database\_id is NULL   
  and State =0 and   
  name not in('distribution','ReportServer','ReportServerTempDB')  
  
  OPEN AG\_cr  
  
  FETCH AG\_cr INTO  
  @db  
  
WHILE @@FETCH\_STATUS >= 0  
        BEGIN  
  
print @db  
--Change database recovery mode to FULL  
  
  IF (SELECT recovery\_model FROM sys.databases WHERE name=@db) <> 1  
        BEGIN  
            PRINT 'Changing recovery model to FULL';  
            DECLARE @ModeChange NVARCHAR(512) = 'ALTER DATABASE [' + @db + ']  SET RECOVERY FULL WITH NO\_WAIT';  
            EXEC sp\_executesql @ModeChange;  
        END;  
    ELSE  
        BEGIN  
            PRINT 'Database is already in FULL recovery mode.'  
        END;  
  
Print 'Initial Database BackUp...'  
  SET @BackUpsql = N'';  
  
  SELECT @BackUpsql += N'BACKUP DATABASE ' + QUOTENAME(@db)+ ' TO DISK = ''' + @path + @db + '.BAK'';'  
        
    PRINT @BackUpsql;  
    EXEC master..sp\_executesql @BackUpsql;  
  
  
Print 'Backing up a database...'  
  SET @BackUpsql = N'';  
  
  SELECT @BackUpsql += N'BACKUP DATABASE ' + QUOTENAME(@db)+ ' TO DISK = ''' + @path + @db + '\_AG.BAK'' WITH  COPY\_ONLY, FORMAT, INIT, SKIP, REWIND, NOUNLOAD, COMPRESSION,  STATS = 5;'  
        
    PRINT @BackUpsql;  
    EXEC master..sp\_executesql @BackUpsql;  
  
Print 'Backing up a Log...'  
  SET @BackUpsql = N'';  
  
  SELECT @BackUpsql += N'BACKUP LOG ' + QUOTENAME(@db)+ ' TO DISK = ''' + @path + @db + '\_AGLOG.TRN'' WITH NOFORMAT, NOINIT, NOSKIP, REWIND, NOUNLOAD, COMPRESSION,  STATS = 5;'  
        
    PRINT @BackUpsql;  
    EXEC master..sp\_executesql @BackUpsql;  
  
set @bakupfile = @db +'\_AG.BAK'  
  
set @sql1 = 'EXEC ' +'['+ @linkedserver+']'+'.master.dbo.sp\_executesql N'' RESTORE DATABASE ' +@db+ ' FROM DISK ='+''''''+ @path + @bakupfile+''''''+ 'with norecovery'''  
  
print @sql1  
  
Exec sp\_executeSQL @sql1  
  
set @bakupfile = @db +'\_AGLOG.TRN'  
  
set @sql1 = 'EXEC ' +'['+ @linkedserver+']'+'.master.dbo.sp\_executesql N'' RESTORE LOG ' +@db+ ' FROM DISK ='+''''''+ @path + @bakupfile+''''''+ 'with norecovery'''  
  
print @sql1  
  
Exec sp\_executeSQL @sql1  
  
  
--join AG group  
DECLARE @AGroupName VARCHAR(200)  
SET @AGroupName= (select Name from sys.availability\_groups)  
  
  
PRINT 'Joining database to availability group...'  
  
DECLARE @AddDbToAG NVARCHAR(2048) = 'USE master' + char(13)+'ALTER AVAILABILITY GROUP [' + @AGroupName + '] ADD DATABASE [' + @db + ']';  
  
EXEC sp\_executesql @AddDbToAG;  
  
PRINT 'Bringing replica online'       
  
DECLARE @AddHADR NVARCHAR(2048) = 'EXEC ' +'['+ @linkedserver+']'+'.master.dbo.sp\_executesql N''ALTER DATABASE [' + @db + '] SET HADR AVAILABILITY GROUP = [' + @AGroupName +'];''';  
  
EXEC sp\_executesql @AddHADR  
  
  
FETCH AG\_cr INTO  
  @db  
  
END  
  
 CLOSE AG\_cr;  
 DEALLOCATE AG\_cr;