

# intellect.dbi

intellect.dbi :

1. intellect.dbi, . , . , . , . :
2. intellect.dbi . , , . , . :

<>, <> [, <>]



CHAR.

|           |  |
|-----------|--|
|           |  |
| id        |  |
| name      |  |
| parent_id |  |
| flags     |  |



!

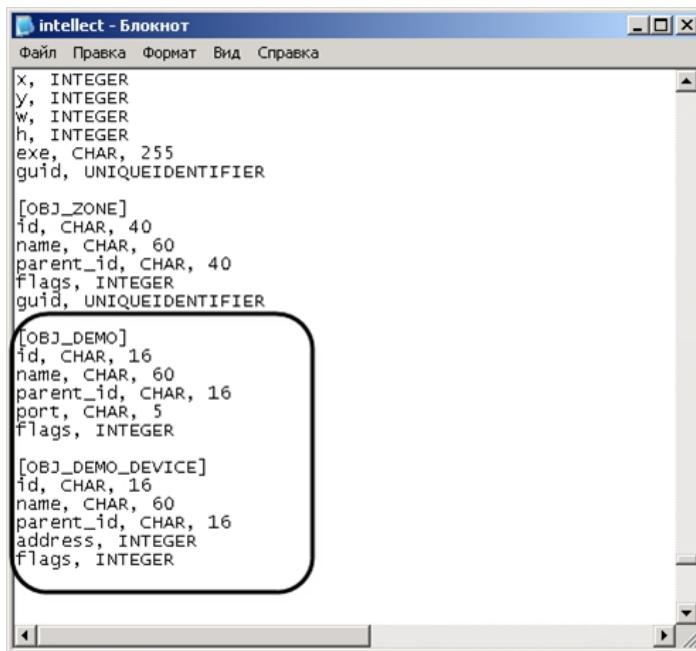
flags

|          |                |
|----------|----------------|
|          |                |
| BIT      | , «» «»        |
| CHAR     | ,              |
| DATETIME | , . - --, - :: |
| DOUBLE   | ,              |
| INTEGER  | ,              |
| TEXT     | ,              |

DEMO, , :

- a. port – COM-;
- b. address – .

intellect.dbi .



```
x, INTEGER
y, INTEGER
w, INTEGER
h, INTEGER
exe, CHAR, 255
guid, UNIQUEIDENTIFIER

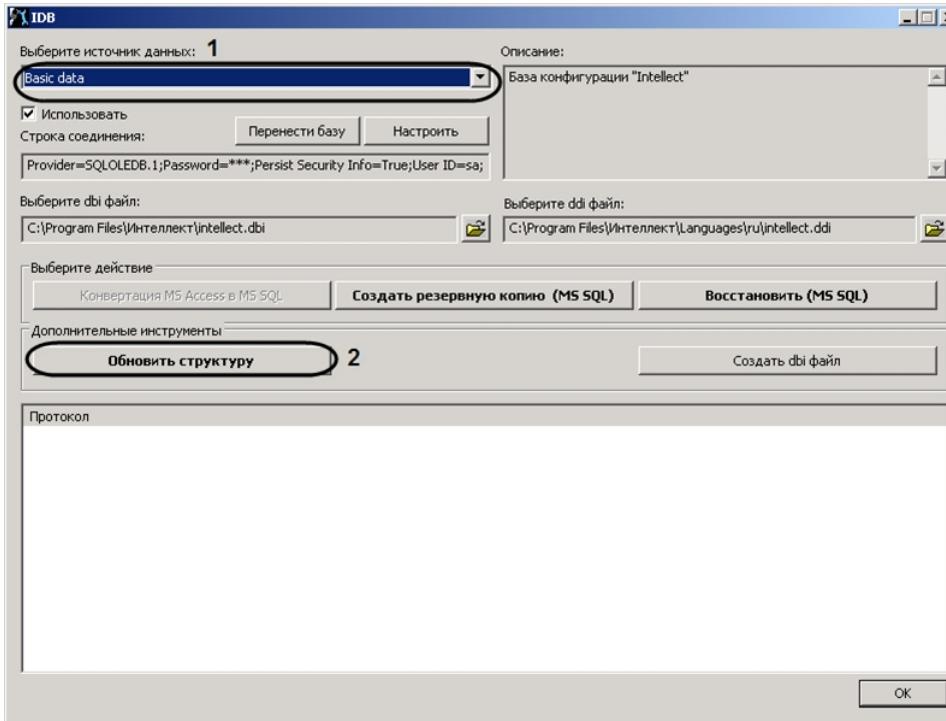
[OBJ_ZONE]
id, CHAR, 40
name, CHAR, 60
parent_id, CHAR, 40
Flags, INTEGER
guid, UNIQUEIDENTIFIER

[OBJ_DEMO]
id, CHAR, 16
name, CHAR, 60
parent_id, CHAR, 16
port, CHAR, 5
Flags, INTEGER

[OBJ_DEMO_DEVICE]
id, CHAR, 16
name, CHAR, 60
parent_id, CHAR, 16
address, INTEGER
Flags, INTEGER
```

3. intellect.dbi.

4. *idb.exe*.



5. : Basic data (1).

6. (2).

*idb.exe*.

The screenshot shows the 'IDB' application window with the 'Protokol' (Protocol) tab selected. The log window displays several SQL commands. A red oval highlights the first few commands related to table updates:

```
SQL (GO):IF (EXISTS(SELECT * FROM dbo.sysobjects WHERE id = OBJECT_ID(N[dbo].[OBJ_DEPARTMENT_WT_SCHEDULE])) AND xtype in (NU))  
ADD COLUMN(S)TO TABLE OBJ_DEMO  
guid  
ADD COLUMN(S)TO TABLE OBJ_DEMO_DEVICE  
guid  
Update [OBJPARAM] table
```

Below these, the log continues with more SQL statements, including file loading and object existence checks. At the bottom right of the log window, there is an 'OK' button.

7. *idb.exe*.

*Intellect*.