

# Tutorials / How-to

We do understand that getting started with Ipesoft D2000 can be complicated. That's why the D2000 documentation includes these simple tutorials.

The following recordings from the webinar, available on YouTube, are also used to familiarize yourself with the basics of the work:

- [What is Ipesoft D2000](#)
- [D2000 architecture, processes, screenshots, and use cases](#)
- [Creating an application](#)
- [Communication: creating Modbus server and Modbus client](#)
- [Historian - putting values into a historical database](#)
- [Structured values](#)
- [Schemes \(pictures\) - visualization, displaying of values](#)
- [Display palettes](#)
- [Smart Web and thin client](#)
- [Exporting values to Excel – D2000 Workbook](#)



## Tip

To familiarize yourself with the D2000 architecture, we recommend the following blogs:

- [SCADA architecture - referential integrity](#)
- [SCADA architecture - DODM](#)
- [SCADA architecture - object value](#)

and the following chapters of the documentation: [The D2000 System Structure](#) and [Dynamic Object Data Model - DODM](#).

When creating a simple SCADA application, we recommend the following procedure:

- [D2000 Installation](#).
- [Creating an application](#).
- [Starting user processes](#) and subsequent work with [D2000 Cnf](#).
- [Configuration of communications](#) - configuration of communications objects: lines, stations, and measurement points (I/O tags).
- [Archiving](#) - creation of primary historical values (archiving of communication objects, or [columns of structured variables](#)), creation of statistical and calculated historical values.
- [Evaluated tags](#) - calculating values from I/O tags ( or columns of structured variables and other object types) without the need for archiving.
- [Alarming](#) - configuration of alarms on I/O tags or evaluated tags, or configuration of Alarm type objects.
- [Scripting](#) - creation of ESL or Java scripts to perform automated actions (triggered on-change, or server scripts that contain procedures that can be called from other scripts or from visualization).
- [Working with databases](#) - configuration of objects of type Database and Table and working with SQL databases using the [D2000 DBManager](#) process.
- Visualization - the creation of schemes (pictures) and compositions in [D2000 GrEditor](#), displaying them in D2000 HI, and creating graphs.

When creating a larger SCADA application that works with several repetitive technological units (production lines, PLCs, machines), we recommend reading the [introduction to working with structure definitions and structured variables](#).

When creating a database-oriented application, we recommend the following procedure:

- Configuration of ODBC connections in Windows ODBC Data Source Administrator for the [D2000 DBManager](#) process (or TNS items in case of using Oracle database and Oracle-specific Dbmanager\_ora process).
- Configuration of objects of [Database](#) and [Database Table](#) types in D2000 Cnf.
- Creation of ESL or Java scripts for performing automated activities (data pumps and the like, or server scripts that contain procedures that can be called from other scripts or from visualization).
- Visualization - creation of schemes (pictures) containing [Browser](#) objects to display data from databases, creation of application logic (script within the scheme).

Other topics:

- Creating an application for multiple languages - [working with the Dictionary](#): in the configuration of descriptions, static texts, and other text parameters of [many types of D2000 objects](#), the syntax `{!Key_name}` can be used, where *Key\_name* is the name of the key in the Dictionary. For individual users, it is possible to set the selected language and all keys will be translated according to the definition in the Dictionary.