

SQL Veri Tipleri;

Tablolar oluşturulurken alanların saklayacağı verilerin tiplerinin belirtilmesi gerekir. Girilecek veri, metin, tam sayı, ondalıklı sayı ya da tarih olabilir. Bu yüzden tablo oluşturmaya geçmeden önce SQL'de kullanılan veri tiplerinin açıklanmasında fayda vardır.

1) Metinsel Veri Tipleri

char: Unicode'u desteklemeyip char(n) şeklinde kullanılırlar. 8000 karaktere kadar değer alabilirler. Belirtilenden(n) az karakter girilse dahi giriş yapılan boyut kadar yer kaplar. Veri giriş boyutları benzer, sabit olan veri kümelerinde kullanılması önerilir

nchar: Unicode(uluslararası karakter setini, tanımlı tüm alfabeleri içerir) destekler. Char'dan farklı olarak maksimum 4000 karaktere kadar değer alabilir.

varchar: Char'dan farklı olarak verinin boyutu kadar yere kaplar. 8000 karaktere kadar depolama yapar. Birbirinden farklı uzunlukta veri girişi yapılacağı zaman kullanılması önerilir. varchar(MAX) kullanımı ile 2GB'a kadar depolama yapılabilir.

nvarchar: Verinin boyutu kadar yer kaplar. Varchar'dan farklı olarak unicode'u destekler. 4000 karaktere kadar değer alabilir.

text: Belirtilenden az değer girilse bile boyutu kadar yer kaplar. 2GB'a kadar metinsel veri depolar. Unicode'u desteklemez.

ntext: Text'den farklı olarak girilen karakter boyutu kadar yer kaplar ve unicode'u destekler.

2) Binary(İkilik) Veri Tipleri

binary: 1 ve 0 ları temsil eden ikilik taban veri tipidir. Sabit uzunluklu veri tiplerinde kullanılırlar. 8000 byte'a kadar depolama yapabilir.

varbinary: Binary'den farklı olarak girilen karakter kadar yer kaplar. Bu yüzden uzunlukların değişken olduğu durumlarda tercih edilir.

image: Resim dosyalarını saklamak için kullanılır. En fazla 2 GB'a kadar veri depolar. Bunun yerine varbinary(MAX) kullanılması tercih edilir.

3) Sayısal Veri Tipleri

bit: Bir byte uzunluğunda tam sayı veri tipidir. Genellikle evet/hayır şeklinde mantıksal bilgileri tutmak için kullanılır.

int: 4 byte büyüklüğünde, -2 milyar / +2 milyar arasında değer tutabilen tam sayı veri tipidir.

bigint: 8 byte büyüklüğünde -2^{63} ve 2^{63} arasında değer tutabilen tam sayı veri tipidir.

smallint: 2 byte büyüklüğünde -32.768 ve 32.768 arası değer alabilen tam sayı veri tipidir.

tinyint: 1 byte büyüklüğüne sahip, 0–255 arası tam sayı verileri için kullanılan tam sayı veri tipidir.

decimal, numeric: İkisinin de kullanımı aynıdır. Bu veri tipinde saklanacak sayının basamak sayısı tanımlanabilir. Veri tipi boyutu belirtilen basamak sayılarına göre değişkenlik gösterebilir. -38 ve +38 basamak arası verileri depolayabilir. -10^{38} , 10^{38} arası ondalık ve tam sayı türünde veri saklayabilir.

4)Parasal Veri Tipleri

money: 8 byte boyutunda, yaklaşık -2^{64} ile 2^{64} arasında parasal değerleri tutmak için kullanılır. 4 basamağa kadar duyarlı ondalık tipli verileri saklar.

smallmoney: 4 byte uzunluğunda yaklaşık -214.000 ile 214.000 arası parasal değerleri tutmak için kullanılır. Money tipinde olduğu gibi 4 basamağa kadar duyarlı ondalık tipli verileri saklarken kullanılır.

5)Tarih-Zaman Veri Tipleri

date: Tarihleri YYYY-AA-GG (yıl-ay-gün) formatında saklar. 3 byte uzunluğunda veri tipidir.

smalldatetime: Tarih ve zaman verilerini yıl-ay-gün ve saat-dakika-saniye-salise şeklinde saklar. 4 byte uzunluğunda veri tipidir.

datetime: YYYY-AA-GG şeklinde tarih ve zaman verilerini tutan 8 byte uzunluğunda veri tipidir. 1 Ocak 1753–31 Aralık 9999 arası veriler için kullanılır.

datetime2: Datetime'dan farklı olarak 1 Ocak 0001–31 Aralık 9999 tarihleri arası verileri tutan ekstra olarak salise hassasiyeti daha yüksektir. Kapladığı alan salise hassasiyetine göre 6–8 byte arası değişebilir.

time: Sadece saat verilerini saat-dakika-saniye-salise(7 basamaklı) şeklinde saklayan , boyutu kullanıcı tarafından değiştirilebilen 3–5 byte arası yer kaplayan veri tipidir.

datetimeoffset: Ülkelere göre değişen zaman farkını hesaplayıp tutarken kullanılır.

6) Diğer Veri Tipleri

sql_variant: sayı,metin, binary gibi farklı veri tiplerini depolamak için kullanılan veri tipidir.Yani bir sütun ya da fonksiyonda birden fazla veri tipi kullanmamız gerektiğinde tercih etmeliyiz.

xml: XML türünde veri saklamak için kullanılır. Kapasitesi 2 GB'dır.Bellekteki boyutu, saklanan XML verisine göre değişkenlik gösterir.

geometry: Öklid koordinat sistemine ait verileri tutmak için kullanılır.Geometrik şekillerin en-boy-yükselik verilerini saklar.

timestamp: Tabloya kayıt eklendiğinde , güncellendiğinde binary türünde özel değer alan veri tipidir.

uniqueidentifier: 16 byte uzunluğunda benzersiz GUID tipinde veri tutar.İki GUID birbirinden tamamen farklıdır eşit olamazlar.

hierarchyid: Ağaç veri modeli ve ya hiyerarşik olarak sınıflandırılmış verileri saklamak için kullanılır.

geography: Coğrafi koordinat ve GPS verilerini tutmak için kullanılır.

Identity:

Otomatik Artan sayı

Kişi oluşturduğu tabloda verilerin ardışık sayılardan oluşan değerleri alması isteniyorsa identity özelliği kullanılır. Yani veri ekleyeceğimiz vakit ürün veya nesnesinin ID sayısı otomatik olarak artar.

Identity alanlar güncellenemezler ve boş bırakılamazlar. identity alanlar genelde primary key olarak kullanılırlar.

Veri bütünlüğü, bir tabloda veri güncelleme, silme veya ekleme gibi işlemler yapılırken diğer tablo ya da tablolardaki verilerin birbirleriyle uyum içinde

olması, dolayısıyla veri tutarlılığının kaybolmamasının garanti altına alınması demektir.

Programsal Veri Bütünlüğü

Tanımsal Veri Bütünlüğü

Programsal Veri Bütünlüğü:

- Trigger'ler, Stored Procedurler, İş seviyesi Uygulamalar
- Transaction'dan sonra devreye giriyor veya, bir sorgu daha kullanarak kontrol ediyor. Çok kaynak tüketir. Her türlü kuralı denetleyebilir.

Tanımsal Veri Bütünlüğü

–**Constraint**; Tablo ile birlikte derlenir ve Transaction'dan önce devreye girer. Etkin ancak dinamik değil, her sorunu çözemeyebilir.

- Primary Key Constraint**
- Unique Constraint**
- Check Constraint**
- Default Constraint**
- Foreign Key Constraint**

–**Rule** = Check Constraint Tablo dışında nesne.

–**Default** = Default Constraint Tablo dışında nesne.

Primary Key Constraint :

- Daha önce girilmemiş değerler girmeye zorlar, her satırın tekilliğini sağlar.
- NULL kalamaz.
- Her tabloda en fazla 1 adet bulunabilir.
- SQL Server Tarafından, arka planda Unique Indeks olarak gerçekleştirir.

Unique Constraint

- Bir tabloda birden fazla sütunda tanımlanabilir.
- SQL Server tarafından, arka planda Unique Indeks olarak gerçekleştirir.
- NULL kalabilir ama NULL değil ise, mutlaka girilmemiş verilerden gelmelidir. (Ya NULL kal ya tekrarlama)

–Örnek:Kişi kimlik bilgilerinin tutulduğu bir tabloda kişinin sicilNumarası , tcKimlikNumarası, vergiNumarası, sosyalGuvenlikNumarası gibi bilgiler kişiye özel bilgileridir. Fakat bu bilgilerden yalnızca bir tanesi primary key olabilir. Diğer kişiye özel bilgilerin tekrarını engellemek için unique key tanımlanır.

Check Constraint =RULE

- Belli bir formata uygun veri girişi için,
- Aynı tablonun aynı satırında iki farklı sütun değerini karşılaştırmak için,

Örnek1: Doktor bilgilerinin tutulduğu bir tabloda doktorun iseBaşlamaTarihi ve

istenAyrılmaTarihi karşılaştırılmalıdır. Çünkü işten ayrılma tarihi başlama tarihinden önce olamaz.

Örnek2: Öğrenci notlarının tutulduğu bir tabloda bir dersten alınabilecek not aralığının 0 ile 100 arasında olmasının kontrolü gibi.

Default Constraint = DEFAULT

–Bir sütuna girilmek üzere değer verilmezse, girilebilecek bir standart değer tanımlar.

–Sadece INSERT işleminde devreye girer.

Örnek :tbl_ogrenci tablosunda sisteme yeni bir öğrenci ekleneceği zaman ogrenciKayıtTarihini günün tarihi olarak eklenmesi. Böyle bir durumun gerçekleşmesi için kayıt tarihinin sisteme eklenmemesi gerekmektedir.

Foreign Key Constraint

–Bir tablodaki bir sütuna ait değerlerin, başka bir tablonun belli sütunundan gelmesini denetler.

–CASCADE UPDATE, CASCADE DELETE özelliği var.

Örnek; OgrenciNot tablosunda bulunan ogr_id Ogrenci tablosundaki ogr_id ile ilişkilendirilmemesi durumunda öğrenci tablosunda bulunmayan bir öğrenciye ait not sisteme kayıt edilebilir.

Tablo Seviyeli Kısıtlamalar:

Veri Bütünlüğü	Kısıtlayıcı (Constraint)	Kontrol
Satır Bazlı	PRIMARY KEY	Bir satırın bir alanı için, NULL alamaz. Girilmiş değerler girilemez, arka planda otomatik indeks tanımlanır.
	UNIQUE	PK'dan başka alanlara girilen değerlerin tekrar girilememesini sağlar. NULL kalabilir.
Sütun Bazlı	CHECK	Bir sütuna girilen değerler belli kümeden mi geliyor, Belli koşullara uygun mu? Büyüklük-küçüklük-format vs.
	DEFAULT	Bir sütun için, INSERT yapılırken, değer yok ise belli bir sabit değer atar.
	REFERANS (PK-FK)	Bir sütuna girilen değerlerin başka bir tablonun başka bir sütunundan geliyor olması
Referans Bütünlüğü	CHECK	Bir tablonun, bir sütuna girilen değerler, diğer sütunlara girilen değerlerle çelişiyor mu?(Aynı satır için)
	FOREIGN KEY	Bir sütuna girilen değerlerin başka bir tablonun başka bir sütunundan geliyor olması