

# **HOME-PC\SQLEXPRESS**

## Servers

Servers 1

HOME-PC\SQLEXPRESS

# HOME-PC\SQLEXPRESS

## Databases ①

 AS2033

## Properties

Name	Value
Product	Microsoft SQL Server
Version Level	RTM
Edition	Express Edition (64-bit)
Version	15.0.2000.5
Language	Русский (Россия)
Collation	Cyrillic_General_CI_AS
Platform	NT x64
OS Version	6.3 (19042)
Processors	4
Physical Memory (Mb)	7543
Clustered	<b>False</b>
Root Directory	C:\Program Files\Microsoft SQL Server\MSSQL15.SQLEXPRESS\MSSQL

## Settings

Name	Value
Default data file path	C:\Program Files\Microsoft SQL Server\MSSQL15.SQLEXPRESS\MSSQL\DATA\
Default log file path	C:\Program Files\Microsoft SQL Server\MSSQL15.SQLEXPRESS\MSSQL\DATA\
Default backup file path	C:\Program Files\Microsoft SQL Server\MSSQL15.SQLEXPRESS\MSSQL\Backup
Recovery Interval (minutes)	0
Default index fill factor (%)	0
Default backup media retention (days)	0
Nested triggers enabled	<b>True</b>
Allow triggers to fire others	<b>True</b>
Default language	русский
Default fulltext language	русский
Default fulltext language LCID	1049
Network packet size	4096
Two digit year cutoff	2049
Max text replication size (bytes)	65536
Locks	0

Blocked process threshold	0
Cursor threshold	-1
Cost threshold for parallelism	5
Scans and runs startup procedures	<b>False</b>
Transform noise words	<b>False</b>
Filestream access level	Disabled
Optimize for 'ad hoc' workloads	<b>False</b>
Remote login timeout (sec)	10

## User databases

### User databases 1

 AS2033

# AS2033

## Description

AUMANU передовая информационная система

## Properties

Name	Value
SQL Server version	Microsoft SQL Server RTM 15.0.2000.5
Compatibility level	SQL Server 2019
Last backup time	N/A
Last backup time of transaction log	N/A
Default index fill factor (%)	0
Creation date	16.03.2023 15:04:22
Number of Users	6
Database Encryption Enabled	False
Database Encryption Algorithm	N/A
Database status	ONLINE
Max size	0 B
Database size (Mb)	1936
Unallocated data space (Mb)	194,19
Default language	N/A
Default fulltext language	русский

## Object Types ③

-  Tables
-  Views
-  Stored Procedures

## Options

Name	Value
Database collation	Cyrillic_General_CI_AS
Restrict access	MULTI_USER
Cleanly shutdown	False
Supplemental logging enabled	False
Is Read-Only	False
Auto close	False
Auto shrink	False
Database is read-only for restore log	False

Is Snapshot	False
Snapshot isolation state	False
Read committed snapshot on	False
Recovery model	SIMPLE
Page verify option	Checksum
Auto create statistics	True
Auto update statistics	True
Auto update statistics asynchronously	False
ANSI NULL default	False
ANSI NULL enabled	False
ANSI padding enabled	False
ANSI warnings enabled	False
Arithmetic abort enabled	False
Concatenating NULL yields NULL	False
Numeric roundabort enabled	False
Quoted Identifier ON	False
Recursive triggers enabled	False
Close cursors on commit	False
Local cursor by default	False
Fulltext enabled	True
Trustworthy	False
Database chaining	False
Forced parameterization	False
Master key encrypted by server	False
Published	False
Merge published	False
Is distribution database	False
Sync with backup	False
Service broker GUID	b849d7d5-869f-4cb3-b6ac-0d4443a092b1
Service broker enabled	False
Log reuse wait	Nothing
Date correlation	False
Change data capture enabled	False
Honor broker priority	False
Database owner	HOME-PC\Akbaa
Delayed durability	Disabled
Containment type	False

## Database Files ②

Name	Type	Size	Max size	Autogrowth	File Name
AS2033	Rows	204 800 KB	N/A	65 536 KB	C:\Program Files\Microsoft SQL Server\MSSQL15.SQLEXPRESS\MSSQL\DATA\AS2033.mdf
AS2033_log	Log	1 777 664 KB	2 147 483 648 KB	65 536 KB	C:\Program Files\Microsoft SQL Server\MSSQL15.SQLEXPRESS\MSSQL\DATA\AS2033_log.ldf

## Tables

### Objects 21

Name	Description
dbo.auth_group	
dbo.auth_group_permissions	
dbo.auth_permission	
dbo.auth_user	
dbo.auth_user_groups	
dbo.auth_user_user_permissions	
dbo.contractor	
dbo.djangoproject_admin_log	
dbo.djangoproject_content_type	
dbo.djangoproject_migrations	
dbo.djangoproject_session	
dbo.items	
dbo.machines	
dbo.machines_cache	
dbo.memec_userprofile	
dbo.messages	
dbo.orders	
dbo.product	
dbo.reports	
dbo.request	
dbo.state	

## dbo.auth\_group

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	3
Created	17.03.2023 9:32:29
Last Modified	17.03.2023 9:32:35

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	id	int	4	10	0	True	1 - 1			False	False	
FK	name	nvarchar	150	0	0	True				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description
PK	auth_group_name_a6ea08ec_uniq	name	True		
PK	PK_auth_gro_3213E83FDB2BD68D	id	True		

### SQL Script

```
CREATE TABLE dbo.auth_group (
    id int IDENTITY,
    name nvarchar(150) NOT NULL,
```

```
PRIMARY KEY CLUSTERED (id),
CONSTRAINT auth_group_name_a6ea08ec_uniq UNIQUE (name)
)
ON [PRIMARY]
GO
```

## Depends On ①



AS2033

## Used By ②

- [dbo.auth\\_group\\_permissions](#)
- [dbo.auth\\_user\\_groups](#)

## dbo.auth\_group\_permissions

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	0
Created	17.03.2023 9:32:29
Last Modified	17.03.2023 9:32:29

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	id	bigint	8	19	0	True	1 - 1			False	False	
CK	group_id	int	4	10	0	True				False	False	
CK	permission_id	int	4	10	0	True				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description
	auth_group_permissions_group_id_b120cbf9	group_id	False		
	auth_group_permissions_group_id_permission_id_0cd325b0_uniq	group_id, permission_id	True		
	auth_group_permissions_permission_id_84c5c92e	permission_id	False		

	PK_auth_group_3213E83FF19A68A8		id	True	
--	--------------------------------	--	----	------	--

## Foreign Keys

Name	Columns	Description
auth_group_permissions_group_id_b120cbf9_fk_auth_group_id	id	
auth_group_permissions_permission_id_84c5c92e_fk_auth_permission_id	id	

## SQL Script

```

CREATE TABLE dbo.auth_group_permissions (
    id bigint IDENTITY,
    group_id int NOT NULL,
    permission_id int NOT NULL,
    PRIMARY KEY CLUSTERED (id)
)
ON [PRIMARY]
GO

CREATE INDEX auth_group_permissions_group_id_b120cbf9
    ON dbo.auth_group_permissions (group_id)
ON [PRIMARY]
GO

CREATE UNIQUE INDEX auth_group_permissions_group_id_permission_id_0cd325b0_uniq
    ON dbo.auth_group_permissions (group_id, permission_id)
    WHERE ([group_id] IS NOT NULL AND [permission_id] IS NOT NULL)
ON [PRIMARY]
GO

CREATE INDEX auth_group_permissions_permission_id_84c5c92e
    ON dbo.auth_group_permissions (permission_id)
    ON [PRIMARY]
GO

ALTER TABLE dbo.auth_group_permissions
    ADD CONSTRAINT auth_group_permissions_group_id_b120cbf9_fk_auth_group_id FOREIGN KEY (group_id) REFERENCES dbo.
        auth_group (id)
GO

ALTER TABLE dbo.auth_group_permissions
    ADD CONSTRAINT auth_group_permissions_permission_id_84c5c92e_fk_auth_permission_id FOREIGN KEY (permission_id)
        REFERENCES dbo.auth_permission (id)
GO

```

## Depends On ②

- dbo.auth\_group
- dbo.auth\_permission

## Used By

No items found

## dbo.auth\_permission

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	28
Created	17.03.2023 9:32:29
Last Modified	17.03.2023 9:32:31

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	id	int	4	10	0	True	1 - 1			False	False	
	name	nvarchar	255	0	0	True				False	False	
CK	content_type_id	int	4	10	0	True				False	False	
FK	codename	nvarchar	100	0	0	True				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description
	auth_permission_content_type_id_2f476e4b	content_type_id	False		
	auth_permission_content_type_id_codename_01ab375a_uniq	content_type_id, codename	True		

	PK_auth_per_3213E83FB5FD663E	id	True		
--	------------------------------	----	------	--	--

## Foreign Keys

Name	Columns	Description
auth_permission_content_type_id_2f476e4b_fk_django_content_type_id	id	

## SQL Script

```

CREATE TABLE dbo.auth_permission (
    id int IDENTITY,
    name nvarchar(255) NOT NULL,
    content_type_id int NOT NULL,
    codename nvarchar(100) NOT NULL,
    PRIMARY KEY CLUSTERED (id)
)
ON [PRIMARY]
GO

CREATE INDEX auth_permission_content_type_id_2f476e4b
    ON dbo.auth_permission (content_type_id)
    ON [PRIMARY]
GO

CREATE UNIQUE INDEX auth_permission_content_type_id_codename_01ab375a_uniq
    ON dbo.auth_permission (content_type_id, codename)
    WHERE ([content_type_id] IS NOT NULL AND [codename] IS NOT NULL)
    ON [PRIMARY]
GO

ALTER TABLE dbo.auth_permission
    ADD CONSTRAINT auth_permission_content_type_id_2f476e4b_fk_django_content_type_id FOREIGN KEY (content_type_id)
        REFERENCES dbo.django_content_type (id)
GO

```

## Depends On ②

- [dbo.djangoproject\\_content\\_type](#)
- AS2033

## Used By ②

- [dbo.auth\\_group\\_permissions](#)
- [dbo.auth\\_user\\_user\\_permissions](#)

## dbo.auth\_user

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	2
Created	17.03.2023 9:32:29
Last Modified	18.03.2023 10:59:50

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	id	int	4	10	0	True	1 - 1			False	False	
	password	nvarchar	128	0	0	True				False	False	
	last_login	datetimeoffset	10	34	7	False				False	False	
	is_superuser	bit	1	1	0	True				False	False	
PK	username	nvarchar	150	0	0	True				False	False	
	first_name	nvarchar	150	0	0	True				False	False	
	last_name	nvarchar	150	0	0	True				False	False	
	email	nvarchar	254	0	0	True				False	False	
	is_staff	bit	1	1	0	True				False	False	
	is_active	bit	1	1	0	True				False	False	

	date_joined	datetimeoffset	10	34	7	True				False	False	
--	-------------	----------------	----	----	---	------	--	--	--	-------	-------	--

## Indexes

Key	Name	Columns	Unique	Type	Description
	auth_user_username_6821ab7c_uniq	username	True		
	PK_auth_use_3213E83FA93C0DDB	id	True		

## SQL Script

```

CREATE TABLE dbo.auth_user (
    id int IDENTITY,
    password nvarchar(128) NOT NULL,
    last_login datetimeoffset NULL,
    is_superuser bit NOT NULL,
    username nvarchar(150) NOT NULL,
    first_name nvarchar(150) NOT NULL,
    last_name nvarchar(150) NOT NULL,
    email nvarchar(254) NOT NULL,
    is_staff bit NOT NULL,
    is_active bit NOT NULL,
    date_joined datetimeoffset NOT NULL,
    PRIMARY KEY CLUSTERED (id),
    CONSTRAINT auth_user_username_6821ab7c_uniq UNIQUE (username)
)
ON [PRIMARY]
GO

```

## Depends On ①



AS2033

## Used By ⑤

- dbo.auth\_user\_groups
- dbo.auth\_user\_user\_permissions
- dbo.djangoproject\_log
- dbo.memec\_userprofile
- dbo.email\_to\_send\_get

## dbo.auth\_user\_groups

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	1
Created	17.03.2023 9:32:29
Last Modified	17.03.2023 9:32:29

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	id	bigint	8	19	0	True	1 - 1			False	False	
CK	user_id	int	4	10	0	True				False	False	
CK	group_id	int	4	10	0	True				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description
	auth_user_groups_group_id_97559544	group_id	False		
	auth_user_groups_user_id_6a12ed8b	user_id	False		
	auth_user_groups_user_id_group_id_94350c0c_uniq	user_id, group_id	True		

	PK__auth_use__3213E83F0C6B4381	id	True	
--	--------------------------------	----	------	--

## Foreign Keys

Name	Columns	Description
auth_user_groups_group_id_97559544_fk_auth_group_id	id	
auth_user_groups_user_id_6a12ed8b_fk_auth_user_id	id	

## SQL Script

```

CREATE TABLE dbo.auth_user_groups (
    id bigint IDENTITY,
    user_id int NOT NULL,
    group_id int NOT NULL,
    PRIMARY KEY CLUSTERED (id)
)
ON [PRIMARY]
GO

CREATE INDEX auth_user_groups_group_id_97559544
    ON dbo.auth_user_groups (group_id)
    ON [PRIMARY]
GO

CREATE INDEX auth_user_groups_user_id_6a12ed8b
    ON dbo.auth_user_groups (user_id)
    ON [PRIMARY]
GO

CREATE UNIQUE INDEX auth_user_groups_user_id_group_id_94350c0c_uniq
    ON dbo.auth_user_groups (user_id, group_id)
    WHERE ([user_id] IS NOT NULL AND [group_id] IS NOT NULL)
    ON [PRIMARY]
GO

ALTER TABLE dbo.auth_user_groups
    ADD CONSTRAINT auth_user_groups_group_id_97559544_fk_auth_group_id FOREIGN KEY (group_id) REFERENCES dbo.auth_group (id)
GO

ALTER TABLE dbo.auth_user_groups
    ADD CONSTRAINT auth_user_groups_user_id_6a12ed8b_fk_auth_user_id FOREIGN KEY (user_id) REFERENCES dbo.auth_user (id)
GO

```

## Depends On ②

- dbo.auth\_group
- dbo.auth\_user

## Used By

No items found

## dbo.auth\_user\_user\_permissions

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	0
Created	17.03.2023 9:32:29
Last Modified	17.03.2023 9:32:29

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	id	bigint	8	19	0	True	1 - 1			False	False	
CK	user_id	int	4	10	0	True				False	False	
CK	permission_id	int	4	10	0	True				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description
	auth_user_user_permissions_permission_id_1fbb5f2c	permission_id	False		
	auth_user_user_permissions_user_id_a95ead1b	user_id	False		
	auth_user_user_permissions_user_id_permission_id_14a6b632_uniq	user_id, permission_id	True		

	PK_auth_use_3213E83F30CE0ED5		id	True	
--	------------------------------	--	----	------	--

## Foreign Keys

Name	Columns	Description
auth_user_user_permissions_permission_id_1fbb5f2c_fk_auth_permission_id	id	
auth_user_user_permissions_user_id_a95ead1b_fk_auth_user_id	id	

## SQL Script

```

CREATE TABLE dbo.auth_user_user_permissions (
    id bigint IDENTITY,
    user_id int NOT NULL,
    permission_id int NOT NULL,
    PRIMARY KEY CLUSTERED (id)
)
ON [PRIMARY]
GO

CREATE INDEX auth_user_user_permissions_permission_id_1fbb5f2c
    ON dbo.auth_user_user_permissions (permission_id)
ON [PRIMARY]
GO

CREATE INDEX auth_user_user_permissions_user_id_a95ead1b
    ON dbo.auth_user_user_permissions (user_id)
    ON [PRIMARY]
GO

CREATE UNIQUE INDEX auth_user_user_permissions_user_id_permission_id_14a6b632_uniq
    ON dbo.auth_user_user_permissions (user_id, permission_id)
    WHERE ([user_id] IS NOT NULL AND [permission_id] IS NOT NULL)
    ON [PRIMARY]
GO

ALTER TABLE dbo.auth_user_user_permissions
    ADD CONSTRAINT auth_user_user_permissions_permission_id_1fbb5f2c_fk_auth_permission_id FOREIGN KEY (permission_id)
        REFERENCES dbo.auth_permission (id)
GO

ALTER TABLE dbo.auth_user_user_permissions
    ADD CONSTRAINT auth_user_user_permissions_user_id_a95ead1b_fk_auth_user_id FOREIGN KEY (user_id) REFERENCES dbo.
    auth_user (id)
GO

```

## Depends On ②

- dbo.auth\_permission
- dbo.auth\_user

## Used By

No items found

## dbo.contractor

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	20
Created	17.03.2023 10:51:29
Last Modified	17.03.2023 12:59:49

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	id	int	4	10	0	True				False	False	
	inn	varchar	50	0	0	False				False	False	
	caption	varchar	50	0	0	False				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description
PK	PK_contractor_id	id	True		

### SQL Script

```
CREATE TABLE dbo.contractor (
    id int NOT NULL,
    inn varchar(50) NULL,
```

```
caption varchar(50) NULL,  
CONSTRAINT PK_contractor_id PRIMARY KEY CLUSTERED (id)  
)  
ON [PRIMARY]  
GO
```

## Depends On ①



AS2033

## Used By ⑤

- [dbo.\\_request\\_items](#)
- [dbo.\\_requests](#)
- [dbo.\\_generate\\_order\\_report](#)
- [dbo.\\_json\\_contractors](#)
- [dbo.\\_json\\_requests](#)

## dbo.djangoproject\_log

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	12
Created	17.03.2023 9:32:29
Last Modified	17.03.2023 9:32:29

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	id	int	4	10	0	True	1 - 1			False	False	
	action_time	datetimeoffset	10	34	7	True				False	False	
	object_id	nvarchar		0	0	False				False	False	
	object_repr	nvarchar	200	0	0	True				False	False	
UQ	action_flag	smallint	2	5	0	True				False	False	
	change_message	nvarchar		0	0	True				False	False	
CK	content_type_id	int	4	10	0	False				False	False	
FK	user_id	int	4	10	0	True				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description
	django_admin_log_content_type_id_c4bce8eb	content_type_id	False		
	django_admin_log_user_id_c564eba6	user_id	False		
	PK_django_a_3213E83F0029CBFA	id	True		

## Check Constraints

Name	Columns	Condition	Description
django_admin_log_action_flag_a8637d59_check	action_flag	([action_flag]>=(0))	

## Foreign Keys

Name	Columns	Description
django_admin_log_content_type_id_c4bce8eb_fk_django_content_type_id	id	
django_admin_log_user_id_c564eba6_fk_auth_user_id	id	

## SQL Script

```

CREATE TABLE dbo.django_admin_log (
    id int IDENTITY,
    action_time datetimeoffset NOT NULL,
    object_id nvarchar(max) NULL,
    object_repr nvarchar(200) NOT NULL,
    action_flag smallint NOT NULL,
    change_message nvarchar(max) NOT NULL,
    content_type_id int NULL,
    user_id int NOT NULL,
    PRIMARY KEY CLUSTERED (id),
    CONSTRAINT django_admin_log_action_flag_a8637d59_check CHECK ([action_flag]>=(0))
)
ON [PRIMARY]
TEXTIMAGE_ON [PRIMARY]
GO

CREATE INDEX django_admin_log_content_type_id_c4bce8eb
    ON dbo.django_admin_log (content_type_id)
    ON [PRIMARY]
GO

CREATE INDEX django_admin_log_user_id_c564eba6
    ON dbo.django_admin_log (user_id)
    ON [PRIMARY]
GO

ALTER TABLE dbo.django_admin_log
    ADD CONSTRAINT django_admin_log_content_type_id_c4bce8eb_fk_django_content_type_id FOREIGN KEY (content_type_id)
        REFERENCES dbo.django_content_type (id)
GO

ALTER TABLE dbo.django_admin_log
    ADD CONSTRAINT django_admin_log_user_id_c564eba6_fk_auth_user_id FOREIGN KEY (user_id) REFERENCES dbo.auth_user (
    id)
GO

```

## Depends On ③

-  dbo.auth\_user
-  dbo.django\_content\_type
-  AS2033

## Used By

No items found

## dbo.django\_content\_type

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	7
Created	17.03.2023 9:32:29
Last Modified	17.03.2023 9:32:31

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	id	int	4	10	0	True	1 - 1			False	False	
PK	app_label	nvarchar	100	0	0	True				False	False	
PK	model	nvarchar	100	0	0	True				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description
	django_content_type_app_label_model_76bd3d3b_uniq	app_label, model	True		
PK	PK_django_c_3213E83F7CB4B6B2	id	True		

### SQL Script

```
CREATE TABLE dbo.django_content_type (
```

```
    id int IDENTITY,
    app_label nvarchar(100) NOT NULL,
    model nvarchar(100) NOT NULL,
PRIMARY KEY CLUSTERED (id)
)
ON [PRIMARY]
GO

CREATE UNIQUE INDEX django_content_type_app_label_model_76bd3d3b_uniq
    ON dbo.django_content_type (app_label, model)
    WHERE ([app_label] IS NOT NULL AND [model] IS NOT NULL)
ON [PRIMARY]
GO
```

### Depends On ①



AS2033

### Used By ②



[dbo.auth\\_permission](#)



[dbodjango\\_admin\\_log](#)

## dbo.djangoproject\_migrations

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	19
Created	17.03.2023 9:32:28
Last Modified	17.03.2023 9:32:28

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	id	bigint	8	19	0	True	1 - 1			False	False	
	app	nvarchar	255	0	0	True				False	False	
	name	nvarchar	255	0	0	True				False	False	
	applied	datetimeoffset	10	34	7	True				False	False	

### Indexes

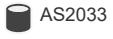
Key	Name	Columns	Unique	Type	Description
PK	PK_django_m_3213E83FBDA9EBC	id	True		

### SQL Script

```
CREATE TABLE dbo.djangoproject_migrations (
```

```
    id bigint IDENTITY,  
    app nvarchar(255) NOT NULL,  
    name nvarchar(255) NOT NULL,  
    applied datetimeoffset NOT NULL,  
    PRIMARY KEY CLUSTERED (id)  
)  
ON [PRIMARY]  
GO
```

## Depends On ①



## Used By

No items found

## dbo.djangoproject\_session

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	22
Created	17.03.2023 9:32:35
Last Modified	17.03.2023 9:32:35

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	session_key	nvarchar	40	0	0	True				False	False	
	session_data	nvarchar		0	0	True				False	False	
IF	expire_date	datetimeoffset	10	34	7	True				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description
	djangosession_expire_date_a5c62663	expire_date	False		
PK	PK_django_s_B3BA0F1FBCE9BEFE	session_key	True		

### SQL Script

```
CREATE TABLE dbo.djangosession (
```

```
session_key nvarchar(40) NOT NULL,
session_data nvarchar(max) NOT NULL,
expire_date datetimeoffset NOT NULL,
PRIMARY KEY CLUSTERED (session_key)
)
ON [PRIMARY]
TEXTIMAGE_ON [PRIMARY]
GO

CREATE INDEX django_session_expire_date_a5c62663
    ON dbo.django_session (expire_date)
    ON [PRIMARY]
GO
```

## Depends On ①



## Used By

No items found

## dbo.items

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	56
Created	17.03.2023 11:01:35
Last Modified	17.03.2023 15:32:12

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	id	int	4	10	0	True				False	False	
	request_id	int	4	10	0	False				False	False	
	product_id	int	4	10	0	False				False	False	
	quantity	int	4	10	0	False				False	False	
	quantityExec	int	4	10	0	False				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description
PK	PK_items_id	id	True		

### SQL Script

```
CREATE TABLE dbo.items (
    id int NOT NULL,
    request_id int NULL,
    product_id int NULL,
    quantity int NULL,
    quantityExec int NULL,
    CONSTRAINT PK_items_id PRIMARY KEY CLUSTERED (id)
)
ON [PRIMARY]
GO
```

## Depends On

No items found

## Used By 9

-  [dbo.\\_items\\_production\\_by\\_time](#)
-  [dbo.\\_request\\_items](#)
-  [dbo.\\_requests](#)
-  [dbo.\\_check\\_for\\_approve](#)
-  [dbo.\\_check\\_for\\_broken](#)
-  [dbo.\\_generate\\_order\\_report](#)
-  [dbo.\\_json\\_request\\_items](#)
-  [dbo.\\_set\\_order](#)
-  [dbo.\\_set\\_request\\_approve](#)

## dbo.machines

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	9
Created	17.03.2023 11:11:48
Last Modified	18.03.2023 18:30:36

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	label	varchar	50	0	0	True				False	False	
	type	varchar	50	0	0	False				False	False	
	port	int	4	10	0	False				False	False	
	current_status	varchar	50	0	0	False				False	False	
	report_id	int	4	10	0	False				False	False	
	product_id	int	4	10	0	False				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description
PK	PK_machines_label	label	True		

## SQL Script

```
CREATE TABLE dbo.machines (
    label varchar(50) NOT NULL,
    type varchar(50) NULL,
    port int NULL,
    current_status varchar(50) NULL,
    report_id int NULL,
    product_id int NULL,
    CONSTRAINT PK_machines_label PRIMARY KEY CLUSTERED (label)
)
ON [PRIMARY]
GO
```

## Depends On ①



AS2033

## Used By ⑫

- [dbo.orders](#)
- [dbo.\\_messages](#)
- [dbo.\\_reports](#)
- [dbo.\\_request\\_items](#)
- [dbo.\\_check\\_for\\_broken](#)
- [dbo.\\_clear\\_orders](#)
- [dbo.\\_generate\\_order\\_report](#)
- [dbo.\\_generate\\_repairs\\_report](#)
- [dbo.\\_json\\_machine\\_status](#)
- [dbo.\\_json\\_machines](#)
- [dbo.\\_set\\_order](#)
- [dbo.\\_set\\_request\\_approve](#)

## dbo.machines\_cache

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	9
Created	17.03.2023 19:07:17
Last Modified	18.03.2023 15:23:58

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	label	varchar	50	0	0	True				False	False	
	type	varchar	50	0	0	False				False	False	
	port	int	4	10	0	False				False	False	
	current_status	varchar	50	0	0	False				False	False	
	report_id	int	4	10	0	False				False	False	
	product_id	int	4	10	0	False				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description
PK	PK_machines_cache_label	label	True		

## SQL Script

```
CREATE TABLE dbo.machines_cache (
    label varchar(50) NOT NULL,
    type varchar(50) NULL,
    port int NULL,
    current_status varchar(50) NULL,
    report_id int NULL,
    product_id int NULL,
    CONSTRAINT PK_machines_cache_label PRIMARY KEY CLUSTERED (label)
)
ON [PRIMARY]
GO
```

## Depends On ①



## Used By ②

- [dbo.\\_check\\_for\\_broken](#)
- [dbo.\\_json\\_machines](#)

## dbo.memec\_userprofile

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	1
Created	18.03.2023 10:59:50
Last Modified	18.03.2023 10:59:50

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
.PrimaryKey	id	bigint	8	19	0	True	1 - 1			False	False	
	teleграмм	nvarchar	50	0	0	True				False	False	
CompositeKey	user_id	int	4	10	0	True				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description
PrimaryKey	PK_memec_us_3213E83F9E50B700	id	True		
CompositeKey	UQ_memec_us_B9BE370EF3772CE2	user_id	True		

### Foreign Keys

Name	Columns	Description
memec_userprofile_user_id_d0e839a4_fk_auth_user_id	id	

## SQL Script

```
CREATE TABLE dbo.memec_userprofile (
    id bigint IDENTITY,
    telegrann nvarchar(50) NOT NULL,
    user_id int NOT NULL,
    PRIMARY KEY CLUSTERED (id),
    UNIQUE (user_id)
)
ON [PRIMARY]
GO

ALTER TABLE dbo.memec_userprofile
ADD CONSTRAINT memec_userprofile_user_id_d0e839a4_fk_auth_user_id FOREIGN KEY (user_id) REFERENCES dbo.auth_user
(id)
GO
```

## Depends On ②



## Used By

No items found

## dbo.messages

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	52
Created	17.03.2023 11:51:00
Last Modified	18.03.2023 13:35:36

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	id	int	4	10	0	True	1 - 1			False	False	
	msg	varchar		0	0	False				False	False	
	viewed	bit	1	1	0	False			(0)	False	False	
	ts	datetime	8	23	3	False			(getdate())	False	False	
	type	int	4	10	0	False				False	False	
	send	bit	1	1	0	False			(0)	False	False	
	report_id	int	4	10	0	False				False	False	
	send_tg	bit	1	1	0	False			(0)	False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description

	PK_messages_id	id	True
--	----------------	----	------

## SQL Script

```
CREATE TABLE dbo.messages (
    id int IDENTITY,
    msg varchar(max) NULL,
    viewed bit NULL DEFAULT (0),
    ts datetime NULL DEFAULT (getdate()),
    type int NULL,
    send bit NULL DEFAULT (0),
    report_id int NULL,
    send_tg bit NULL DEFAULT (0),
    CONSTRAINT PK_messages_id PRIMARY KEY CLUSTERED (id)
)
ON [PRIMARY]
TEXTIMAGE_ON [PRIMARY]
GO
```

## Depends On 1



AS2033

## Used By 5

- [dbo.\\_messages](#)
- [dbo.\\_check\\_for\\_broken](#)
- [dbo.\\_send\\_message](#)
- [dbo.email\\_sended](#)
- [dbo.email\\_to\\_send\\_get](#)

## dbo.orders

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	0
Created	17.03.2023 14:02:12
Last Modified	18.03.2023 18:30:36

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	id	int	4	10	0	True	1 - 1			False	False	
	item_id	varchar	50	0	0	False				False	False	
C	label	varchar	50	0	0	False				False	False	
	quantity	int	4	10	0	False				False	False	
	priority	int	4	10	0	False				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description
PK	PK_orders_id	id	True		

### Foreign Keys

Name	Columns	Description
FK_orders_label	label	

## SQL Script

```

CREATE TABLE dbo.orders (
    id int IDENTITY,
    item_id varchar(50) NULL,
    label varchar(50) NULL,
    quantity int NULL,
    priority int NULL
    CONSTRAINT PK_orders_id PRIMARY KEY CLUSTERED (id)
)
ON [PRIMARY]
GO

ALTER TABLE dbo.orders
ADD CONSTRAINT FK_orders_label FOREIGN KEY (label) REFERENCES dbo.machines (label)
GO

```

## Depends On ②

-  dbo.machines
-  AS2033

## Used By ⑩

-  dbo.\_free\_machines
-  dbo.\_items\_production\_by\_time
-  dbo.\_request\_items
-  dbo.\_requests
-  dbo.\_check\_for\_approve
-  dbo.\_check\_for\_broken
-  dbo.\_clear\_orders
-  dbo.\_generate\_order\_report
-  dbo.\_set\_order
-  dbo.\_set\_request\_approve

## dbo.product

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	1895
Created	17.03.2023 11:03:13
Last Modified	17.03.2023 15:56:09

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	id	int	4	10	0	True				False	False	
	code	varchar	50	0	0	False				False	False	
	caption	varchar	500	0	0	False				False	False	
	millingTime	int	4	10	0	False				False	False	
	latheTime	int	4	10	0	False				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description
PK	PK_product_id	id	True		

### SQL Script

```
CREATE TABLE dbo.product (
    id int NOT NULL,
    code varchar(50) NULL,
    caption varchar(500) NULL,
    millingTime int NULL,
    latheTime int NULL,
    CONSTRAINT PK_product_id PRIMARY KEY CLUSTERED (id)
)
ON [PRIMARY]
GO
```

## Depends On ①



AS2033

## Used By ⑥

- [dbo.\\_items\\_production\\_by\\_time](#)
- [dbo.\\_request\\_items](#)
- [dbo.\\_automatic\\_distribution\\_machines](#)
- [dbo.\\_generate\\_order\\_report](#)
- [dbo.\\_json\\_products](#)
- [dbo.\\_json\\_request\\_items](#)

## dbo.reports

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	56
Created	17.03.2023 9:43:52
Last Modified	17.03.2023 14:10:03

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	id	int	4	10	0	True	1 - 1			False	False	
	label	varchar	255	0	0	False				False	False	
	title	varchar	255	0	0	False				False	False	
	body	varchar	255	0	0	False				False	False	
	description	varchar	0	0	0	False				False	False	
	user_name	varchar	50	0	0	False				False	False	
	html	varchar	0	0	0	False				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description

	PK_reports_id	id	True		
--	---------------	----	------	--	--

## SQL Script

```
CREATE TABLE dbo.reports (
    id int IDENTITY,
    label varchar(255) NULL,
    title varchar(255) NULL,
    body varchar(255) NULL,
    description varchar(max) NULL,
    user_name varchar(50) NULL,
    html varchar(max) NULL,
    CONSTRAINT PK_reports_id PRIMARY KEY CLUSTERED (id)
)
ON [PRIMARY]
TEXTIMAGE_ON [PRIMARY]
GO
```

## Depends On 1



AS2033

## Used By 3

[dbo.\\_reports](#)[dbo.\\_generate\\_order\\_report](#)[dbo.\\_generate\\_repairs\\_report](#)

## dbo.request

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	8
Created	17.03.2023 10:56:24
Last Modified	17.03.2023 13:07:35

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	id	int	4	10	0	True				False	False	
	number	varchar	50	0	0	False				False	False	
	date	datetime	8	23	3	False				False	False	
	description	varchar	1000	0	0	False				False	False	
	releaseDate	datetime	8	23	3	False				False	False	
	state	varchar	50	0	0	False				False	False	
	contractor_id	int	4	10	0	False				False	False	
	report_id	int	4	10	0	False				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description

	PK_request_id			
		id	True	

## SQL Script

```
CREATE TABLE dbo.request (
    id int NOT NULL,
    number varchar(50) NULL,
    date datetime NULL,
    description varchar(1000) NULL,
    releaseDate datetime NULL,
    state varchar(50) NULL,
    contractor_id int NULL,
    report_id int NULL,
    CONSTRAINT PK_request_id PRIMARY KEY CLUSTERED (id)
)
ON [PRIMARY]
GO
```

## Depends On 1



AS2033

## Used By 9

- [dbo\\_free\\_machines](#)
- [dbo\\_request\\_items](#)
- [dbo\\_requests](#)
- [dbo\\_requests\\_automatic\\_distribution\\_machines](#)
- [dbo\\_check\\_for\\_approve](#)
- [dbo\\_check\\_for\\_broken](#)
- [dbo\\_generate\\_order\\_report](#)
- [dbo\\_json\\_requests](#)
- [dbo\\_set\\_request\\_approve](#)

## dbo.state

### Description

### Properties

Name	Value
Collation	
File Group	
Filestream File Group	
Is Partitioned	False
Partitioned Column	
Partition Scheme	
File Groups	
Filestream Partition Scheme	
Filestream File Groups	
Heap	False
Full Text Catalog	
Full Text	
Compression	NONE
Row Count (~)	9
Created	17.03.2023 10:55:18
Last Modified	17.03.2023 13:05:00

### Columns

Key	Name	Data Type	Length	Precision	Scale	Not Null	Identity	Rule	Default	Computed	Persisted	Description
PK	code	varchar	50	0	0	True				False	False	
	caption	varchar	50	0	0	False				False	False	

### Indexes

Key	Name	Columns	Unique	Type	Description
PK	PK_state_code	code	True		

### SQL Script

```

CREATE TABLE dbo.state (
    code varchar(50) NOT NULL,
    caption varchar(50) NULL,
    CONSTRAINT PK_state_code PRIMARY KEY CLUSTERED (code)
)

```

```
ON [PRIMARY]  
GO
```

## Depends On ①



## Used By

No items found

## Views

### Objects 7

Name	Description
dbo_free_machines	
dbo_items_production_by_time	
dbo_messages	
dbo_reports	
dbo_request_items	
dbo_requests	
dbo_requests_automatic_distribution_machines	

## dbo.\_free\_machines

### Description

### Properties

Name	Value
Collation	Cyrillic_General_CI_AS
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Schema Bound	False
Created	17.03.2023 19:54:14
Last Modified	18.03.2023 19:52:43

### Columns

Key	Name	Description
	label	
	type_caption	
	type	
	port	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER VIEW dbo._free_machines
AS SELECT
    m.label
    ,CASE m.type
        WHEN 'milling' THEN 'фрезерный'
        WHEN 'lathe' THEN 'токарный'
    END AS type_caption
    ,type
    ,m.port
FROM machines m
LEFT JOIN orders o
    ON m.label = o.label
LEFT JOIN items i
    ON o.item_id = i.id
LEFT JOIN request r
    ON i.request_id = r.id
WHERE m.current_status IN ('WAITING', 'unknown')
AND ISNULL(r.state, '') NOT IN ('DRAFT', 'IN_PRODUCTION', 'unknown')
GO

```

### Depends On ②

-  dbo.orders
-  dbo.request

### Used By ②



dbo.\_automatic\_distribution\_machines



dbo.\_check\_for\_broken

## dbo.\_items\_production\_by\_time

### Description

### Properties

Name	Value
Collation	
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Schema Bound	False
Created	18.03.2023 12:35:55
Last Modified	18.03.2023 15:27:48

### Columns

Key	Name	Description
	id	
	request_id	
	product_id	
	quantity	
	all_milling_time	
	all_lathe_time	
	cur_quant_milling	
	cur_quant_lathe	
	demand_milling_quant	
	demand_lathe_quant	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER VIEW dbo._items_production_by_time
AS SELECT
    i.id
    ,i.request_id
    ,i.product_id
    ,i.quantity
    ,ISNULL(i.quantity,0) * ISNULL(p.millingTime,0) AS all_milling_time
    ,ISNULL(i.quantity,0) * ISNULL(p.latheTime,0) AS all_lathe_time
    ,CASE WHEN ISNULL(p.millingTime,0) > 0 THEN ISNULL(o.quantity,0) ELSE 0 END AS cur_quant_milling
    ,CASE WHEN ISNULL(p.latheTime,0) > 0 THEN ISNULL(o.quantity,0) ELSE 0 END AS cur_quant_lathe
    ,CASE WHEN ISNULL(p.millingTime,0) > 0 THEN ISNULL(i.quantity,0) - ISNULL(o.quantity,0) - ISNULL(i.quantityExec,0)
    ELSE 0 END AS demand_milling_quant
    ,CASE WHEN ISNULL(p.latheTime,0) > 0 THEN ISNULL(i.quantity,0) - ISNULL(o.quantity,0) - ISNULL(i.quantityExec,0)
    ELSE 0 END AS demand_lathe_quant
FROM items i
LEFT JOIN product p

```

```
    ON i.product_id = p.id  
LEFT JOIN orders o ON o.item_id = i.id  
GO
```

## Depends On ③

-  [dbo.items](#)
-  [dbo.product](#)
-  [dbo.orders](#)

## Used By ②

-  [dbo.\\_requests\\_automatic\\_distribution\\_machines](#)
-  [dbo.\\_automatic\\_distribution\\_machines](#)

## dbo.\_messages

### Description

### Properties

Name	Value
Collation	Cyrillic_General_CI_AS
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Schema Bound	False
Created	18.03.2023 12:01:01
Last Modified	18.03.2023 18:40:48

### Columns

Key	Name	Description
	id	
	msg	
	viewed	
	ts	
	type	
	send	
	report_id	
	port	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER VIEW dbo._messages
AS SELECT m.id
      ,m.msg
      ,m.viewed
      ,m.ts
      ,m.type
      ,m.send
      ,m.report_id
      ,ISNULL(CAST(m1.port AS VARCHAR(50)), '') AS port
  FROM messages m
  LEFT JOIN machines m1 ON m.report_id = m1.report_id
GO

```

### Depends On ②

-  dbo.messages
-  dbo.machines

## Used By

No items found

## dbo.\_reports

### Description

### Properties

Name	Value
Collation	Cyrillic_General_CI_AS
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Schema Bound	False
Created	18.03.2023 17:39:59
Last Modified	18.03.2023 18:41:00

### Columns

Key	Name	Description
	id	
	label	
	title	
	body	
	description	
	user_name	
	html	
	port	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER VIEW dbo._reports
AS SELECT
    r.id
    ,r.label
    ,r.title
    ,r.body
    ,r.description
    ,r.user_name
    ,r.html
    ,isnull(cast(m1.port as varchar(50)), '') AS port
FROM reports r
LEFT JOIN machines m1
    ON r.id = m1.report_id
GO

```

### Depends On ②

-  dbo.reports
-  dbo.machines

## Used By

No items found

## dbo.\_request\_items

### Description

### Properties

Name	Value
Collation	Cyrillic_General_CI_AS
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Schema Bound	False
Created	17.03.2023 17:39:13
Last Modified	18.03.2023 19:13:45

### Columns

Key	Name	Description
	request_id	
	number	
	date	
	description	
	releaseDate	
	contractor_caption	
	quantity	
	product_code	
	product_caption	
	item_id	
	machines_list	
	quantity_left_lathe	
	quantity_left_milling	
	state_order	
	state_order_int	

### SQL Script

```
SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER VIEW dbo._request_items
AS SELECT
    r.id AS request_id
    ,r.number
    ,r.date
    ,r.description
```

```

,r.releaseDate
,c.caption AS contractor_caption
,i.quantity
,p.code AS product_code
,p.caption AS product_caption
,i.id AS item_id
,ot.machines_list
,i.quantity-isnull(q_all_lathe,0) AS quantity_left_lathe
,i.quantity-isnull(q_all_milling,0) AS quantity_left_milling

,CASE
WHEN r.state = 'DRAFT' THEN CASE
    WHEN quant.i_quantity <= quant.o_quantity THEN 'Все распределено'
    WHEN quant.i_quantity > quant.o_quantity AND quant.o_quantity>0 THEN 'Частичное распределение'
    ELSE ''
END
ELSE ''
END AS state_order

,CASE
WHEN r.state = 'DRAFT' AND
    quant.i_quantity <= quant.o_quantity THEN 1
    ELSE 0
END AS state_order_int

FROM request r
LEFT JOIN items i
ON i.request_id = r.id
LEFT JOIN product p
ON i.product_id = p.id
LEFT JOIN contractor c
ON r.contractor_id = c.id
OUTER APPLY (SELECT
    o.item_id
    ,STRING_AGG(o.label + ' ' + CAST(o.quantity AS VARCHAR) + ' шт',CHAR(13)) AS machines_list
    FROM orders o
    GROUP BY o.item_id
    HAVING o.item_id = i.id) AS ot

OUTER APPLY (SELECT
    sum(isnull(o.quantity,0)) AS q_all_lathe
    FROM orders o
    inner JOIN machines m ON o.label = m.label and type='lathe'
    GROUP BY o.item_id
    HAVING o.item_id = i.id) AS q_ot_lathe

OUTER APPLY (SELECT
    sum(isnull(o.quantity,0)) AS q_all_milling
    FROM orders o
    inner JOIN machines m ON o.label = m.label and type='milling'
    GROUP BY o.item_id
    HAVING o.item_id = i.id) AS q_ot_milling

OUTER APPLY (SELECT
    SUM(ISNULL(i.quantity, 0)) AS i_quantity
    ,SUM(ISNULL(o.quantity, 0)) AS o_quantity
    FROM items i
    LEFT JOIN orders o
    ON o.item_id = i.id
    WHERE i.request_id = r.id) quant
GO

```

## Depends On 6

- dbo.request
- dbo.items
- dbo.product
- dbo.contractor
- dbo.orders
- dbo.machines

## Used By

No items found

No items found

No items found

## dbo.\_requests

### Description

### Properties

Name	Value
Collation	Cyrillic_General_CI_AS
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Schema Bound	False
Created	17.03.2023 12:43:31
Last Modified	18.03.2023 14:22:13

### Columns

Key	Name	Description
	request_id	
	number	
	date	
	description	
	releaseDate	
	state	
	state_caption	
	contractor_inn	
	contractor_caption	
	report_id	
	i_quantity	
	o_quantity	
	state_order	
	state_order_int	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER VIEW dbo._requests
AS SELECT
    r.id AS request_id
    ,r.number
    ,r.date
    ,r.description
    ,r.releaseDate
    ,s.code AS state

```

```
,s.caption AS state_caption  
,c.inn AS contractor_inn  
,c.caption AS contractor_caption  
  
,r.report_id  
,quant.i_quantity,quant.o_quantity  
  
,CASE  
WHEN r.state = 'DRAFT' THEN CASE  
WHEN quant.i_quantity <= quant.o_quantity THEN 'Все распределено'  
WHEN quant.i_quantity > quant.o_quantity AND quant.o_quantity>0 THEN 'Частичное распределение'  
ELSE ''  
END  
ELSE ''  
END AS state_order  
  
,CASE  
WHEN r.state = 'DRAFT' AND  
quant.i_quantity <= quant.o_quantity THEN 1  
ELSE 0  
END AS state_order_int  
  
FROM request r  
LEFT JOIN contractor c  
ON r.contractor_id = c.id  
LEFT JOIN state s  
ON r.state = s.code  
OUTER APPLY (SELECT  
SUM(ISNULL(i.quantity, 0)) AS i_quantity  
,SUM(ISNULL(o.quantity, 0)) AS o_quantity  
FROM items i  
LEFT JOIN orders o  
ON o.item_id = i.id  
WHERE i.request_id = r.id) quant  
GO
```

#### Depends On 4

-  dbo.request
-  dbo.contractor
-  dbo.items
-  dbo.orders

#### Used By

No items found

## dbo.\_requests\_automatic\_distribution\_machines

### Description

### Properties

Name	Value
Collation	
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Schema Bound	False
Created	18.03.2023 13:03:08
Last Modified	18.03.2023 15:29:38

### Columns

Key	Name	Description
	id	
	days	
	date	
	releaseDate	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER VIEW dbo._requests_automatic_distribution_machines
AS SELECT
    r.id
    ,sss.all_time / 86400 AS days
    ,r.date
    ,r.releaseDate
    FROM request r
    LEFT JOIN (SELECT ipbt.request_id, SUM(ipbt.all_miling_time) + SUM(ipbt.all_lathe_time) AS all_time FROM
    _items_production_by_time ipbt
    GROUP BY ipbt.request_id) AS sss ON sss.request_id = r.id

    WHERE
    --NOT EXISTS (SELECT
    -- 1
    -- FROM orders r
    -- LEFT JOIN items i
    -- ON i.id = r.item_id
    -- WHERE i.request_id = r.id)
    --AND
    r.state = 'DRAFT'
GO

```

### Depends On ②

-  dbo.request
-  dbo.\_items\_production\_by\_time

Used By ①



dbo\_automatic\_distribution\_machines

## Programmability

### Objects ①

 Stored Procedures

## Stored Procedures

### Objects 17

Name	Description
dbo._automatic_distribution_machines	
dbo._check_for_approve	
dbo._check_for_broken	
dbo._clear_orders	
dbo._generate_order_report	
dbo._generate_repairs_report	
dbo._json_contractors	
dbo._json_machine_status	
dbo._json_machines	
dbo._json_products	
dbo._json_request_items	
dbo._json_requests	
dbo._send_message	
dbo._set_order	
dbo._set_request_approve	
dbo.email_sended	
dbo.email_to_send_get	

## dbo.\_automatic\_distribution\_machines

### Description

### Properties

Name	Value
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Execute As	
Assembly	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER PROCEDURE dbo._automatic_distribution_machines
AS
    SET NOCOUNT ON

    DECLARE @requests TABLE (
        id INT IDENTITY (1, 1)
        ,requests_id INT
        ,items_id INT
        ,quantity INT
        -- ,millingTime INT --фрезерный
        -- ,latheTime INT --токарный
        ,demand_milling_quant INT
        ,demand_lathe_quant INT
        ,priority INT
    )

    IF (SELECT
            COUNT(*)
        FROM _free_machines)
    = 0
    BEGIN
        EXEC _send_message @msg = 'В текущий момент нет свободных станков!'
        ,@type = 1
        ,@report_id = NULL
        RETURN 0
    END

    --получаем список не распределенных заказов
    INSERT @requests (requests_id, items_id, quantity,-- millingTime, latheTime,
    demand_milling_quant, demand_lathe_quant, priority)
    SELECT
        ss.id
        ,i.id AS items_id
        ,0
        --,p.millingTime
        --,p.latheTime
        ,i.demand_milling_quant
        ,i.demand_lathe_quant
        ,CASE
            WHEN ss.days > 2 AND
                ss.date = (SELECT
                    MIN(date)
                FROM _requests_automatic_distribution_machines
                WHERE days > 2) THEN 1
            WHEN ss.days > 2 AND
                ss.date = (SELECT
                    MIN(date)
                FROM _requests_automatic_distribution_machines
                WHERE days > 2) THEN 1
        END
    
```

```

ss.releaseDate = (SELECT
    MIN(releaseDate)
    FROM _requests_automatic_distribution_machines
    WHERE days > 2) THEN 2
WHEN ss.days < 2 AND
    ss.date = (SELECT
        MIN(date)
        FROM _requests_automatic_distribution_machines
        WHERE days < 2) THEN 3
WHEN ss.days < 2 AND
    ss.releaseDate = (SELECT
        MIN(releaseDate)
        FROM _requests_automatic_distribution_machines
        WHERE days < 2) THEN 4
ELSE 5
END
FROM _requests_automatic_distribution_machines AS ss
LEFT JOIN _items_production_by_time i
    ON i.request_id = ss.id
LEFT JOIN product p
    ON i.product_id = p.id
WHERE ss.days > 2
ORDER BY CASE
    WHEN ss.days > 2 AND
        ss.date = (SELECT
            MIN(date)
            FROM _requests_automatic_distribution_machines
            WHERE days > 2) THEN 1
    WHEN ss.days > 2 AND
        ss.releaseDate = (SELECT
            MIN(releaseDate)
            FROM _requests_automatic_distribution_machines
            WHERE days > 2) THEN 2
    WHEN ss.days < 2 AND
        ss.date = (SELECT
            MIN(date)
            FROM _requests_automatic_distribution_machines
            WHERE days < 2) THEN 3
    WHEN ss.days < 2 AND
        ss.releaseDate = (SELECT
            MIN(releaseDate)
            FROM _requests_automatic_distribution_machines
            WHERE days < 2) THEN 4
    ELSE 5
END

SELECT
    *
FROM @requests

IF (SELECT
    COUNT(*)
    FROM @requests)
    = 0
BEGIN
    EXEC _send_message @msg = 'Нет заявок для автораспределения!'
                ,@type = 1
                ,@report_id = NULL
    RETURN 0
END

DECLARE @current_str INTEGER
    ,@max_str INTEGER

DECLARE @items_id INT
    ,@quantity INT
    ,@demand_milling_quant INT
    ,@demand_lathe_quant INT
    ,@port INT
    ,@milingTime INT
    ,@latheTime INT
    ,@priority INT

SELECT
    @current_str = MIN(id)
    ,@max_str = MAX(id)

```

```

    FROM @requests

    WHILE @current_str <= @max_str
    BEGIN
        IF (SELECT
                COUNT(*)
            FROM _free_machines)
        = 0
        BEGIN
            EXEC _send_message @msg = 'Распределение завершено. Все станки в работе!'
                ,@type = 1
                ,@report_id = NULL
            BREAK
        END

        SELECT
            @items_id = items_id
            ,@quantity = quantity
            -- ,@millingTime = millingTime
            -- ,@latheTime = latheTime
            ,@demand_lathe_quant = ISNULL(demand_lathe_quant, 0)
            ,@demand_milling_quant = ISNULL(demand_milling_quant, 0)
            ,@priority = priority
        FROM @requests
        WHERE id = @current_str

        IF ISNULL(@latheTime, 0) > 0
        AND @demand_lathe_quant > 0
        BEGIN
            IF EXISTS (SELECT TOP 1
                        1
                    FROM _free_machines
                    WHERE type = 'lathe') -- Токарный
            BEGIN
                SET @port = (SELECT TOP 1
                            port
                        FROM _free_machines
                        WHERE type = 'lathe')
                EXEC _set_order @item_id = @items_id
                                ,@port = @port
                                ,@quantity = @demand_lathe_quant
                                ,@priority = @priority
            END
        END

        IF ISNULL(@latheTime, 0) > 0
        AND @demand_milling_quant > 0
        BEGIN
            IF EXISTS (SELECT TOP 1
                        1
                    FROM _free_machines
                    WHERE type = 'milling') -- фрезейрный
            BEGIN
                SET @port = (SELECT TOP 1
                            port
                        FROM _free_machines
                        WHERE type = 'milling')
                EXEC _set_order @item_id = @items_id
                                ,@port = @port
                                ,@quantity = @demand_milling_quant
                                ,@priority = @priority
            END
        END

        SET @current_str = @current_str + 1
    END
    RETURN 0
GO

```

Depends On 6



-  [dbo.\\_send\\_message](#)
-  [dbo.\\_requests\\_automatic\\_distribution\\_machines](#)
-  [dbo.\\_items\\_production\\_by\\_time](#)
-  [dbo.product](#)
-  [dbo.\\_set\\_order](#)

## Used By

No items found

## dbo.\_check\_for\_approve

### Description

### Properties

Name	Value
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Execute As	
Assembly	

### Parameters

Name	Data Type	Length	Description
@request_id	int	4	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER PROCEDURE dbo._check_for_approve @request_id INT
AS
BEGIN
    DECLARE @r INT = 1
    ,@q1 INT, @q2 INT
    --проверим что у этого заказа статус
    IF NOT EXISTS (SELECT
        id
        FROM request r
        WHERE r.state IN ('DRAFT'))
    --
    SET @r = 0
    --проверим, что количество
    SELECT @q1=sum(i.quantity) from request r
    LEFT JOIN items i ON i.request_id=r.id
    WHERE r.id=@request_id
    SELECT @q2=
    SUM( o.quantity)
    FROM items i
    LEFT JOIN orders o
    ON o.item_id = i.id
    WHERE i.request_id = @request_id
    --если распределено q2 больше или равно, то пойдет
    IF @q1>@q2
        SET @r=0
    SELECT
        @r AS result
    END
    GO

```

### Depends On ③





dbo.items



dbo.orders

## Used By

No items found

## dbo.\_check\_for\_broken

### Description

### Properties

Name	Value
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Execute As	
Assembly	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER PROCEDURE dbo._check_for_broken
AS
BEGIN
    --находит те заказы, в которых в заказ нарядах есть станки, у которых сейчас статус СЛОМАН или РЕМОНТ
    DECLARE @label VARCHAR(50)
        ,@label_new VARCHAR(50)
        ,@msg VARCHAR(255)
        ,@report_id INT
        ,@var_report_id INT
        ,@var_request_id INT
        ,@product_id INT
        ,@var_number VARCHAR(50)

    DECLARE @machines TABLE (
        port VARCHAR(50)
        ,label VARCHAR(50)
        ,type VARCHAR(50)
        ,product_id VARCHAR(50)
        ,mark VARCHAR(50)
    )

    --станки, которые стали сломанными
    INSERT INTO @machines (port, label, type, product_id, mark)
        SELECT
            m.port
            ,m.label
            ,m.type
            ,m.product_id
            ,CASE--надо ему искать замену или нет
                WHEN m_prev.product_id IS NOT NULL AND
                    m_prev.current_status = 'WORKING' THEN 'BROKEN'
                ELSE 'NEED REPLACE'
            END AS mark
        FROM machines_cache m_prev
        LEFT JOIN machines m
        ON m_prev.label = m.label

    /* OUTER APPLY (SELECT SUM(orders.quantity) AS s_q FROM orders LEFT JOIN items i ON i.id = orders.item_id LEFT JOIN
    request r ON r.id = i.request_id WHERE r.state IN ('DRAFT', 'IN_PRODUCTION') AND orders.label = m_prev.label) o*/
        WHERE m.current_status IN ('BROKEN', 'REPAIRING')
        AND m_prev.current_status NOT IN ('BROKEN', 'REPAIRING')

    -- SELECT * ,1' FROM @machines

    WHILE EXISTS (SELECT TOP 1

```

```
1
FROM @machines
WHERE mark IN ('NEED REPLACE', 'BROKEN')
BEGIN
--по очереди все обрабатываем сломавшиеся
SELECT TOP 1
@label = label
FROM @machines
WHERE mark IN ('NEED REPLACE', 'BROKEN')

SET @label_new = NULL

IF @label = ''
BREAK

--ищем подходящую замену, если нужно
SELECT
@label_new = fm.label
FROM @machines m
INNER JOIN _free_machines fm
ON m.type = fm.type
AND m.label = @label
AND m.mark IN ('NEED REPLACE')

--SELECT @label label ,@label_new label_new

IF ISNULL(@label_new, '') = ''
BEGIN
--создадим сообщение
EXEC _generate_repairs_report @label
,@report_id OUTPUT

IF EXISTS (SELECT
label
FROM @machines
WHERE label = @label
AND mark = 'NEED REPLACE')
SET @msg = 'Станок ' + @label + ' сломан. На замену никто не назначен. Сформирована заявка на ремонт'
ELSE
SET @msg = 'Станок ' + @label + ' сломан. Сформирована заявка на ремонт'

EXEC _send_message @msg
,2 --красный
,@report_id
END
ELSE
BEGIN
--у замены нужно поменять статус и продукт_ид
UPDATE machines
SET product_id = @product_id
,current_status = 'WORKING'
WHERE label = @label_new

--у старого кстати очистим продукт
UPDATE machines
SET product_id = NULL
WHERE label = @label

--перевесим все невыполненные наряды
UPDATE orders
SET label = @label_new
WHERE label = @label
AND item_id IN (SELECT
i.id AS item_id
FROM request r
LEFT JOIN items i
ON i.request_id = r.id
WHERE r.state IN ('DRAFT', 'IN_PRODUCTION', 'unknown'))

--по идее можно обновить заказ наряд.
DECLARE cur CURSOR FAST_FORWARD READ_ONLY LOCAL FOR SELECT

r.report_id
,r.id
,r.number
FROM request r
```

```

LEFT JOIN items i
    ON i.request_id = r.id
LEFT JOIN orders o
    ON o.item_id = i.id
WHERE label = @label
AND r.state IN ('IN_PRODUCTION')

OPEN cur

FETCH NEXT FROM cur INTO @var_report_id, @var_request_id, @var_number

WHILE @@FETCH_STATUS = 0
BEGIN

--по этому报 оуту уберем в msg
UPDATE messages
SET report_id = NULL
    ,msg = msg + ' - Аннулирован в связи с поломкой станка'
WHERE report_id = @var_report_id

--создадим сообщение
EXEC _generate_order_report @var_request_id
                                ,@report_id OUTPUT

SET @msg = 'В связи с заменой оборудования переформирован заказ-наряд на заказ ' + @var_number

EXEC _send_message @msg
                    ,0 --зеленый
                    ,@report_id

FETCH NEXT FROM cur INTO @var_report_id, @var_request_id, @var_number

END

CLOSE cur
DEALLOCATE cur

--добавляем в список тех, кого в реестре напинают
INSERT INTO @machines (port, label, type, product_id, mark)
SELECT
    m.port
    ,m.label
    ,m.type
    ,m.product_id
    ,'NEW WORKING'
FROM machines m
WHERE m.label = @label_new
AND m.product_id IS NOT NULL

--создадим сообщение
EXEC _generate_repairs_report @label
                                ,@report_id OUTPUT

SET @msg = 'Станок ' + @label + ' сломан, в замену ему назначен ' + @label_new + '. Сформирована заявка на ремонт'
EXEC _send_message @msg
                    ,2 --красный
                    ,@report_id

END

--обрабатываем удалаем
DELETE FROM @machines
WHERE label = @label

END

SELECT
    port
    ,label
    ,product_id
    ,mark
FROM @machines
WHERE mark = 'NEW WORKING'

```

END  
GO

## Depends On 10

-  [dbo.machines\\_cache](#)
-  [dbo.machines](#)
-  [dbo.\\_free\\_machines](#)
-  [dbo.\\_generate\\_repairs\\_report](#)
-  [dbo.\\_send\\_message](#)
-  [dbo.orders](#)
-  [dbo.request](#)
-  [dbo.items](#)
-  [dbo.messages](#)
-  [dbo.\\_generate\\_order\\_report](#)

## Used By

No items found

## dbo.\_clear\_orders

### Description

### Properties

Name	Value
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Execute As	
Assembly	

### Parameters

Name	Data Type	Length	Description
@item_id	int	4	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER PROCEDURE dbo._clear_orders
@item_id INT
AS
SET NOCOUNT ON

UPDATE machines
SET current_status='WAITING'
WHERE current_status='WORKING' AND label IN (
SELECT o.label FROM orders o WHERE item_id=@item_id)

DELETE FROM orders
WHERE item_id=@item_id
GO

```

### Depends On ②

-  dbo.machines
-  dbo.orders

### Used By

No items found

## dbo.\_generate\_order\_report

### Description

### Properties

Name	Value
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Execute As	
Assembly	

### Parameters

Name	Data Type	Length	Description
@request_id	int	4	
@report_id	int	4	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER PROCEDURE dbo._generate_order_report @request_id INT, @report_id INT OUTPUT
AS

DECLARE @number VARCHAR(50)
,@date VARCHAR(10)
,@description VARCHAR(MAX)
,@releaseDate DATETIME
,@contragent_name VARCHAR(50)
,@inn VARCHAR(500)
,@contragent_caption VARCHAR(500)
,@quantity_all VARCHAR(50) = CAST(100 AS VARCHAR)

,@current_str INT
,@max_str INT

DECLARE @order_table TABLE (
    id INT IDENTITY (1, 1)
    ,machine_name VARCHAR(50)
    ,operation_type VARCHAR(50)
    ,quantity INT
    ,sort INT
)
INSERT @order_table (machine_name, operation_type, quantity)
SELECT
    m.label
    ,m.type
    ,o.quantity
FROM orders o
LEFT JOIN machines m
    ON o.label = m.label
ORDER BY m.type

```

```

DECLARE @items_table TABLE(
    id INT IDENTITY (1, 1)
    ,code VARCHAR(50)
    ,caption VARCHAR(50)
    ,quantity_product VARCHAR(50)
    ,str VARCHAR(500)
)

INSERT @items_table (code, caption, quantity_product, str)
SELECT
    p.code
    ,p.caption
    ,i.quantity
    ,ot.ssss
FROM items i
LEFT JOIN product p
    ON i.product_id = p.id
OUTER APPLY (SELECT
    o.item_id
    ,STRING_AGG(o.label + ' ' + CAST(o.quantity AS VARCHAR), '<br>') AS ssss
    FROM orders o
    GROUP BY o.item_id
    HAVING o.item_id = i.id) AS ot
WHERE i.request_id = @request_id

SELECT
    @date = ISNULL(CONVERT(VARCHAR(10), r.date, 104), '')
    ,@number = ISNULL(r.number, '')
    ,@description = ISNULL(r.description, '')
    ,@contragent_name = ISNULL(c.caption, '')
    ,@inn = ISNULL(c.inn, '')
FROM dbo.request AS r
LEFT JOIN dbo.contractor AS c
    ON c.id = r.contractor_id
WHERE r.id = @request_id

SET @quantity_all = ISNULL(CAST((SELECT
    SUM(CAST(quantity_product AS INT))
    FROM @items_table)
    AS VARCHAR), '')

DECLARE @html VARCHAR(MAX) = '

Заказ наряд на выполнение






```

```

<table style="border-collapse: collapse; width: 100%;" border="1">
<tbody>
<tr>
<td style="width: 25.0%; text-align: center;"><strong>Код продукта</strong></td>
<td style="width: 25.0%; text-align: center;"><strong>Наименование</strong></td>
<td style="width: 25.0%; text-align: center;"><strong>Количество</strong></td>
<td style="width: 25.0%; text-align: center;"><strong>По станкам</strong></td>
</tr>'



SELECT
    @current_str = MIN(id)
    ,@max_str = MAX(id)
FROM @items_table

DECLARE @code VARCHAR(50)
    ,@caption_product VARCHAR(50)
    ,@quantity_product VARCHAR(50)
    ,@str VARCHAR(500)
WHILE @current_str <= @max_str
BEGIN
SET @html = @html + '<tr>

SELECT
    @code = ISNULL(it.code, '')
    ,@caption_product = ISNULL(it.caption, '')
    ,@quantity_product = ISNULL(CAST(it.quantity_product AS VARCHAR), '')
    ,@str = ISNULL(it.str,'')
FROM @items_table AS it
WHERE it.id = @current_str
SET @html = @html
SET @html = @html + '<td style="width: 25.0%; text-align: center;">' + @code + '</td>
+ '<td style="width: 25.0%; text-align: center;">' + @caption_product + '</td>
+ '<td style="width: 25.0%; text-align: center;">' + @quantity_product + '</td>
+ '<td style="width: 25.0%; text-align: center;">' + @str + '</td>'
SET @html = @html + '</tr>'
SET @current_str = @current_str + 1
END

--@КОД
--@наименование
--@количество

SET @html = @html +
'</tbody>
</table>'

-- SET @html = @html +
-- -- SELECT
-- --     @current_str = MIN(id)
-- --     ,@max_str = MAX(id)
-- -- FROM @order_table

-- -- DECLARE @machine_label VARCHAR(50)
-- --     ,@quantity_machine VARCHAR(50)

-- -- WHILE @current_str <= @max_str
-- -- BEGIN
-- --     SELECT
-- --         @machine_label = ISNULL(ot.machine_name, '')
-- --         ,@quantity_machine = ISNULL(CAST(ot.quantity AS VARCHAR), '')
-- --     FROM @order_table AS ot
-- --     WHERE ot.id = @current_str

-- --     SET @html = @html +
-- --     SET @html = @html +

```

```
-- SET @current_str = @current_str + 1  
-- END  
-- SET @html = @html + '  
--  
--
```

Станок	Количество
' + @machine_label + '	' + @quantity_machine + '

```
SET @html = @html + '<p></p>  
<p style="text-align: left;">Подпись_____</p>'  
  
INSERT reports (title, html)  
VALUES ('Заказ-наряд на ' + ISNULL(@number, ''), @html);  
  
SET @report_id = @@IDENTITY;  
  
UPDATE request  
SET report_id = @report_id  
WHERE id = @request_id  
GO
```

## Depends On 7

- dbo.orders
- dbo.machines
- dbo.items
- dbo.product
- dbo.request
- dbo.contractor
- dbo.reports

## Used By 2

- dbo.\_check\_for\_broken
- dbo.\_set\_request\_approve

## dbo.\_generate\_repairs\_report

### Description

### Properties

Name	Value
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Execute As	
Assembly	

### Parameters

Name	Data Type	Length	Description
@label	varchar	50	
@report_id	int	4	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER PROCEDURE dbo._generate_repairs_report
@label VARCHAR(50) = NULL,
@report_id INT OUTPUT
AS
DECLARE @date VARCHAR(10) = CONVERT(VARCHAR(10), CAST(GETDATE() AS DATE), 104)

    ,@description VARCHAR(500)
IF @description IS NULL
    SET @description = 'Прошу службу главного механика срочно произвести ремонт оборудования!'

DECLARE @html VARCHAR(MAX) = '<p><strong>Заявка на ремонт станка</strong></p>
<table style="border-collapse: collapse; width: 100%;" border="1">
<tbody>
<tr>
<td style="width: 30%;"><strong>Станок</strong></td>
<td style="width: 70%;">' + @label + '</td>
</tr>
<tr>
<td style="width: 30%;"><strong>Дата заявки</strong></td>
<td style="width: 70%;">' + @date + '</td>
</tr>
</tbody>
</table>
<br>
<p>' + @description + '</p>
<p> </p>
<p style="text-align: left;">Подпись_____</p>

INSERT reports (title, html)
VALUES ('Заявка на ремонт станка ' + ISNULL(@label, ''), @html);
SET @report_id = @@IDENTITY;

UPDATE machines
SET report_id = @report_id
WHERE label = @label'

```

**Depends On ②**

-  dbo.reports
-  dbo.machines

**Used By ①**

-  dbo.\_check\_for\_broken

## dbo.\_json\_contractors

### Description

### Properties

Name	Value
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Execute As	
Assembly	

### Parameters

Name	Data Type	Length	Description
@json	varchar	-1	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER PROCEDURE dbo._json_contractors @json VARCHAR(MAX)
AS
BEGIN
    DECLARE @r INT

    /* IF @json IS NULL SET @json = '[{"id":1,"inn":"7451356475","caption":"000\"КомпозитГруппЧелябинск\""}, {"id":2,"inn":"1903011647","caption":"000\"Фактор\""}, {"id":3,"inn":"7801621813","caption":"000\"Союзпромтех\""}, {"id":4,"inn":"6732236252","caption":"000\"Барса\""}, {"id":5,"inn":"6658397081","caption":"000\"Гаммаинпром\""}, {"id":6,"inn":"5037011495","caption":"000\"НПКИнновация\""}, {"id":7,"inn":"7325114941","caption":"000\"СП\""}, {"id":8,"inn":"7810184969","caption":"000\"Фабрик\""}, {"id":9,"inn":"7727204754","caption":"000\"Оконные Системы Технологии\""}, {"id":10,"inn":"1006001845","caption":"ПК\"Маяк\""}, {"id":11,"inn":"5021010457","caption":"000\"ПТХ\""}, {"id":12,"inn":"6664022972","caption":"АО\"Промбезопасность\""}, {"id":13,"inn":"1650013048","caption":"ПИТК\"Приоритет\""}, {"id":14,"inn":"3665051138","caption":"000\"Техконтроль\""}, {"id":15,"inn":"5401278725","caption":"000\"Пищетогр\""}, {"id":16,"inn":"4712124451","caption":АО\"Приозерская Горная Компания\""}, {"id":17,"inn":"3327837307","caption":"000\"Элитком\""}, {"id":18,"inn":"5814100370","caption":"000\"Фабрика Интерьерной Ковки\""}, {"id":19,"inn":"4813014303","caption":"000\"НПКЭкотекс\""}, {"id":20,"inn":"5260368853","caption":"000\"Традиции\""}] */ */

    TRUNCATE TABLE contractor;

    INSERT INTO contractor (id, inn, caption)
    SELECT
        id,
        inn, caption
    FROM OPENJSON(@json)
    WITH (
        id INT 'strict $.id',
        inn varchar(50) 'strict $.inn',
        caption varchar(50) 'strict $.caption'
    );
    SET @r = @@ROWCOUNT
    select @r AS result
END
GO

```

## Depends On ①

 dbo.contractor

## Used By

No items found

## dbo.\_json\_machine\_status

### Description

### Properties

Name	Value
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Execute As	
Assembly	

### Parameters

Name	Data Type	Length	Description
@port	int	4	
@json	varchar	-1	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER PROCEDURE dbo._json_machine_status @port INT, @json VARCHAR(MAX)
AS
BEGIN
    DECLARE @r INT

    /*IF @json IS NULL BEGIN SET @port =1045 SET @json = '{"id":22,"code":"mm-4","state":
    {"code":"WORKING","caption":"Используется"}, "advInfo":{"advInfo":{"productId":"40"}}, "beginDateTime":"2023-03-
    17T14:26:02.075271","endDateTime":null}' END */
    UPDATE machines
    SET current_status = state_code
    FROM (SELECT
        state.state_code
        FROM OPENJSON(@json)
        WITH (
            id INT '$.id',
            code VARCHAR(50) '$.code',
            state NVARCHAR(MAX) '$.state' AS JSON,
            advInfo NVARCHAR(MAX) '$.advInfo' AS JSON,
            beginDateTime VARCHAR(50) '$.beginDateTime',
            endDateTime VARCHAR(50) '$.endDateTime'
        )
        OUTER APPLY OPENJSON(state)
        WITH (state_code VARCHAR(50) '$.code') AS state
    /* OUTER APPLY OPENJSON(advInfo) WITH (productId VARCHAR(50) '$.productId') AS advInfo */
    ) S
    WHERE port = @port

    SET @r = @@ROWCOUNT
    SELECT
        @r AS result
    -- 
    END
GO

```

### Depends On ①

## Used By

No items found

## dbo.\_json\_machines

### Description

### Properties

Name	Value
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Execute As	
Assembly	

### Parameters

Name	Data Type	Length	Description
@json	varchar	-1	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER PROCEDURE dbo._json_machines @json VARCHAR(MAX)
AS
BEGIN
    DECLARE @r INT

    /* IF @json IS NULL SET @json = '{"milling":{"mm-4":1045,"mm-3":1043,"mm-2":1042,"mm-1":1041,"mm-5":1047}, "lathe": {"lm-2":1052,"lm-3":1054,"lm-1":1051,"lm-4":1056}}' */

    DECLARE @m TABLE (label varchar(50), report_id INT)

    --закешируем текущий стак репортов
    INSERT INTO @m (label, report_id)
    SELECT m.label, m.report_id FROM machines m

    --сохраним что было
    TRUNCATE TABLE machines_cache;
    INSERT INTO machines_cache (label, type, port, current_status, report_id)
    SELECT
        label
        ,type
        ,port
        ,current_status
        ,report_id
    FROM machines

    TRUNCATE TABLE machines;

    INSERT INTO machines (label, port, type)
    SELECT
        mill.[key]
        ,mill.value
        ,'milling' AS type
    FROM OPENJSON(@json)
    WITH (
        milling NVARCHAR(MAX) '$.milling' AS JSON
    )
    OUTER APPLY OPENJSON(milling) AS mill;
    SET @r = @@ROWCOUNT

```

```
INSERT INTO machines (label, port, type)
SELECT
    lathe.[key]
    ,lathe.value
    ,'lathe' AS type
FROM OPENJSON(@json)
WITH (
    lathe NVARCHAR(MAX) '$.lathe' AS JSON
)
OUTER APPLY OPENJSON(lathe) AS lathe
;

SET @r = @r + @@ROWCOUNT

--ВОССТАНОВИМ СТАК репортов
UPDATE machines
SET report_id=[@m].report_id
FROM @m
WHERE [@m].label=machines.label

SELECT
    @r AS result
END
GO
```

## Depends On ②

-  dbo.machines
-  dbo.machines\_cache

## Used By

No items found

## dbo.\_json\_products

### Description

### Properties

Name	Value
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Execute As	
Assembly	

### Parameters

Name	Data Type	Length	Description
@json	varchar	-1	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER PROCEDURE dbo._json_products @json VARCHAR(MAX)
AS
BEGIN
    DECLARE @r INT
    /* IF @json IS NULL SET @json = '[{"id":40,"code":"ML81LT143","caption":"ØP81-TK143","millingTime":81,"latheTime":143},
    {"id":4,"code":"ML28LT27","caption":"ØP28-TK27","millingTime":28,"latheTime":27},
    {"id":41,"code":"ML80LT79","caption":"ØP80-TK79","millingTime":80,"latheTime":79},
    {"id":5,"code":"ML57LT88","caption":"ØP57-TK88","millingTime":57,"latheTime":88},
    {"id":6,"code":"ML96LT145","caption":"ØP96-TK145","millingTime":96,"latheTime":145},
    {"id":115,"code":"ML55LT40","caption":"ØP55-TK40","millingTime":55,"latheTime":40},
    {"id":42,"code":"ML108LT14","caption":"ØP108-TK14","millingTime":108,"latheTime":14},
    {"id":7,"code":"ML105LT38","caption":"ØP105-TK38","millingTime":105,"latheTime":38},
    {"id":564,"code":"ML30LT132","caption":"ØP30-TK132","millingTime":30,"latheTime":132},
    {"id":8,"code":"ML105LT93","caption":"ØP105-TK93","millingTime":105,"latheTime":93},
    {"id":88,"code":"ML22LT53","caption":"ØP22-TK53","millingTime":22,"latheTime":53},
    {"id":43,"code":"ML117LT35","caption":"ØP117-TK35","millingTime":117,"latheTime":35},
    {"id":9,"code":"ML21LT111","caption":"ØP21-TK111","millingTime":21,"latheTime":111}] ' */
    TRUNCATE TABLE product;

    INSERT INTO product (id, code, caption, millingTime, latheTime)
    SELECT
        id
        ,code
        ,caption
        ,millingTime
        ,latheTime
    FROM OPENJSON(@json)
    WITH (
        id INT '$.id',
        code VARCHAR(50) '$.code',
        caption VARCHAR(50) '$.caption',
        millingTime INT '$.millingTime',
        latheTime INT '$.latheTime'
    );
    SET @r = @@ROWCOUNT
    SELECT
        @r AS result
    --

```

END  
GO

## Depends On ①

 dbo.product

## Used By

No items found

## dbo.\_json\_request\_items

### Description

### Properties

Name	Value
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Execute As	
Assembly	

### Parameters

Name	Data Type	Length	Description
@request_id	int	4	
@json	varchar	-1	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER PROCEDURE dbo._json_request_items @request_id INT, @json VARCHAR(MAX)
AS
BEGIN
    DECLARE @r INT

    /* IF @json IS NULL BEGIN SET @request_id = 324 SET @json = '[{"id":2057,"request":{"id":2057,"number":"2023-03-17-319","date":"2023-03-17","contractor":{"id":1,"inn":"7451356475","caption":"ООО\КомпозитГруппЧелябинск\"}, "description":"","state":{"code":"DRAFT","caption":"Формируется"},"releaseDate":"2023-03-23"}, "product":{"id":1929,"code":"ML34LT88","caption":"ФР34-TK88","millingTime":34,"latheTime":88}, "quantity":932,"quantityExec":0}, {"id":2058,"request":{"id":2058,"number":"2023-03-17-319","date":"2023-03-17","contractor":{"id":1,"inn":"7451356475","caption":"ООО\КомпозитГруппЧелябинск\"}, "description":"","state":{"code":"DRAFT","caption":"Формируется"},"releaseDate":"2023-03-23"}, "product":{"id":1930,"code":"ML29LT100","caption":"ФР29-TK100","millingTime":29,"latheTime":100}, "quantity":139,"quantityExec":0}, {"id":2059,"request":{"id":2059,"number":"2023-03-17-319","date":"2023-03-17","contractor":{"id":1,"inn":"7451356475","caption":"ООО\КомпозитГруппЧелябинск\"}, "description":"","state":{"code":"DRAFT","caption":"Формируется"},"releaseDate":"2023-03-23"}, "product":{"id":1931,"code":"ML110LT22","caption":"ФР110-TK22","millingTime":110,"latheTime":22}, "quantity":276,"quantityExec":0}, {"id":2060,"request":{"id":2060,"number":"2023-03-17-319","date":"2023-03-17","contractor":{"id":1,"inn":"7451356475","caption":"ООО\КомпозитГруппЧелябинск\"}, "description":"","state":{"code":"DRAFT","caption":"Формируется"},"releaseDate":"2023-03-23"}, "product":{"id":1932,"code":"ML26LT105","caption":"ФР26-TK105","millingTime":26,"latheTime":105}, "quantity":645,"quantityExec":0}, {"id":2054,"request":{"id":2054,"number":"2023-03-17-319","date":"2023-03-17","contractor":{"id":1,"inn":"7451356475","caption":"ООО\КомпозитГруппЧелябинск\"}, "description":"","state":{"code":"DRAFT","caption":"Формируется"},"releaseDate":"2023-03-23"}, "product":{"id":1926,"code":"ML26LT146","caption":"ФР26-TK146","millingTime":26,"latheTime":146}, "quantity":539,"quantityExec":0}, {"id":2055,"request":{"id":2055,"number":"2023-03-17-319","date":"2023-03-17","contractor":{"id":1,"inn":"7451356475","caption":"ООО\КомпозитГруппЧелябинск\"}, "description":"","state":{"code":"DRAFT","caption":"Формируется"},"releaseDate":"2023-03-23"}, "product":{"id":1927,"code":"ML32LT86","caption":"ФР32-TK86","millingTime":32,"latheTime":86}, "quantity":800,"quantityExec":0}, {"id":2056,"request":{"id":2056,"number":"2023-03-17-319","date":"2023-03-17","contractor":{"id":1,"inn":"7451356475","caption":"ООО\КомпозитГруппЧелябинск\"}, "description":"","state":{"code":"DRAFT","caption":"Формируется"},"releaseDate":"2023-03-23"}, "product":{"id":1928,"code":"ML51LT138","caption":"ФР51-TK138","millingTime":51,"latheTime":138}, "quantity":727,"quantityExec":0}]] END */
    --
    DELETE FROM items
    WHERE request_id=@request_id

```

```
INSERT INTO items(id, request_id, product_id, quantity, quantityExec)
SELECT
    id
    ,@request_id AS request_id
    ,product_id
    ,quantity
    ,quantityExec
FROM OPENJSON(@json)
WITH (
    id INT '$.id',
    product NVARCHAR(MAX) '$.product' AS JSON,
    quantity INT '$.quantity',
    quantityExec INT '$.quantityExec'
)
OUTER APPLY OPENJSON(product)
WITH (product_id INT '$.id') AS product;

SET @r = @@ROWCOUNT
SELECT
    @r AS result
--
```

END  
GO

## Depends On ②

-  dbo.items
-  dbo.product

## Used By

No items found

## dbo.\_json\_requests

### Description

### Properties

Name	Value
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Execute As	
Assembly	

### Parameters

Name	Data Type	Length	Description
@json	varchar	-1	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER PROCEDURE dbo._json_requests @json VARCHAR(MAX)
AS
BEGIN
    DECLARE @r INT
    /*IF @json IS NULL SET @json = '[{"id":324,"number":"2023-03-17-319","date":"2023-03-17","contractor":
    {"id":1,"inn":"7451356475","caption":"ООО \"КомпозитГруппЧелябинск\""}, "description":"","state":
    {"code":"DRAFT","caption":"Формируется"}, "releaseDate":"2023-03-23"}, {"id":325,"number":"2023-03-17-320","date":"2023-
    03-17","contractor":{"id":2,"inn":"1903011647","caption":"ООО \"Фактор\""}, "description":"","state":
    {"code":"DRAFT","caption":"Формируется"}, "releaseDate":"2023-03-26"}, {"id":321,"number":"2023-03-17-316","date":"2023-
    03-17","contractor":{"id":6,"inn":"5037011495","caption":"ООО \"НПКИнновация\""}, "description":"","state":
    {"code":"DRAFT","caption":"Формируется"}, "releaseDate":"2023-03-25"}, {"id":319,"number":"2023-03-12-314","date":"2023-
    03-12","contractor":{"id":7,"inn":"7325114941","caption":"ООО \"СП\""}, "description":"","state":
    {"code":"DRAFT","caption":"Формируется"}, "releaseDate":"2023-03-15"}, {"id":320,"number":"2023-03-12-315","date":"2023-
    03-12","contractor":{"id":11,"inn":"5021010457","caption":"ООО \"ПТХ\""}, "description":"","state":
    {"code":"DRAFT","caption":"Формируется"}, "releaseDate":"2023-03-14"}, {"id":322,"number":"2023-03-17-317","date":"2023-
    03-17","contractor":{"id":18,"inn":"5814100370","caption":"ООО \"Фабрика Интерьерной Ковки\""}, "description":"","state":
    {"code":"DRAFT","caption":"Формируется"}, "releaseDate":"2023-03-22"}, {"id":323,"number":"2023-03-17-318","date":"2023-
    03-17","contractor":{"id":19,"inn":"4813014303","caption":"ООО \"НПКЭкотекс\""}, "description":"","state":
    {"code":"DRAFT","caption":"Формируется"}, "releaseDate":"2023-03-26"}]' */
    --
    TRUNCATE TABLE request;

    INSERT INTO request (id, number, date, description, releaseDate, state, contractor_id)
    SELECT
        id
        ,number
        ,CONVERT(DATETIME, date, 20) AS date
        ,description
        ,CONVERT(DATETIME, releaseDate, 20) AS releaseDate
        ,st.code
        ,contractor_id
    FROM OPENJSON(@json)
    WITH (
        id INT '$.id',
        number VARCHAR(50) '$.number',
        date VARCHAR(50) '$.date',
        description VARCHAR(1000) '$.description',
        releaseDate VARCHAR(50) '$.releaseDate',
        --
    )

```

```
state NVARCHAR(MAX) '$.state' AS JSON,
contractor NVARCHAR(MAX) '$.contractor' AS JSON
)
OUTER APPLY OPENJSON(state)
WITH (code NVARCHAR(50) '$.code',
caption NVARCHAR(50) '$.caption') AS st

OUTER APPLY OPENJSON(contractor)
WITH (contractor_id NVARCHAR(50) '$.id',
inn NVARCHAR(50) '$.inn', caption NVARCHAR(50) '$.caption') AS contr
;

SET @r = @@ROWCOUNT
SELECT
    @r AS result
-- END
GO
```

## Depends On ②

-  dbo.request
-  dbo.contractor

## Used By

No items found

## dbo.\_send\_message

### Description

### Properties

Name	Value
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Execute As	
Assembly	

### Parameters

Name	Data Type	Length	Description
@msg	varchar	-1	
@type	int	4	
@report_id	int	4	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER PROCEDURE dbo._send_message
    @msg varchar(max) = NULL
    ,@type int = NULL
    ,@report_id INT=NULL
AS
--0 зеленый
--1 желтый
--2 красный
    INSERT messages (msg, type, report_id)
    VALUES (@msg,@type,@report_id)
GO

```

### Depends On ①



dbo.messages

### Used By ③

-  dbo.\_automatic\_distribution\_machines
-  dbo.\_check\_for\_broken
-  dbo.\_set\_request\_approve

## dbo.\_set\_order

### Description

### Properties

Name	Value
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Execute As	
Assembly	

### Parameters

Name	Data Type	Length	Description
@item_id	int	4	
@port	int	4	
@quantity	int	4	
@priority	int	4	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER PROCEDURE dbo._set_order @item_id INT,
@port INT,
@quantity INT,
@priority INT = NULL
AS
BEGIN
    SET NOCOUNT ON

    --устанавливает заказ наряд
    --проверяет, чтобы не назначилось лишнего
    --проверить статус станка в таблице waitung or анон и еще тип
    --количество в quantitiExce общая сумма уже назначена

    DECLARE @type VARCHAR(50)
        ,@state VARCHAR(50)
        ,@error_message VARCHAR(500) = NULL

    --получаем тип станка и его статус
    SELECT
        @type = m.type
        ,@state = m.current_status
    FROM machines m
    WHERE m.port = @port
    --Типы станков
    --lathe
    --milling

    DECLARE @quantity_items INT --нужно сделать всего
        ,@quantityExec_items INT --уже сделано
        ,@quantity_distribute_items INT
        ,@quantity_current_orders INT --делается сейчас по заявкам

    --сколько сейчас делается текущей детали на станках

```

```

SELECT
    @quantity_current_orders = SUM(isnull(o.quantity,0))
FROM orders o
LEFT JOIN machines m
    ON o.label = m.label
WHERE m.type = @type
AND o.item_id = @item_id

SET @quantity_current_orders=isnull(@quantity_current_orders,0)
--проверяем сколько в заявке и сколько уже сделано
SELECT
    @quantity_items = i.quantity
    ,@quantityExec_items = i.quantityExec
FROM items i
WHERE i.id = @item_id

--сколько осталось распределить
SET @quantity_distribute_items = @quantity_items - @quantityExec_items - @quantity_current_orders

IF @quantity_distribute_items < @quantity
BEGIN
    SET @error_message = 'Заведено неверное количество '+cast(@quantity AS varchar(50))+''!
    GOTO ERROR
END

IF @quantity > @quantity_distribute_items
    SET @quantity = @quantity_distribute_items

--проверяю что станок не занят
IF (@state = 'unknown')
    OR (@state = 'WAITING')
BEGIN
    --вставляем ордер
    INSERT orders (item_id, label, quantity, priority)
        SELECT
            @item_id
            ,label
            ,@quantity
            ,@priority
        FROM machines m
        WHERE m.port = @port;
    SET @error_message = ''
    GOTO ERROR
END
ELSE
BEGIN
    SET @error_message = 'Станок занят!'
    GOTO ERROR
END

ERROR:
BEGIN
    SELECT
        @error_message AS msg
END
END
GO

```

## Depends On ③

-  dbo.machines
-  dbo.orders
-  dbo.items

## Used By ①

-  dbo.\_automatic\_distribution\_machines

## dbo.\_set\_request\_approve

### Description

### Properties

Name	Value
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Execute As	
Assembly	

### Parameters

Name	Data Type	Length	Description
@request_id	int	4	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER PROCEDURE dbo._set_request_approve @request_id INT
AS
    SET NOCOUNT ON;

    DECLARE @report_id INT, @number varchar(50), @msg varchar(MAX)

    SELECT @msg='Сформирован заказ-наряд на заказ '+isnull(r.number,'') FROM request r WHERE id=@request_id

    --генерим заказ наряд
    EXEC _generate_order_report @request_id, @report_id OUTPUT

    EXEC _send_message @msg
        ,0---зеленый
        ,@report_id

    --ставит у заказа статус
    UPDATE request
    SET state = 'IN_PRODUCTION'
    WHERE id = @request_id;

    --проставляем у каждого станка статус

    UPDATE machines
    SET current_status = 'WORKING'
        ,product_id = S.product_id
    FROM (SELECT
        o.label
        ,i.product_id
    FROM items i
    LEFT JOIN orders o
    ON o.item_id = i.id
    WHERE i.request_id = @request_id) S
    WHERE S.label = machines.label

    --и рапортуем, что в тот рест тоже надо статусы проставить
    SELECT
        o.label
        ,m.port
        ,i.product_id
    
```

```
FROM items i
LEFT JOIN orders o
ON o.item_id = i.id
LEFT JOIN machines m
ON o.label = m.label
WHERE i.request_id = @request_id
GO
```

## Depends On ⑥

-  dbo.request
-  dbo.\_generate\_order\_report
-  dbo.\_send\_message
-  dbo.machines
-  dbo.items
-  dbo.orders

## Used By

No items found

## dbo.email\_sended

### Description

### Properties

Name	Value
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Execute As	
Assembly	

### Parameters

Name	Data Type	Length	Description
@id_claim_message	int	4	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER PROCEDURE dbo.email_sended
@id_claim_message INT
AS
UPDATE messages
SET
    send = 1
WHERE id = @id_claim_message
GO

```

### Depends On ①



### Used By

No items found

## dbo.email\_to\_send\_get

### Description

### Properties

Name	Value
ANSI Nulls ON	True
Quoted Identifier ON	True
Encrypted	False
Execute As	
Assembly	

### SQL Script

```

SET QUOTED_IDENTIFIER, ANSI_NULLS ON
GO
CREATE OR ALTER PROCEDURE dbo.email_to_send_get
AS
BEGIN

--ПОТОМ раскоментить
SELECT
    m.msg AS text_message
    ,au.email AS email
    ,'Сообщение от системы AUMANU' AS header_message
    ,m.id AS id_claim_message
FROM messages m
    ,auth_user au
WHERE ISNULL(m.send, 0) = 0
AND m.type <> 1

END
GO

```

### Depends On 2

-  dbo.messages
-  dbo.auth\_user

### Used By

No items found