



Redis

# Какие возможности у Redis



1. Хранение и управление структурированными данными
2. Высокая производительность благодаря встроенной индексации и кэшированию
3. Поддержка множества различных типов данных, включая строки, списки, множества, хэши и т.д.
4. Возможность репликации и шардирования для обеспечения масштабируемости. (Вспоминаем прошлое занятие)
5. Поддержка транзакций и публикации/подписки для обеспечения надежности и согласованности данных.

# Установка Redis

Установка Redis не поддерживается из коробки для Windows. [Releases · tporadowski/redis\(github.com\)](https://github.com/tporadowski/redis/releases)

Есть специальные сборки для windows.

The screenshot shows the GitHub interface for the repository `tporadowski / redis`. The top navigation bar includes links for Code, Issues (69), Pull requests (1), Discussions, Actions, Projects, Wiki, Security, and Insights. Below this, the 'Releases' tab is selected, and a search bar 'Find a release' is visible. The main content area displays the release 'Redis for Windows 5.0.14.1' with a 'Latest' badge. The release description states: 'This is a bugfix/maintenance release that works around issue #130 related to usage of modules during asynchronous save operations. If you are not using modules there is no need to upgrade.' Below the description, there are four assets listed in a table:

Asset	Size	Published
<a href="#">Redis-x64-5.0.14.1.msi</a>	6.79 MB	Feb 17, 2022
<a href="#">Redis-x64-5.0.14.1.zip</a>	12 MB	Feb 17, 2022
<a href="#">Source code (zip)</a>		Feb 17, 2022
<a href="#">Source code (tar.gz)</a>		Feb 17, 2022

At the bottom of the release section, there are reaction icons (thumbs up, thumbs down, heart, etc.) and a count of '596 people reacted'.

Активация Windows  
Чтобы активировать Windows, перейдите в раздел

# Подключение к Redis

После установки Redis есть возможность его запустить

```
Microsoft Windows [Version 10.0.19045.4291]  
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены  
  
C:\Users\danil>redis-cli  
127.0.0.1:6379>
```

# Pre Redis commander

При установке redis commander необходимо установить npm/node [Setting Up npm on Windows | Step-by-Step Guide \(ioflood.com\)](https://ioflood.com/Setting-Up-npm-on-Windows-Step-by-Step-Guide)

## Step 1: Download the Node.js Installer

Begin by navigating to the [official Node.js website](https://nodejs.org/) from your Windows browser. Here, you'll be greeted with two versions of Node.js for download: LTS (Long Term Support) and Current. For most users, especially beginners, selecting the LTS version is recommended as it's more stable.

## Step 2: Run the Installer

Once the download is complete, locate the installer file (typically named `node-vxx.xx.x-x64.msi` for 64-bit systems) and double-click to initiate the installation process. The setup wizard will guide you through several steps:

1. Accept the license agreement.
2. Choose the installation location.
3. Select components to install (ensure npm is checked).
4. Configure tools (optional).
5. Install.

## Step 3: Verify Installation

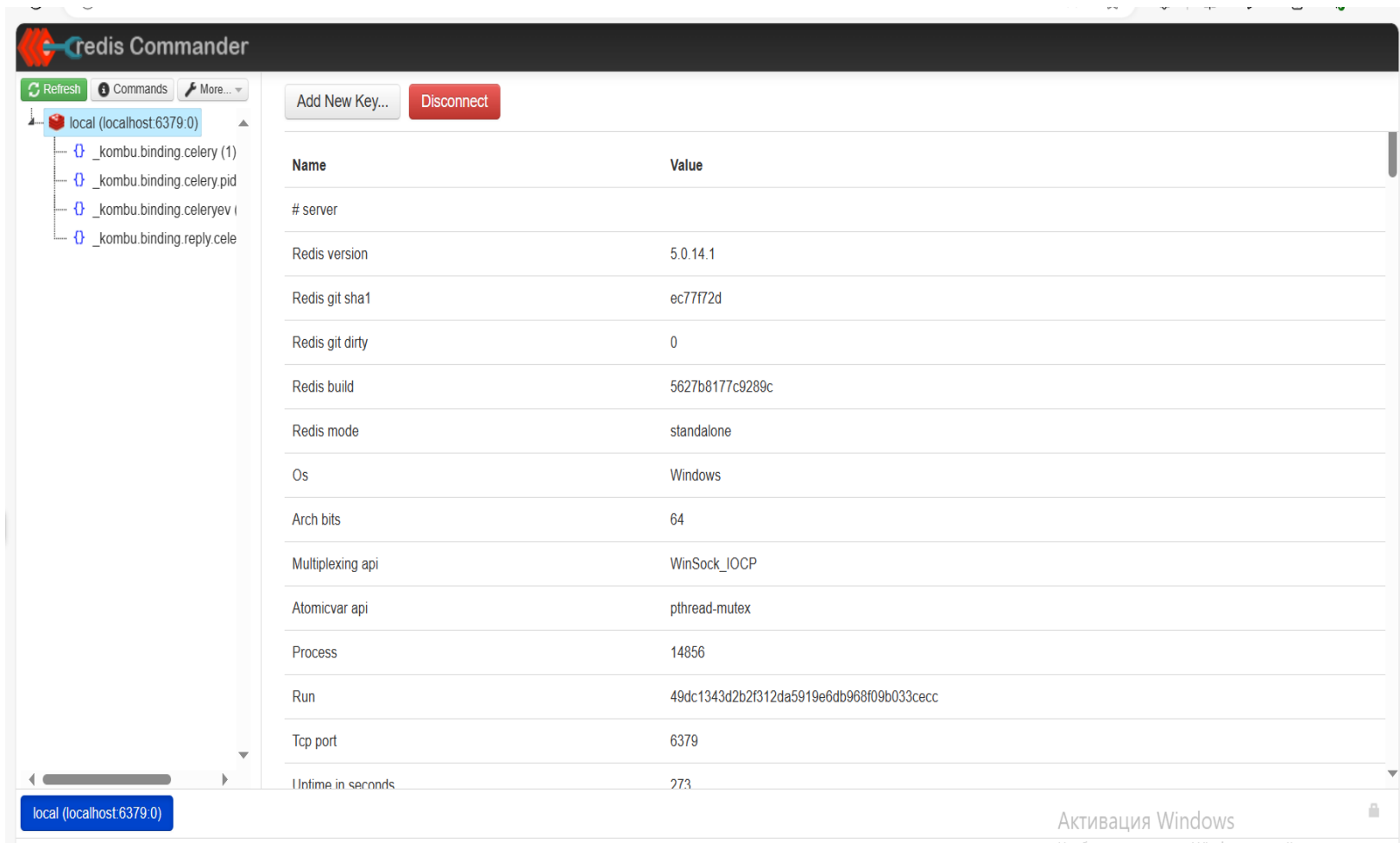
After installation, it's crucial to verify that Node.js and npm were successfully installed. Open a new command prompt or PowerShell window and enter the following commands:

```
1 | node -v
2 | npm -v
3 |
4 | # Output:
5 | # v14.17.0
6 | # 6.14.13
```

The output will display the installed versions of Node.js and npm, respectively. Seeing these version numbers confirms a successful installation.

# Redis commander

Позволяет смотреть какие данные хранятся и управлять ими, установить можно с помощью прм [Redis Commander | webinstall.dev](https://redis-commander.github.io/docs/installation-and-usage/webinstall.dev)



The screenshot displays the Redis Commander web interface. On the left, a sidebar shows a tree view of the local Redis instance (localhost:6379:0) with several keys listed: `_kombu.binding.celery (1)`, `_kombu.binding.celery.pid`, `_kombu.binding.celeryev (1)`, and `_kombu.binding.reply.celery`. The main panel shows the configuration of the selected instance. At the top, there are buttons for 'Add New Key...' and 'Disconnect'. Below this is a table with two columns: 'Name' and 'Value'.

Name	Value
# server	
Redis version	5.0.14.1
Redis git sha1	ec77f72d
Redis git dirty	0
Redis build	5627b8177c9289c
Redis mode	standalone
Os	Windows
Arch bits	64
Multiplexing api	WinSock_IOCP
Atomicvar api	pthread-mutex
Process	14856
Run	49dc1343d2b2f312da5919e6db968f09b033cecc
Tcp port	6379
Uptime in seconds	273

At the bottom of the interface, there is a status bar showing 'local (localhost:6379:0)' and a Windows watermark for 'Активация Windows'.

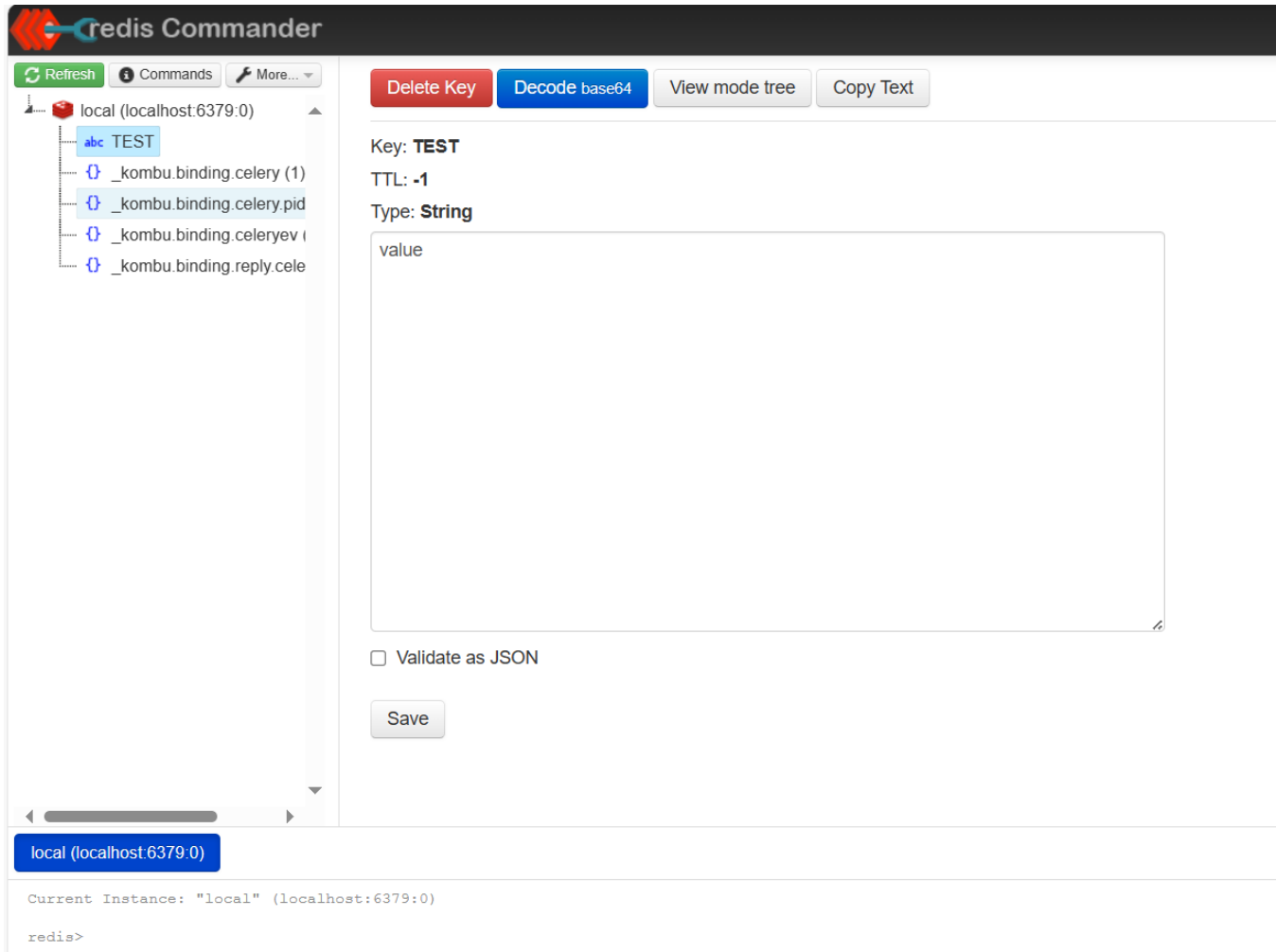
# Кеширование данных

Redis предоставляет возможности для кеширования данных, что позволяет значительно увеличить производительность приложений.

Кеширование в Redis осуществляется с помощью структуры данных под названием "ключ-значение", что делает его очень быстрым и эффективным кешем.

```
C:\Users\daniil>redis-cli
127.0.0.1:6379> SET TEST 'value'
OK
127.0.0.1:6379> GET TEST
"value"
127.0.0.1:6379>
```

# Просмотр данных в Commander



The screenshot displays the Redis Commander web interface. The top header shows the Redis logo and the text "redis Commander". Below the header, there is a navigation bar with a "Refresh" button, a "Commands" dropdown, and a "More..." button. The main content area is divided into two panels. The left panel shows a tree view of the Redis instance "local (localhost:6379:0)". The tree view includes a key "TEST" and several sub-keys: "\_kombu.binding.celery (1)", "\_kombu.binding.celery.pid", "\_kombu.binding.celeryev", and "\_kombu.binding.reply.celery". The right panel displays the details for the selected key "TEST". It shows the key name "TEST", the TTL "-1", and the type "String". Below this, there is a large text area labeled "value" for entering the key's value. At the bottom of the right panel, there is a checkbox labeled "Validate as JSON" and a "Save" button. The bottom of the interface shows a terminal window with the text "Current Instance: 'local' (localhost:6379:0)" and a prompt "redis>".

redis Commander

Refresh Commands More...

local (localhost:6379:0)

- TEST
- \_kombu.binding.celery (1)
- \_kombu.binding.celery.pid
- \_kombu.binding.celeryev
- \_kombu.binding.reply.celery

Delete Key Decode base64 View mode tree Copy Text

Key: **TEST**  
TTL: -1  
Type: **String**

value

☐ Validate as JSON

Save

local (localhost:6379:0)

Current Instance: "local" (localhost:6379:0)

redis>



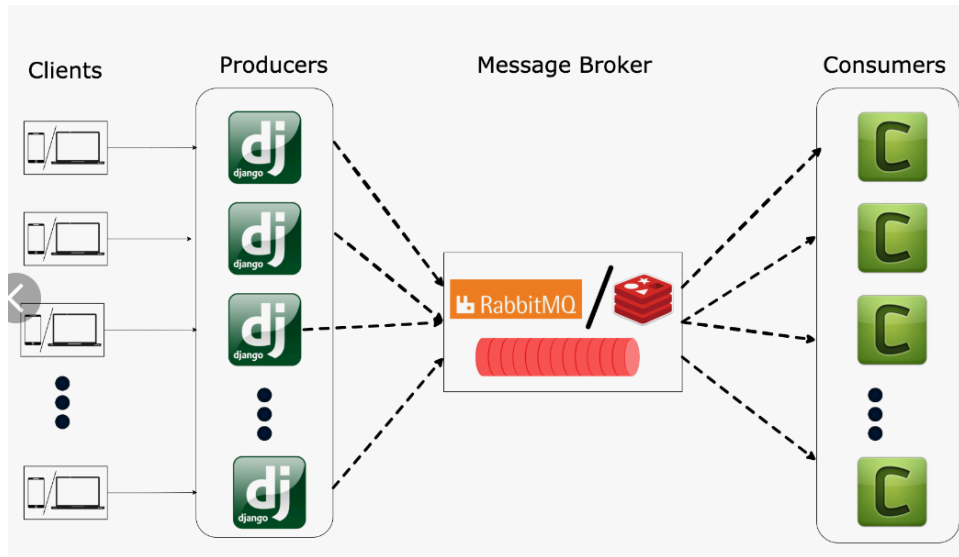
# Работа с redis в Python

Redis имеет адаптер для языка python, ссылка на пакет: [redis · PyPI](#),  
Пакет для использования ниже

## Basic Example

```
>>> import redis
>>> r = redis.Redis(host='localhost', port=6379, db=0)
>>> r.set('foo', 'bar')
True
>>> r.get('foo')
b'bar'
```

# Redis как брокер сообщений



**Producer** (производитель) - это компонент, который генерирует задачи и отправляет их в очередь для обработки. В контексте Celery и Redis, producer может быть любым приложением или скриптом, которое создает и отправляет задачи в очередь Celery.

**Consumer** (потребитель) - это компонент, который забирает задачи из очереди и выполняет их. В случае Celery и Redis, consumer - это воркер, запущенный с помощью Celery, который подключается к брокеру (Redis) и выполняет задачи, полученные из очереди.

**Broker** (брокер) - это посредник между producer и consumer, который хранит задачи в очереди и передает их воркерам для выполнения. В случае Celery и Redis, Redis выступает в роли брокера, храня задачи в виде сообщений в очереди.