

Omron Sysmac ve MySQL Server Arasında Veri İletişimi

İÇİNDEKİLER

- Giriş
- SQL Nedir?
- SQL Server Konfigürasyonu
- Uygulama Adımları

Giriş

Endüstriyel otomasyon sistemlerinde, kontrol cihazları ile bilgi teknolojileri altyapısı arasındaki veri akışı giderek daha önemli hale gelmektedir. Omron Sysmac platformu, NJ/NX serisi kontrolörlerde bulunan yerleşik SQL Client fonksiyon blokları sayesinde verilerin doğrudan ilişkisel veritabanlarına aktarılmasına olanak tanır. Bu yapı sayesinde üretim hattındaki sensör, aktüatör ve proses bilgileri güvenli bir şekilde MySQL Server üzerinde saklanabilir, analiz edilebilir ve raporlanabilir.

Bu dokümanda Omron NJ Serisi PLC ile MySQL Server arasındaki bağlantının kurulması, gerekli ön koşullar ve örnek uygulama adımları ele alınacaktır. Amaç, otomasyon verilerinin gerçek zamanlı olarak veritabanına entegrasyonunu sağlayarak hem üretim verimliliğini artırmak hem de izlenebilirliği geliştirmektir.

Database bağlantısını destekleyen CPU modelleri aşağıdaki tablolarda mevcuttur.

Ürün	Max. number of synchronous axes			
☐ NX701-1620	128			
☐ NX701-1720	256			
Ürün	Max. number of synchronous axes			
☐ NX502-1300	16			
☐ NX502-1400	32			
☐ NX502-1500	64			
☐ NX502-1600	128			
☐ NX502-1700	256			
Ürün	Supports motion control	Max. number of synchronous axes		
☐ NX102-1020	Yes	2		
☐ NX102-1120	Yes	4		
☐ NX102-1220	Yes	8		
☐ NX102-9020	No	0		
Ürün	Max. number of synchronous axes	Supports robotics control	Max. number of robots	
☐ NJ501-1320	16	No	0	
☐ NJ501-1420	32	No	0	
☐ NJ501-1520	64	No	0	
☐ NJ501-4320	16	Yes	4	
Ürün	Supports motion control	Max. number of synchronous axes		
☐ NJ101-1020	Yes	2		
☐ NJ101-9020	No	0		

SQL Nedir?

SQL (Structured Query Language), ilişkisel veritabanı yönetim sistemlerinde verilerin düzenlenmesi ve yönetilmesi için kullanılan standart bir sorgulama dilidir. SQL sayesinde veriler üzerinde **ekleme (INSERT)**, **okuma (SELECT)**, **güncelleme (UPDATE)** ve **silme (DELETE)** işlemleri gerçekleştirilebilir.

Endüstriyel otomasyonda SQL'in en büyük avantajı, PLC gibi kontrol cihazlarından toplanan verilerin merkezi bir veritabanında saklanması ve bu verilerin analiz, raporlama, bakım planlama ve üretim optimizasyonu için kullanılabilmesidir. Böylece sahadaki veriler yalnızca anlık kontrol için değil, aynı zamanda kurumsal bilgi sistemleriyle entegrasyon için de değer kazanır.

Günümüzde yaygın olarak kullanılan bazı popüler SQL tabanlı veritabanı yönetim sistemleri şunlardır: **MySQL**, **Microsoft SQL Server**, **Oracle Database** ve **PostgreSQL**.

Bu dökümanda uygulama için **MySQL Server** kullanılacaktır.

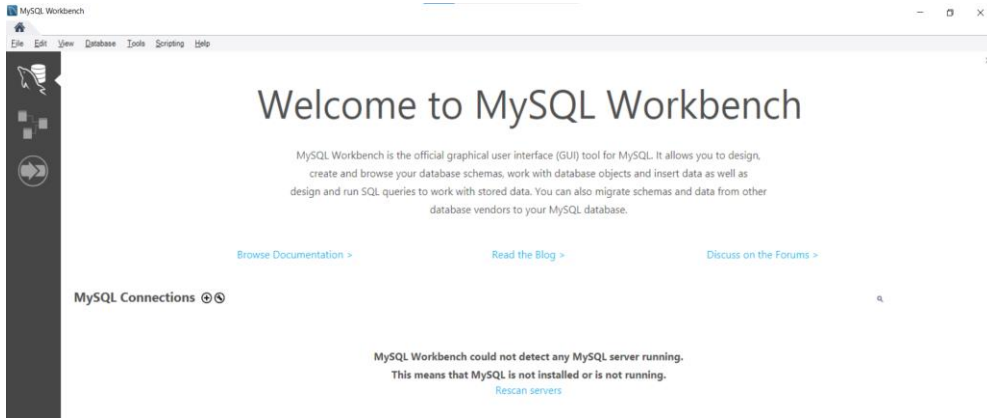


MySQL Server, resmi internet sitesinden indirilebilir:

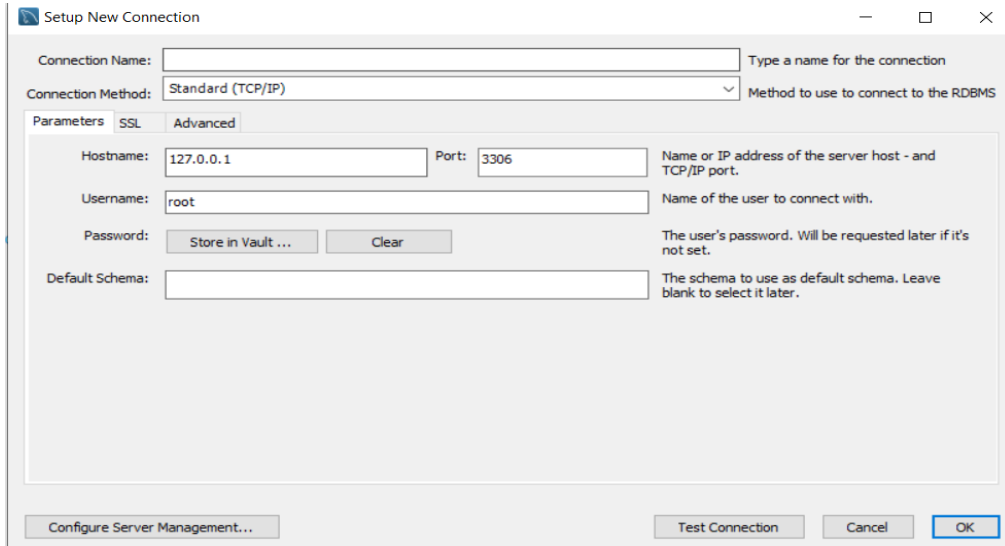
<https://dev.mysql.com/downloads/mysql/>

SQL Server Konfigürasyonu

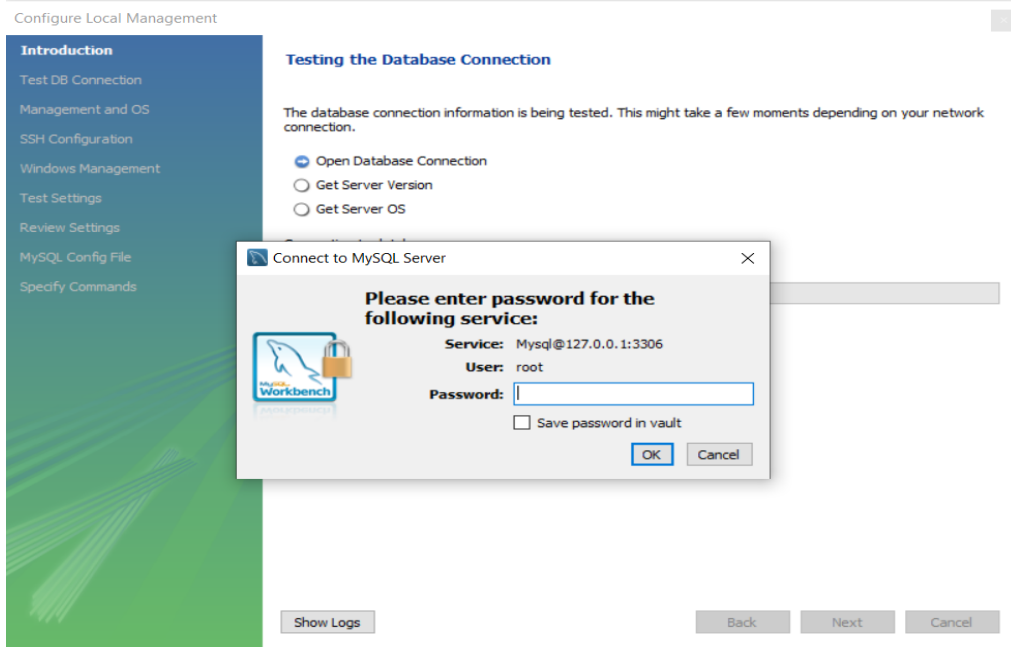
MySQL Workbench açıldığında karşımıza gelen ilk ekranda, sol tarafta yer alan **MySQL Connections** kısmından yeni bir bağlantı oluşturmak için artı (+) işaretine tıklanır.



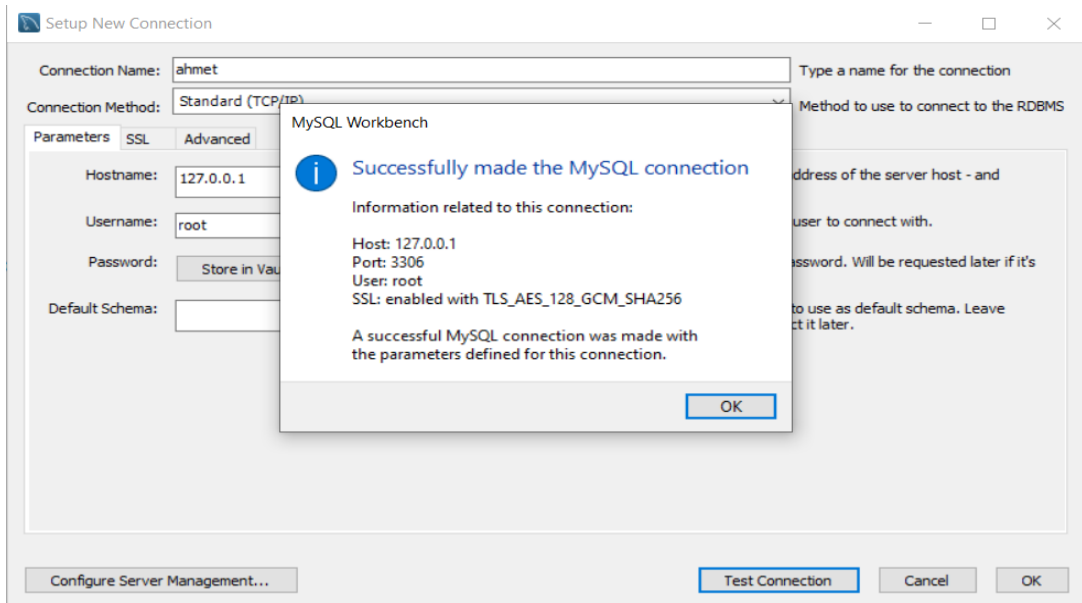
Açılan ekranda bağlantı bilgileri varsayılan olarak gelmektedir. Bu aşamada sadece **Configure Server Management** seçeneğine tıklanarak sunucu ayarları ekranına geçilir.



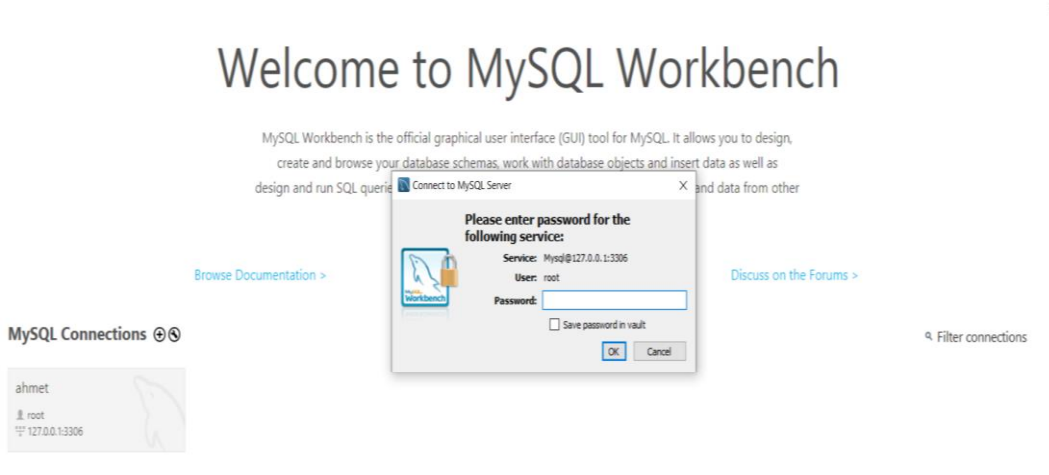
Bu aşamada açılan pencerede **root kullanıcısı** için bir şifre belirlenir. Şifre yazıldıktan sonra **OK** butonuna basılarak sunucu bağlantısı doğrulanır.



Ardından MySQL Connections ekranına geri dönerek bu aşamada bağlantıya bir isim verilir (örneğin **ahmet**) ve ardından **Test Connection** seçeneği tıklanarak bağlantının doğru çalışıp çalışmadığı kontrol edilir. Test başarılı olduğunda **OK** butonuna basılarak bağlantı kaydedilir.



Oluşturulan sunucuya bağlanmak için, bağlantı adının üzerine tıklanır ve kurulum aşamasında belirlenen parola ilgili alana girilerek oluşturulan MySQL Server'a erişilir.



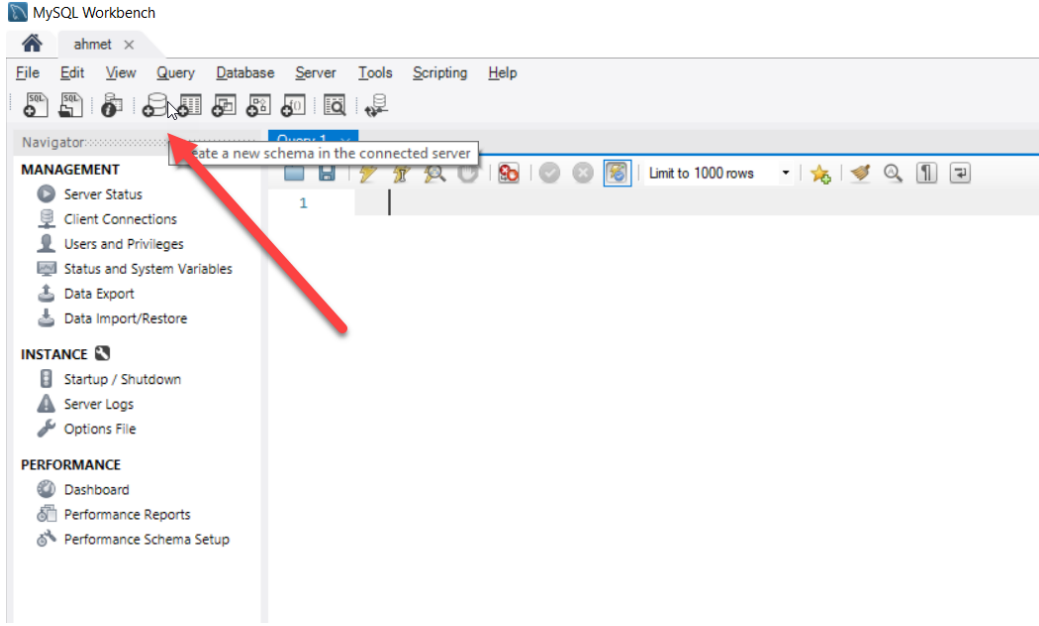
Uygulama Adımları

1. MySQL Uygulama Kısım

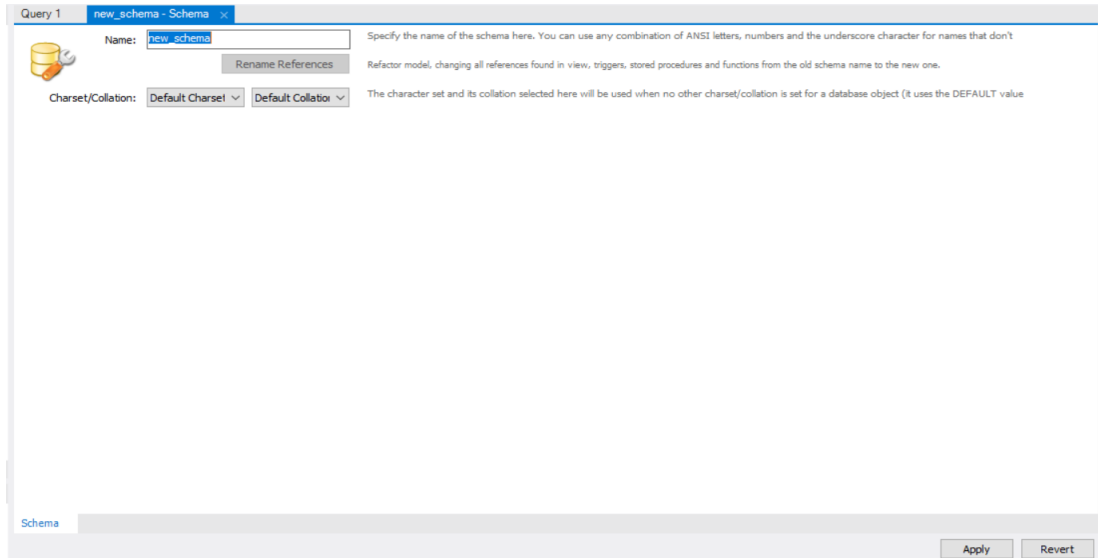
Uygulamada bir servo motorun hız, konum ve çalışma durumu gibi değerleri sürekli olarak MySQL Server ile paylaşılacaktır. Bu nedenle, söz konusu verilerin depolanabilmesi için tabloda ilgili alanların (*ID*, *VELOCITY*, *POSITION*, *POWERON*) uygun veri tipleri ile tanımlanması gerekmektedir. Böylece servo motorun çalışma parametreleri düzenli olarak kayıt altına alınarak daha sonra analiz ve kontrol amaçlı kullanılabilir.

İlk uygulama adımı olarak, MySQL Server üzerinde bir veritabanı oluşturulacak ve bu veritabanı içerisinde ilgili parametrelerin tanımlandığı bir tablo oluşturulacaktır. Daha sonraki aşamada oluşturulan bu veritabanına Omron Sysmac üzerinden bağlantı sağlanarak veri alışverişi gerçekleştirilecektir.

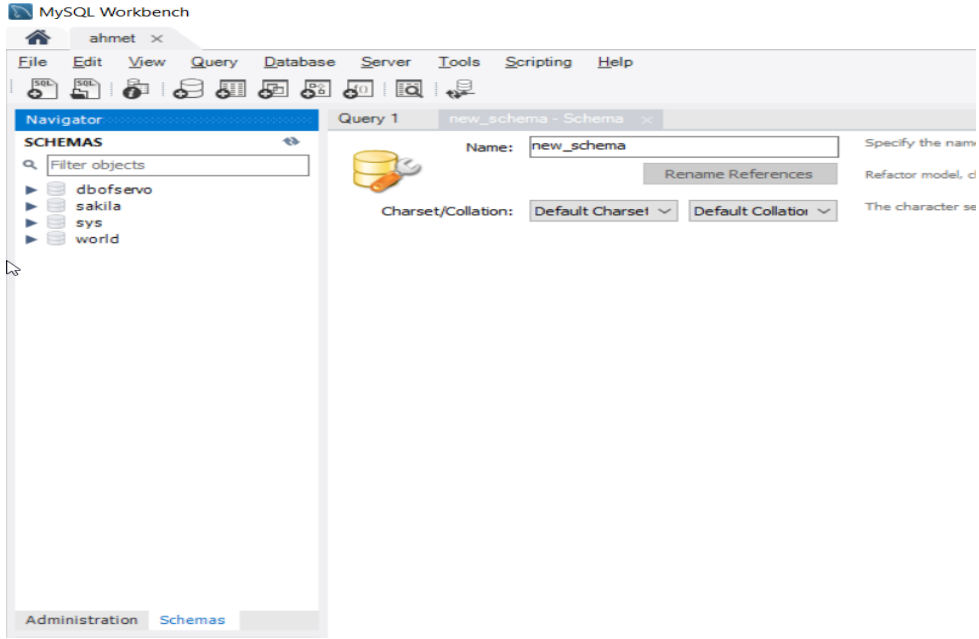
Data base oluşturmak için aşağıda gösterilen butona tıklanır;



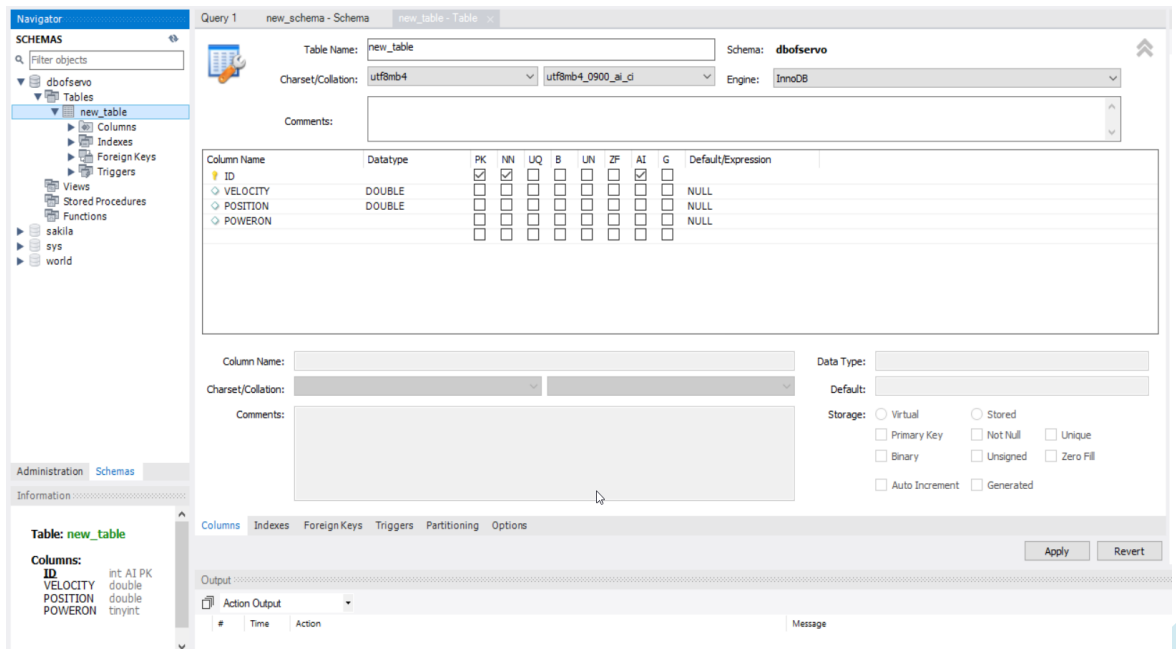
Veritabanı oluşturma adımında, ilgili veritabanına uygun bir isim verildikten sonra işlem onaylanır ve oluşturma süreci tamamlanır.



Ardından *Schemas* sekmesine tıklanarak oluşturulan veritabanının özellikler kısmına erişilir.



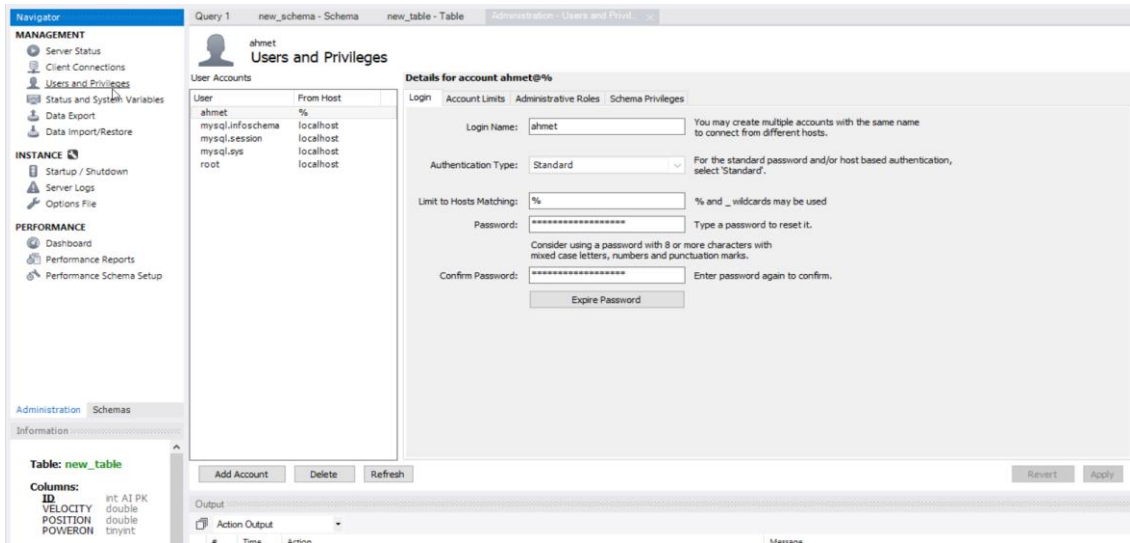
Ardından, *new_table* kısmında yer alan ayarlar bölümüne geçilerek kullanılacak değişkenler uygun veri tipleri ile tanımlanır. Bu kapsamda, servo motor uygulamasında gerekli olan kimlik (*ID*), hız (*VELOCITY*), konum (*POSITION*) ve çalışma durumu (*POWERON*) değişkenleri tabloda ilgili veri tipleriyle yapılandırılır. Değişkenler uygun şekilde tanımlandıktan sonra *Apply* seçeneğine tıklanarak tablo oluşturma işlemi tamamlanır.



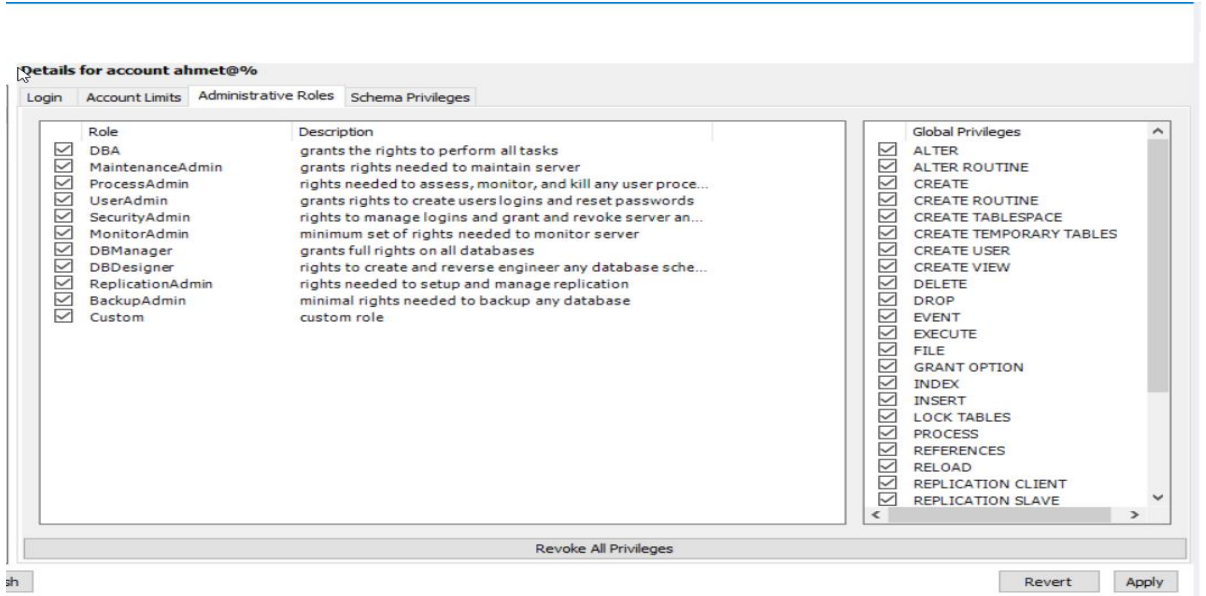
Bu işlemden sonra tablo MySQL tarafında oluşturulmuş olur. Ardından Sysmac tarafında ilgili veritabanına erişim sağlandıktan sonra tabloda gerçekleşen herhangi bir değişim (örneğin hız, konum veya çalışma durumu gibi parametrelerdeki güncellemeler), anlık olarak kayıt altına alınır ve sistem tarafından işlenebilir hale gelir.

Son adımda, oluşturulan veritabanının kullanıcılar tarafından erişilebilir hale gelebilmesi için MySQL Server üzerinden kullanıcı tabanlı yetkilendirme yapılması gerekmektedir. Bu işlem ile kullanıcıya belirli erişim izinleri (okuma, yazma, güncelleme vb.) atanarak veri güvenliği ve sistem bütünlüğü sağlanır.

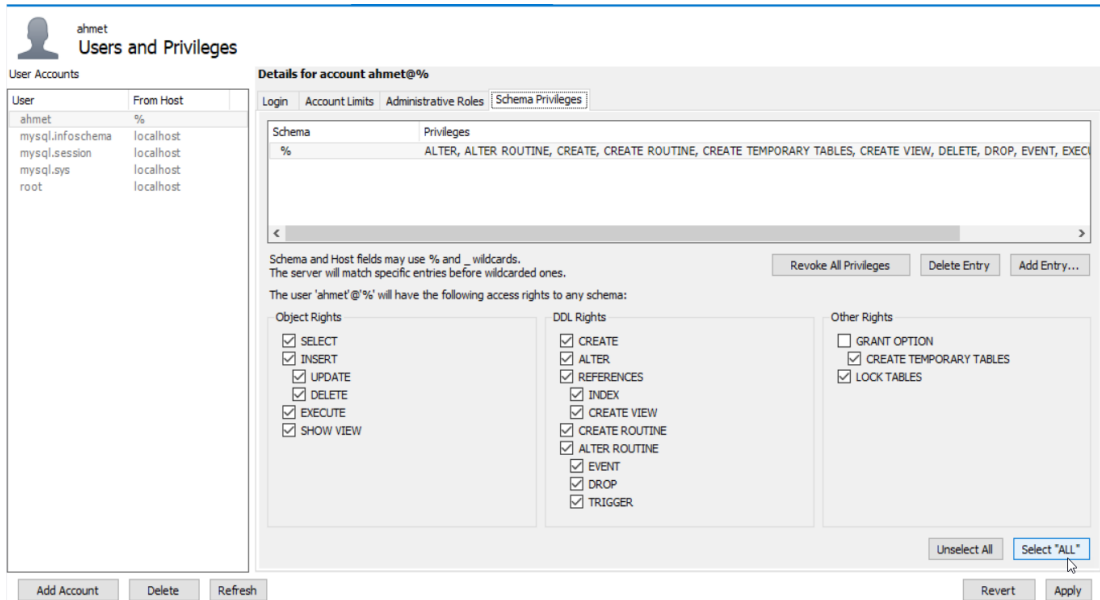
MySQL Workbench'te **Users and Privileges** bölümünden kullanıcı adı ve şifre tanımlanır. **Login Name** kısmına isim yazılır, **Authentication Type** *Standard* seçilir. **Limit to Hosts Matching** alanında % girilirse tüm istemcilerden, localhost girilirse sadece yerelden erişim sağlanır.



Kullanıcı oluşturulduktan sonra, **Users and Privileges** → **Administrative Roles** sekmesinden gerekli roller atanır.

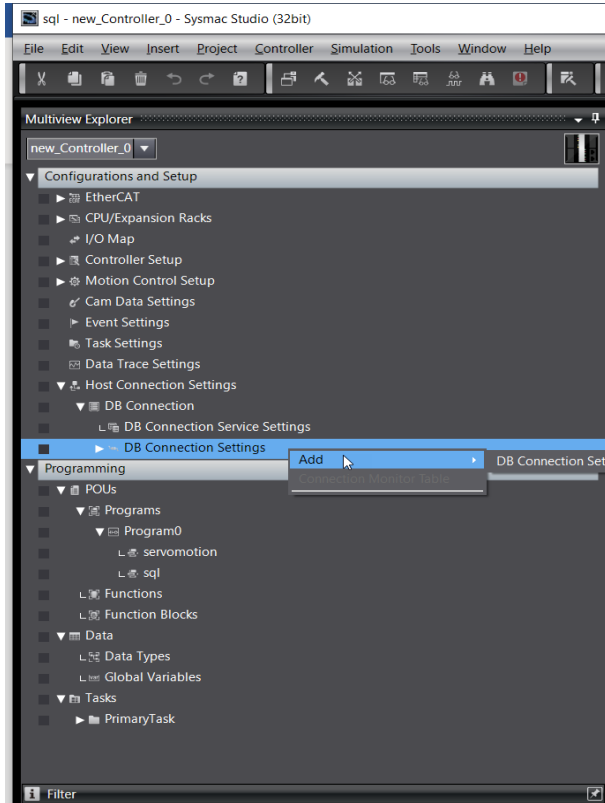


Kullanıcı oluşturulduktan sonra **Schema Privileges** sekmesine geçilerek ilgili veritabanı veya tüm şemalar için yetkiler belirlenir. Burada **Select All** seçeneği kullanılarak kullanıcıya tüm temel haklar (SELECT, INSERT, UPDATE, DELETE vb.) tanımlanır. Son olarak **Apply** butonuna basılarak izinler aktif hale getirilir.

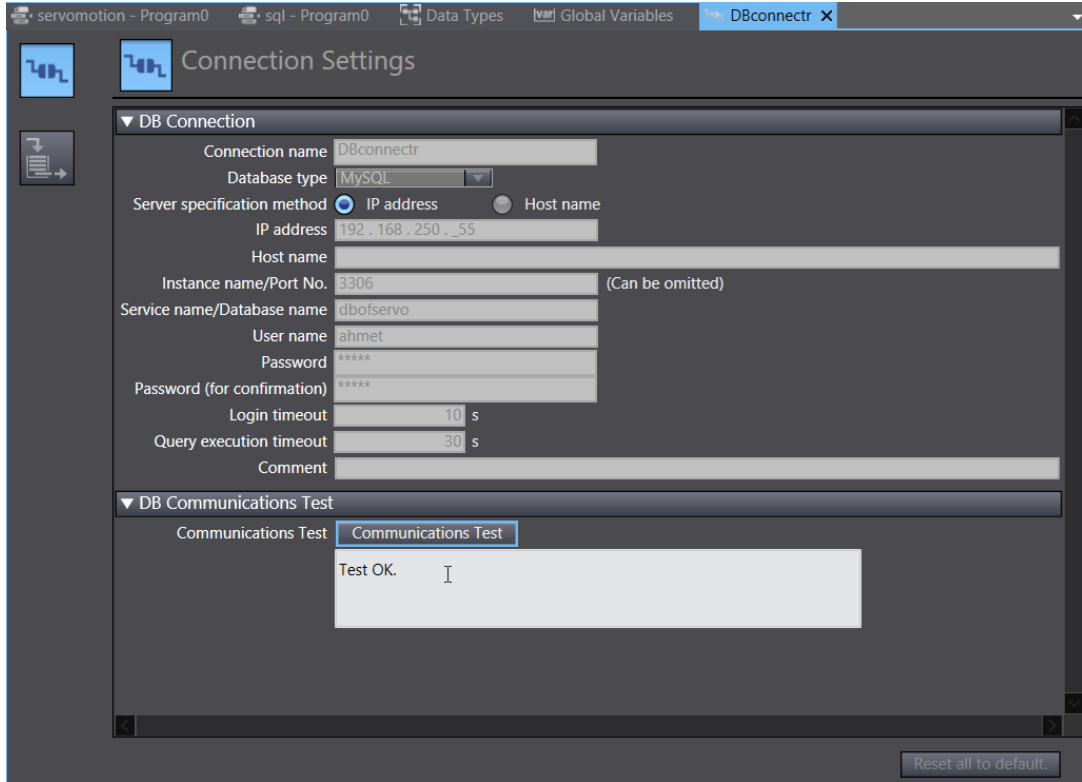


2. Sysmac Studio Uygulama Kısımı

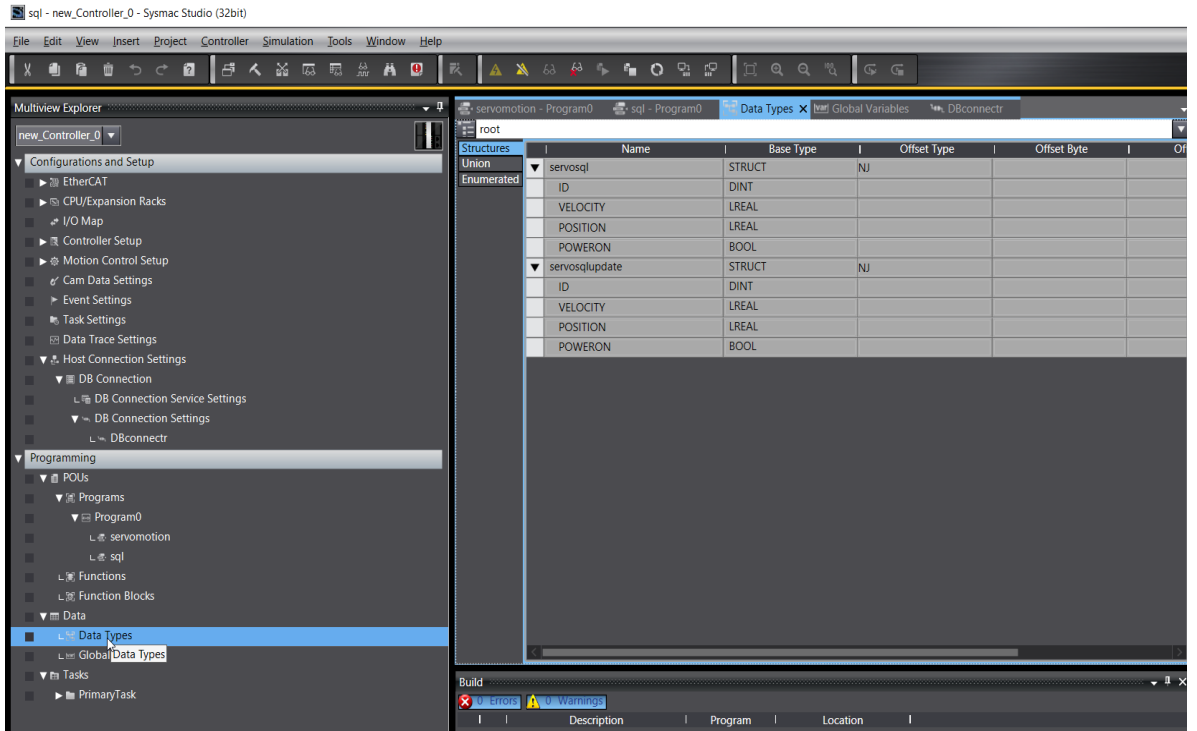
İlk olarak Sysmac Studio ortamında *Host Connection Settings* menüsü altında yer alan *DB Connection* bölümüne gelinir. Buradan *DB Connection Settings* seçeneği açılarak ‘Add’ komutu ile yeni bir bağlantı tanımlaması yapılır.



MySQL tarafında kullanıcı ve yetkiler tanımlandıktan sonra, Sysmac Studio ortamında **DB Connection Settings** penceresi üzerinden bağlantı ayarları yapılır. Bu aşamada veritabanı türü *MySQL* seçilir, sunucuya ait **IP adresi** ve **Port numarası (3306)** girilir. Ayrıca MySQL tarafında oluşturulan **veritabanı adı**, **kullanıcı adı** ve **şifre** bilgileri doldurulur. Son olarak Online olduktan sonra **Communications Test** butonu ile bağlantı testi gerçekleştirilir ve *Test OK* sonucu alınarak yapılandırma tamamlanır.



Bağlantı doğrulandıktan sonra Sysmac Studio ortamında **Data Types** sekmesine geçilerek, MySQL tarafında oluşturulan tablo ile uyumlu olacak şekilde bir **Structure (STRUCT)** tanımlanır. Bu yapı içerisinde tablo sütunlarına karşılık gelecek değişkenler (örneğin ID, VELOCITY, POSITION, POWERON) uygun veri tipleri ile belirtilir. Böylece PLC tarafındaki değişkenler ile veritabanı tablosu arasında doğrudan eşleşme sağlanır.

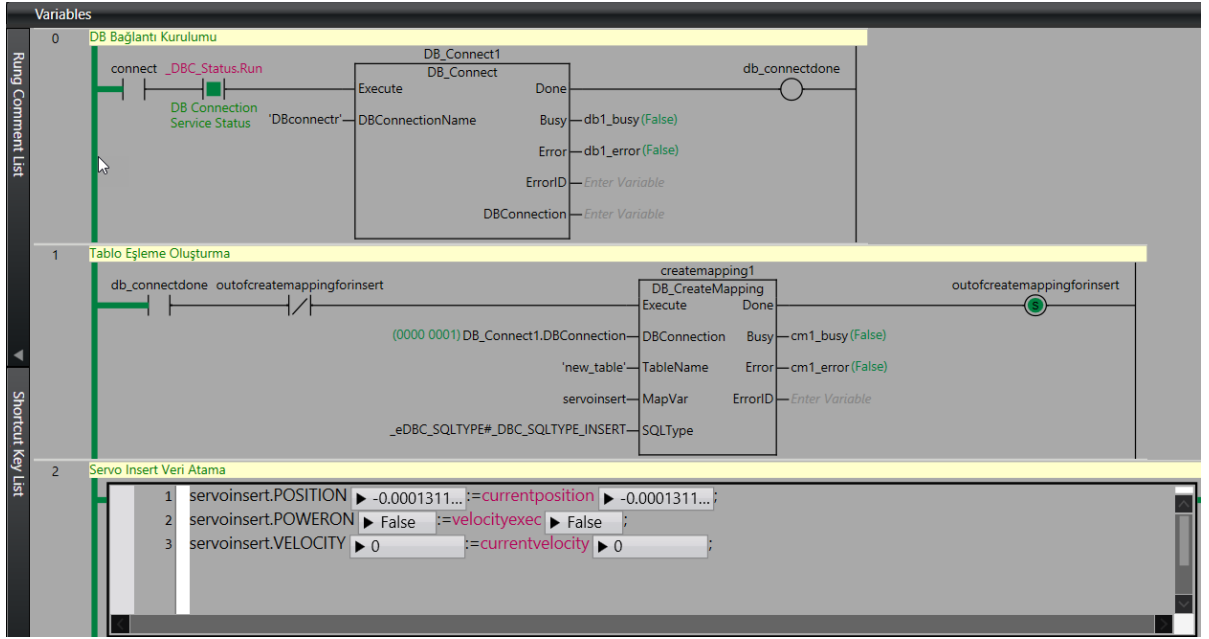


Veri yapıları tanımlandıktan sonra **Programming** bölümüne geçilerek gerekli program yazılır. Bu aşamada tanımlanan **structure** değişkenleri kullanılarak MySQL tablosuna veri ekleme, güncelleme veya okuma işlemleri gerçekleştirilir. Böylece PLC üzerinde çalışan uygulama ile veritabanı arasında tam entegrasyon sağlanır.

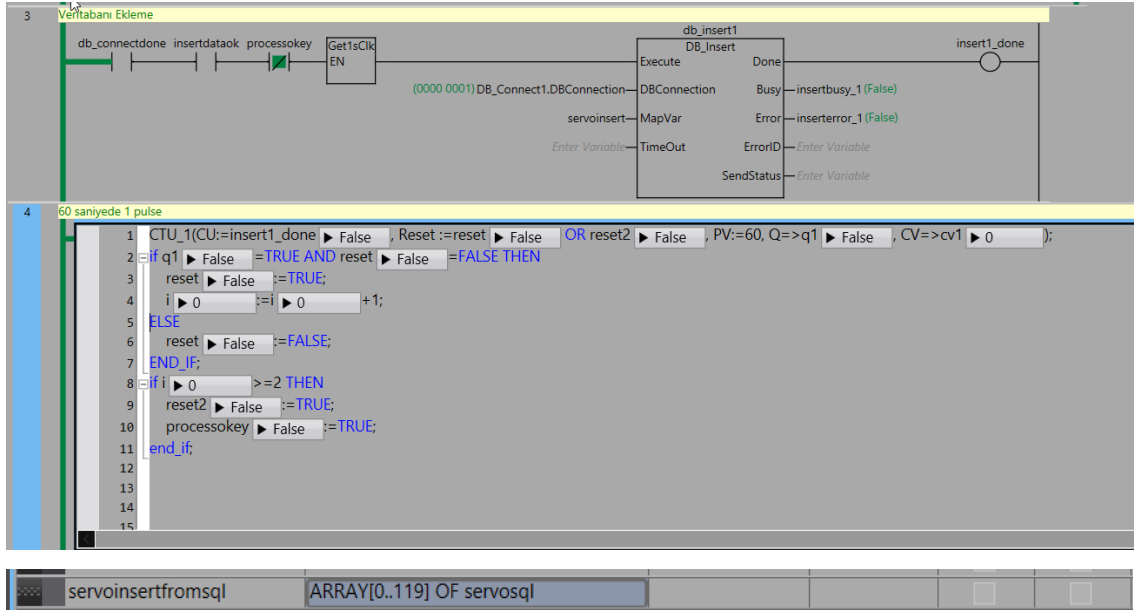


Programlama aşamasında oluşturulan **servosql** structure, veritabanına veri ekleme işlemleri için **servo_insert** olarak tanımlanmıştır

Uygulamada ilk olarak **DB_Connect** fonksiyonu kullanılarak önceden yapılandırılan MySQL veritabanına bağlantı kurulmuştur. Bağlantı başarılı olduktan sonra, Sysmac Studio'da tanımlanan **servosql** yapısı, MySQL tarafında oluşturulan tablo ile **DB_CreateMapping** fonksiyonu aracılığıyla eşleştirilmiştir. Bu fonksiyonda SQL type kullanım amacına göre değişkenlik gösterebilmektedir. Bu sayede PLC'deki değişkenler doğrudan veritabanı tablosuna aktarılacak şekilde yapılandırılmıştır.



Bağlantı kurulup tablo eşleştirmesi tamamlandıktan sonra **DB_Insert** fonksiyonu kullanılarak veriler periyodik olarak MySQL tablosuna aktarılmıştır. Programda her 60 adet *done* sinyali sonrasında sayaç değeri bir artırılmakta, bu sırada **q1** kısa süreli bir tetikleme (pulse) üretmektedir. Sayaç değeri ikiye ulaştığında ise **processokey** aktif olmakta ve ekleme işlemi tamamlanmaktadır. Böylece PLC tarafından elde edilen veriler belirli aralıklarla güvenilir şekilde veritabanına yazılmaktadır.



Uygulamanın bu aşamasında, tabloda kullanılan mod **SELECT** olarak ayarlanmıştır. Bunun nedeni, MySQL tarafında belirli kayıtların okunarak PLC ortamına aktarılmak istenmesidir. Bu amaçla **servosql** yapısı, **servoinsertfromsql** adında bir dizi olarak tanımlanmış ve tablo ile eşleştirilmiştir. Böylece SQL sunucusuna yapılan sorgular sonucunda elde edilen satırlar, doğrudan bu diziye aktarılmaktadır.

Örneğin, 120 kez insert işlemi yapıldığında veritabanında oluşan 120 satır, aynı anda **servoinsertfromsql** dizisine gelmektedir. Bu yapı sayesinde PLC, veritabanında bulunan verileri toplu halde alabilir ve uygulama içerisinde kullanabilir.

aktarılır. Daha sonra **ikinci 60 saniyelik periyot** tamamlandığında, yani 120. saniyede, yeni kayıtlar tekrar diziyeye alınır ve böylece işlem tamamlanır.

The screenshot displays the Omron Sysmac Studio interface. The top menu bar includes 'servomotion - Program0', 'sql - Program0', 'Data Types', 'Global Variables', and 'DBconnect'. The main workspace shows a ladder logic program with the following steps:

```

1 CTU_1(CU:=insert1_done ▶ False, Reset :=reset ▶ False OR reset2 ▶ True, PV:=60, Q=>q1 ▶ False, CV=>cv1 ▶ 0);
2 If q1 ▶ False =TRUE AND reset ▶ False =FALSE THEN
3 reset ▶ False :=TRUE;
4 i ▶ 2 :=i ▶ 2 + 1;
5 ELSE
6 reset ▶ False :=FALSE;
7 END_IF;
8 If i ▶ 2 >=2 THEN
9 reset2 ▶ True :=TRUE;
10 processkey ▶ True :=TRUE;

```

Below the ladder logic, the 'Watch (Table1)' window is open, showing a table with the following data:

ID (DINT)	VELOCITY (LREAL)	POSITION (LREAL)	POWERON (BOOL)
[0] 1	100.000000000364	10287.7003457069	True
[1] 2	99.9999999985448	10387.0003457069	True
[2] 3	99.9999999985448	10487.0003457069	True
[3] 4	99.9999999985448	10587.0003457069	True
[4] 5	99.9999999985448	10687.0003457069	True
[5] 6	99.9999999985448	10787.0003457069	True
[6] 7	100.000000000364	10886.9003457069	True
[7] 8	99.9999999985448	10987.0003457069	True
[8] 9	99.9999999985448	11087.0003457069	True

At the bottom, there are dimension input fields: Dim. 1: 0, Dim. 2: 0, Dim. 3: 0, and a 'Go to' button.