**Azure Blob Storage**

**Portal**  
Şəkil fayllarını Azure Blob Storage-də saxlamaq üçün ilk öncə [Azure Portal](https://portal.azure.com) - a daxil oluruq və əgər hesabımız yoxdursa özümüzə hesab açırıq.

**Storage Account**  
Qeydiyyat prosesi tamamlandıqdan sonra əsas səhifədə Storage Accounts bölməsinə daxil olub Create düyməsinə klikləyirik.Primary Service olaraq “Azure Blob Storage or Azure Data Lake Storage Gen 2” seçilir. Çünki biz şəkil və video fayl saxlamaq məqsədi ilə və lazım olarsa endirmək məqsədi ilə bu Storage Account-u yaradırıq.

Create prosesi yekunlaşdıqdan sonra yaranan Storage Account-a daxil oluruq. Burada Containers-ə daxil olaraq yeni Container(images və ya uploads adlı) yaradırıq. Yaranan Container-in Access Level-i default olaraq Private olur.

**Container & ConnectionString**  
Yaranan container-in ConnectionString-i əldə edilməlidir. Bunun üçün Storage Account səhifəsində Access Keys bölməsinə daxil olub Key1 yazılı hissədə Connection String-i əldə edirik. Key2 hissəsi Key1 ilə eyni funksionallığa malikdir. Sadəcə hər hansısa açarın birində sıxıntı yaşanarsa problem yaşamadan ikinci açarı istifadə etmək mümküdür.

**BlobServiceClient**

Azure.Storage.Blobs package vasitəsilə əldə etdiyimiz BlobServiceClient class-dan instance alaraq Storage Account səviyyəsində Container-lər üzərində əməliyyatlar icra edə bilərik. Container yaratmaq, əldə etmək və silmək mümkündür. Instance alan zaman Constructor, ConnectionString tələb edir.

GetBlobContainers() - Storage Account daxilində bütün Container-ləri əldə etmək üçün

CreateBlobContainer(string containerName) - Storage Account daxilində verilmiş adda Container yaratmaq üçün

DeleteBlobContainer(string containerName) - Storage Account daxilində verilmiş adda Container və daxilində yer alan Blob-ları silmək üçün

**BlobContainerClient**

BlobContainerClient obyekti vasitəsilə Container səviyyəsində Blob-lar üzərində əməliyyatlar icra edə bilərik. Blob yaratmaq, silmək, Container yaratmaq və silmək əməliyyatlar icra edə bilərik.

BlobServiceClient obyekinə tətbiq edilən GetBlobContainerClient(blobContainerName) metod ilə Container-i əldə edə bilərik. Həmçinin bir başa connectionString və containerName parametrlərini göndərərək instance ala bilərik.

CreateIfNotExist() - Storage Account daxilində həmin Container yoxdursa yaradır.

DeleteIfNotExist() - Storage Account daxilində həmin Container varsa silmək üçün.

GetBlobs() - Storage Account daxilində yer alan bütün Blob-ların siyahısını əldə etmək üçün.

GetBlobClient(blobName) - Container daxilində verilmiş adda Blob üzərində işləmək üçün lazım olan BlobClient obyekti əldə etmək üçün. Beləliklə BlobClient https://<account>.[blob.core.windows.net/](http://blob.core.windows.net/)<container>/<blobName> ünvanını təmsil etmək üçün yaranır lakin var olub-olmamasının yoxlanılması üçün Azure-a heç bir sorğu göndərilmir.

Exist() - Storage Account daxilində həmin Container-in olub-olmamasını yoxlamaq üçün.

GetAccessPolicy() - Container-in icazəsini əldə etmək üçün

SetAccessPolicy() - Container-in Access Policy tətbiq etmək üçün.

SetAccessPolicy metodun qəbul etdiyi Enum dəyərlər aşağıdaki məqsədlə istifadə edilir.

PublicAccessType.None Fayllar anonim birbaşa link vasitəsilə baxıla bilməz. Bunun üçün SAS link generate edilməlidir.

PublicAccessType.Blob Fayllar link vasitəsilə birbaşa baxıla bilər. Amma Container daxilindəki bütün fayllar siyahısına baxmaq olmur.

PublicAccessType.BlobContainer Fayllar link vasitəsilə birbaşa baxıla bilər. Həmçinin Container daxilindəki bütün fayllar siyahısına da baxmaq olur.

**BlobClient**

BlobClient obyekti vasitəsilə Fayl upload,oxumaq,delete,download və s. əməliyyatlar

icra edə bilərik. BlobContainerClient obyektinə tətbiq edilən GetBlocClient(blobName) metod vasitəsilə Blob əldə edilir.

Upload(stream) - Fayl upload etmək üçün.

DownloadTo(fileName) - Fayl download etmək üçün. Burada verilmiş fayl adına uyğun olaraq local-a həmin adda yüklənir.

DeleteIfExist() - Blob-u əgər varsa silmək üçün. Blob yoxdursa geriyə false dəyər qaytarır.

Exist() - Blob-un olub-olmamasını yoxlamaq üçün.

GenerateSasUri - SAS Link yaratmaq üçün.  
OpenReadAsync - Stream kimi oxumaq üçün.

**Upload**

Faylın upload prosesi zamanı BlobClient obyektinə Upload metod tətbiq edilir və parametr olaraq Stream obyekti və BlobHttpHeaders obyekti göndərilir. Əlavə olaraq overwrite prosesinə ehtiyac duyularsa overwrite:true bu zaman eyni adda fayl olarsa silinir və yenisi yaranır. Əgər üzərinə yazılması istənilməzsə bu zaman overwrite:false göndərilir. Bu zaman eyni adda fayl olarsa RequestFailedException(409 Confilict Status Code) atır.

Container daxilində yer alan Blob-ların link vasitəsilə bir başa müraciət edilə bilməsi üçün BlobContainerClient.SetAccessPolicyAsync(PublicAccessType.Blob) metod tətbiq edilməlidir. Default olaraq PublicAccessType.None olur. Əlavə olaraq bilmək laazımdır ki, Account-un öz Configuration-larında “Allow Blob anonymous access” enable olmalıdır.

Stream obyekti, IFormFile obyektinə OpenReadStream() metod tətbiq edilərək əldə edilir.  
BlobHttpHeaders obyekt yaradılaraq ContentType property-ə əldə edilən IFormFile obyektinin Content Type-ı mənimsədilir. Beləliklə fayl, link vasitəsilə müraciət edilən zaman browser-də öz tipində açılır.

**Private Get Blob Uri**

BlobContainerClient obyektinə tətbiq edəcəyimiz GetBlobClient metoda parametr olaraq Blob yəni faylın adı göndəriləcək. BloblClient obyekti əldə ediləcək.

Daha sonra BlobSasBuilder obyekti yaradılır və müəyyən property-lərə dəyər set olunur.

BlobContainerName = \_containerClient.Name,

BlobName = fileName,

Resource = "b",

ExpiresOn = DateTimeOffset.UtcNow.AddMinutes(validMinutes)

Daha sonra BlobSasBuilder obyektinə SetPermissons metod tətbiq edilir və parametr olaraq BlobSasPermissions.Read enum-ı göndərilir.

Sonda isə BloblClient obyektinə GenerateSasUri metod tətbiq edilir və parametr olaraq BlobSasBuilder obyekti göndəriləcək. Əldə edilən Uri vasitəsilə artıq biz şəkili əldə edə bilərik.

**Public Get Blob Uri**

BlobContainerClient obyektinə tətbiq edəcəyimiz GetBlobClient metoda parametr olaraq Blob yəni faylın adı göndəriləcək. BloblClient obyekti əldə ediləcək və Uri property istifadə edilərək link əldə ediləcək. Storage Account-un öz Configuration-larında “Allow Blob anonymous access” Enable olduğu üçün və Container-in daxilində yer alan Blob-ların Access Policy-ləri Blob və ya BlobContainer olduğu üçün property vasitəsilə əldə edilən link vasitəsilə şəkil browser-də açılacaq.

**Delete**

Faylı storage-dən silmək üçün BloblClient obyektinə Delete metod tətbiq edilərək fayl silinir.

Lakin daha öncədən silinmiş və ya ümumiyyətlə var olmayan Blob silinən zaman error verəcək. Bunun qarşısını almaq üçün də silmədən öncə yoxlanış aparmaq lazımdır. BlobClient obyektinə Exist metod tətbiq edilərək var olub-olmaması barədə məlumat ala bilərik. Və ya hər iki əməliyyatı eyni anda DeleteIfExist metod vasitəsilə həyata keçirə bilərik.