**DATABASE MANAGEMENT SYSTEMS**

**PROJECT-REPORT**

**PROPERTY-ALLSTAR-WEBSITE**

**-BUY,SELL,PEDICT PRICE**

**EFFORTS BY:-**

**RAKSHIT MALHOTRA-17BCE0941**

**AGRIM NAUTIYAL-17BCE2046**

**ROHIT CHORGHE-17BCE2264**

**SIMRAN KOUL-17BCE2210**

**SOFTWARES USED**

In the implementation of the project two major softwares were used. We have made used for python’s Flask framework

for the purpose of writing the backend code for our website.

In addition to this various libraries like numpy, render\_template,

request , url\_for have been used for the implementation.

The database used by us is REDIS which is a no SQL key based database and was quite challenging to work with.

For Frontend we have have made use of HTML which is hyper text markup language.

Also we have made use of CSS’s framework Bootstrap to give a relatively better look to our website.

So In we could list our softwares as:-

FOR BACKEND SUPPORT:-

* **PYTHON(FLASK)**
* **REDIS(NO-SQL)**

FOR FRONTEND :-

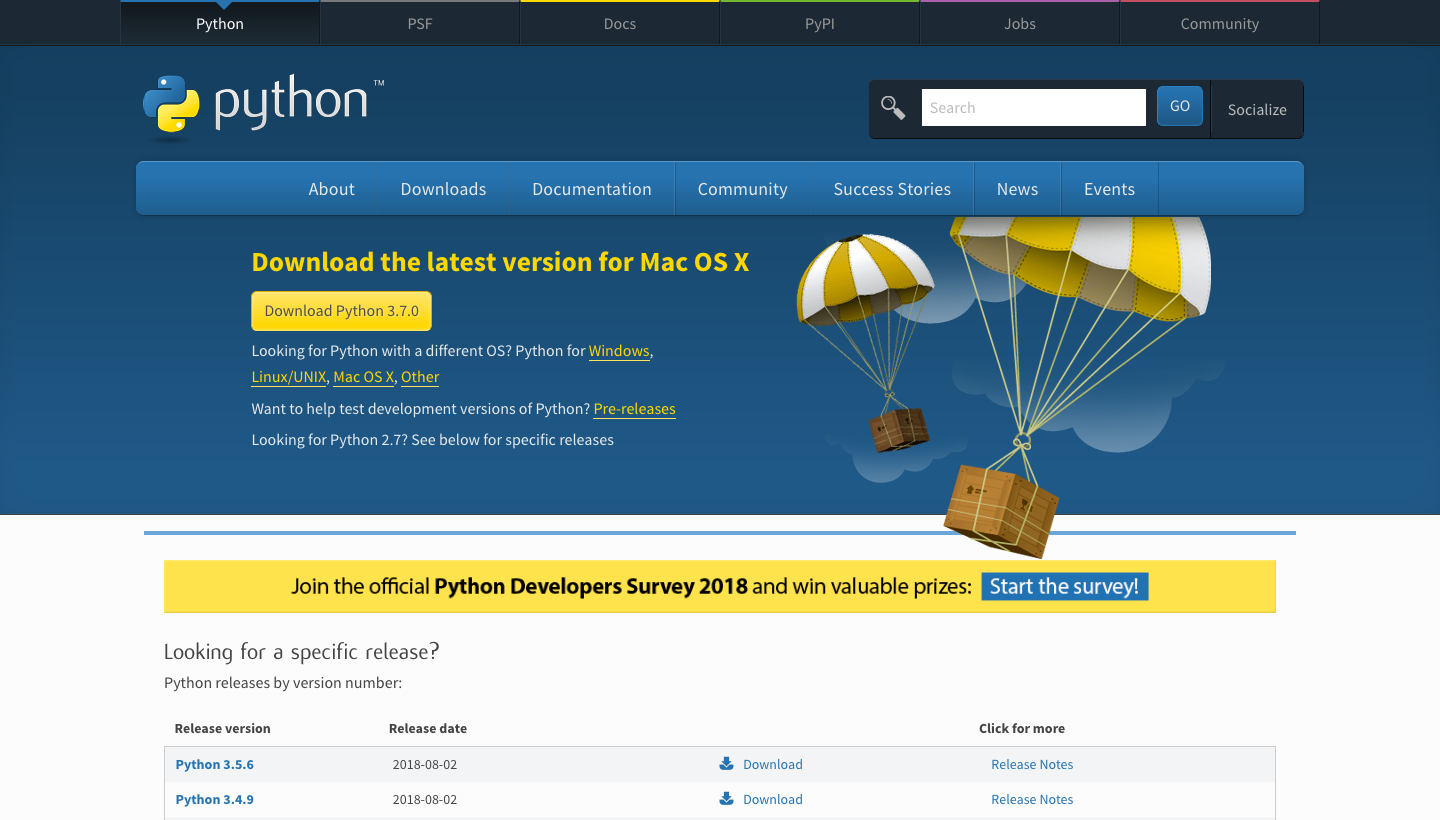
* **HTML**
* **BOOTSTRAP**

**Installation guide for Redis , Python and Flask**

**PYTHON:-**

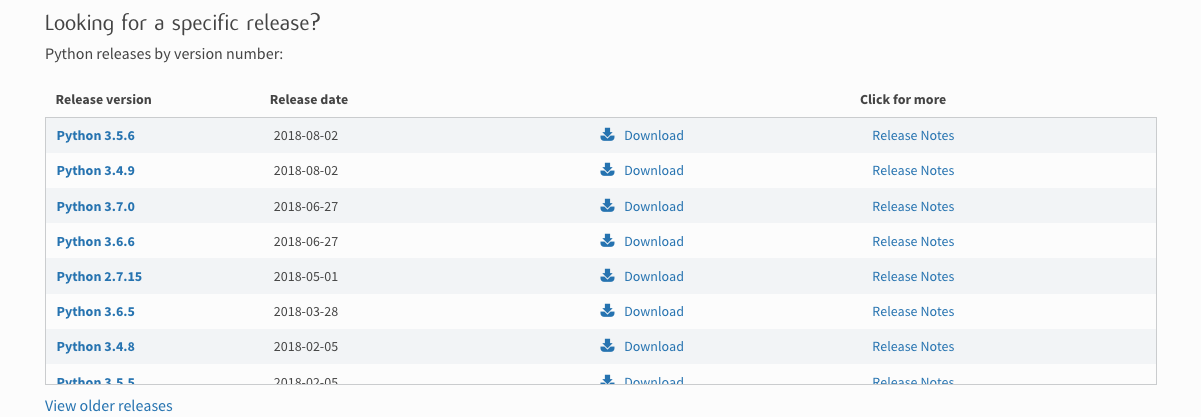
Python is generally pre-intalled but if you want to install python, go to

<https://www.python.org/downloads/>



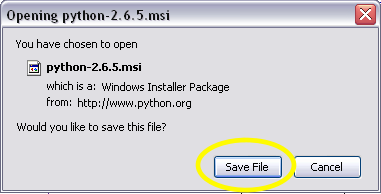
**Step 2:-**

Select your Operating system and the version you want to download and click download

****

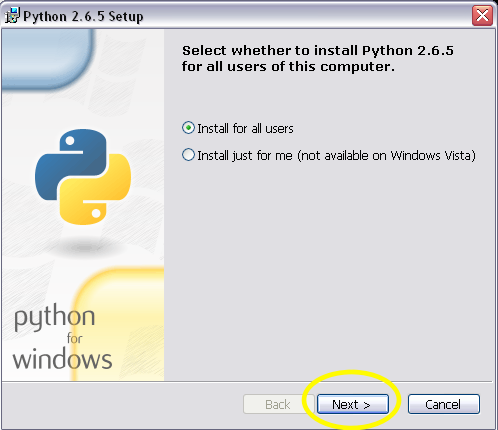
**Step 3:-**

**For windows save the file**



**Step 4:-**

**Open the Setup and keep pressing next and finish to exit the installer**

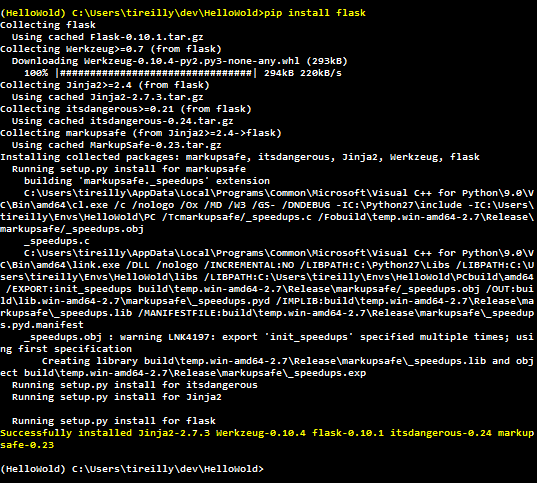




**Flask:-**

**For installing flask you may use the virtual environment or install directly using**

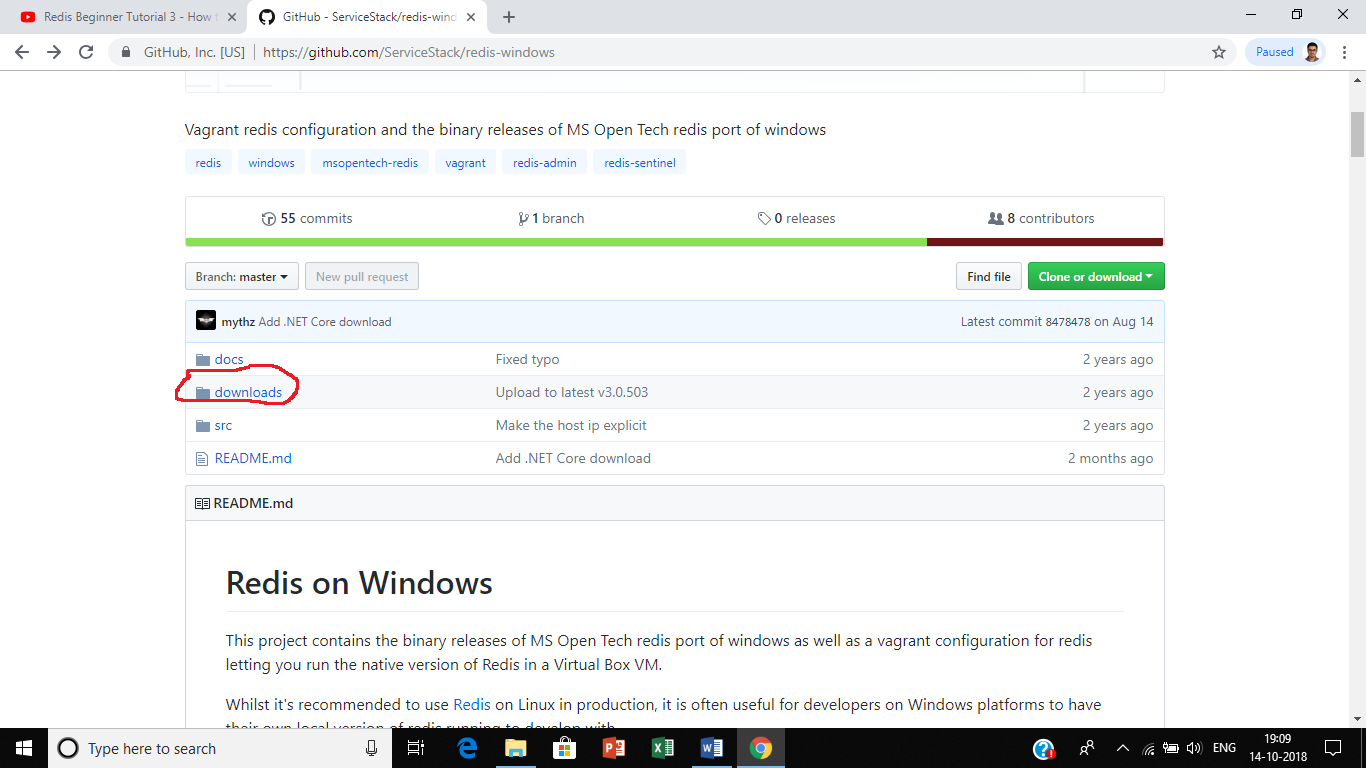
**pip install flask after your python installation is finished**



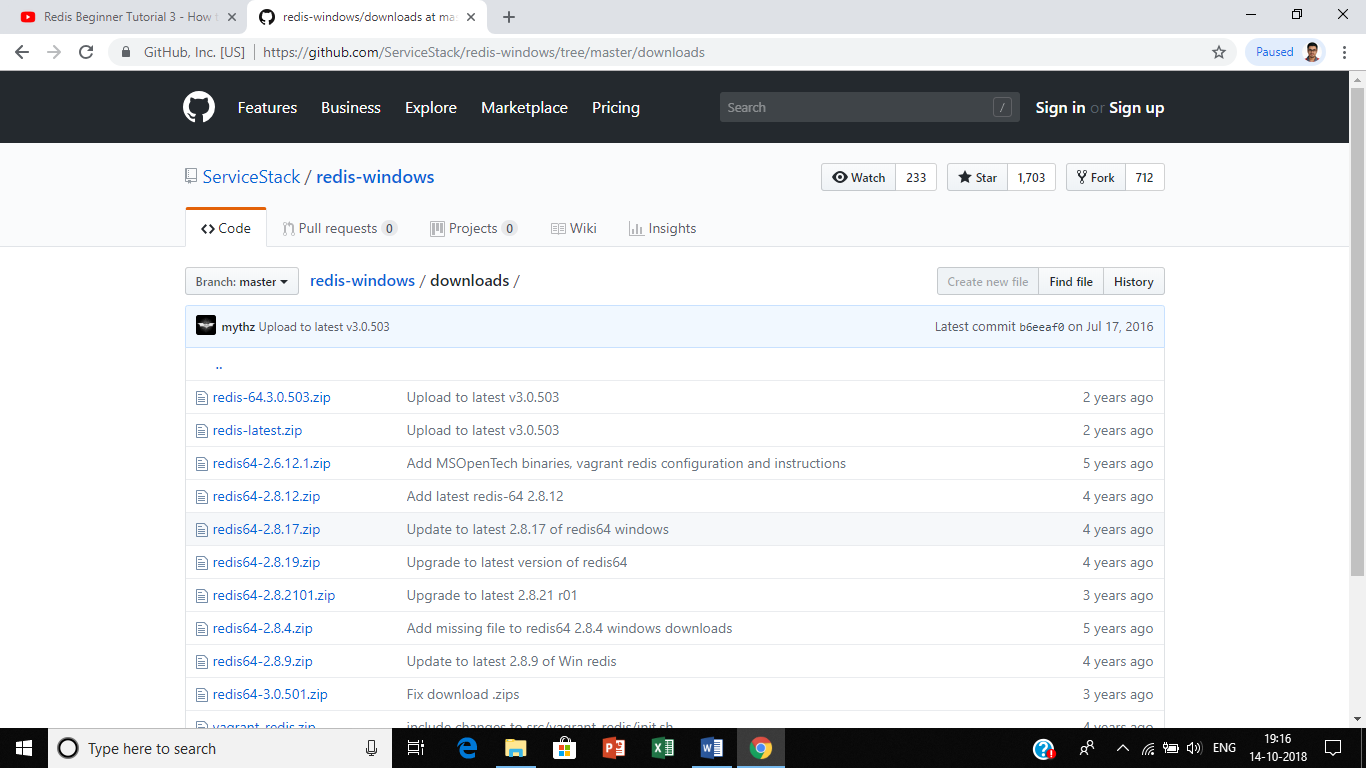
**REDIS:-**

**Steps to install:**

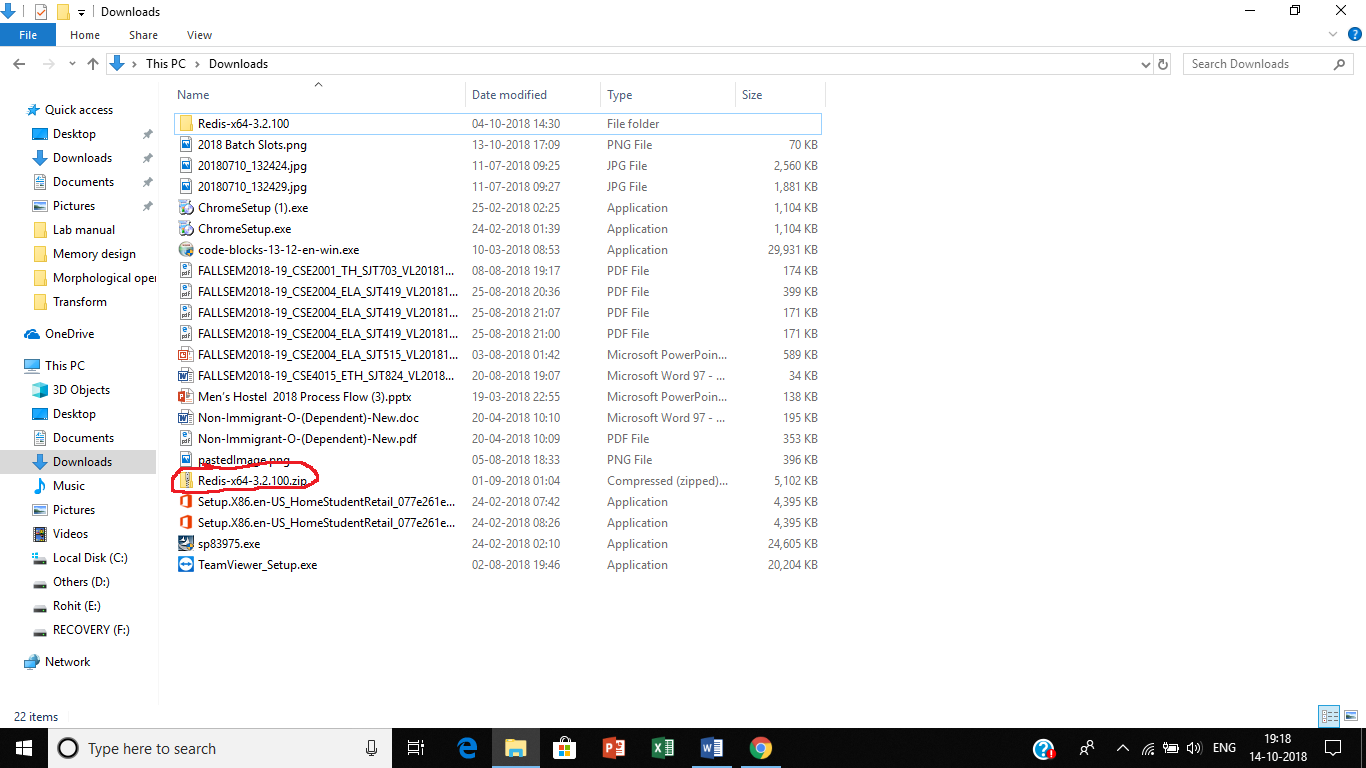
1. [**https://github.com/MicrosoftArchive/redis/releases**](https://github.com/MicrosoftArchive/redis/releases)
2. **A Github page will open, where we have to select the downloads option**



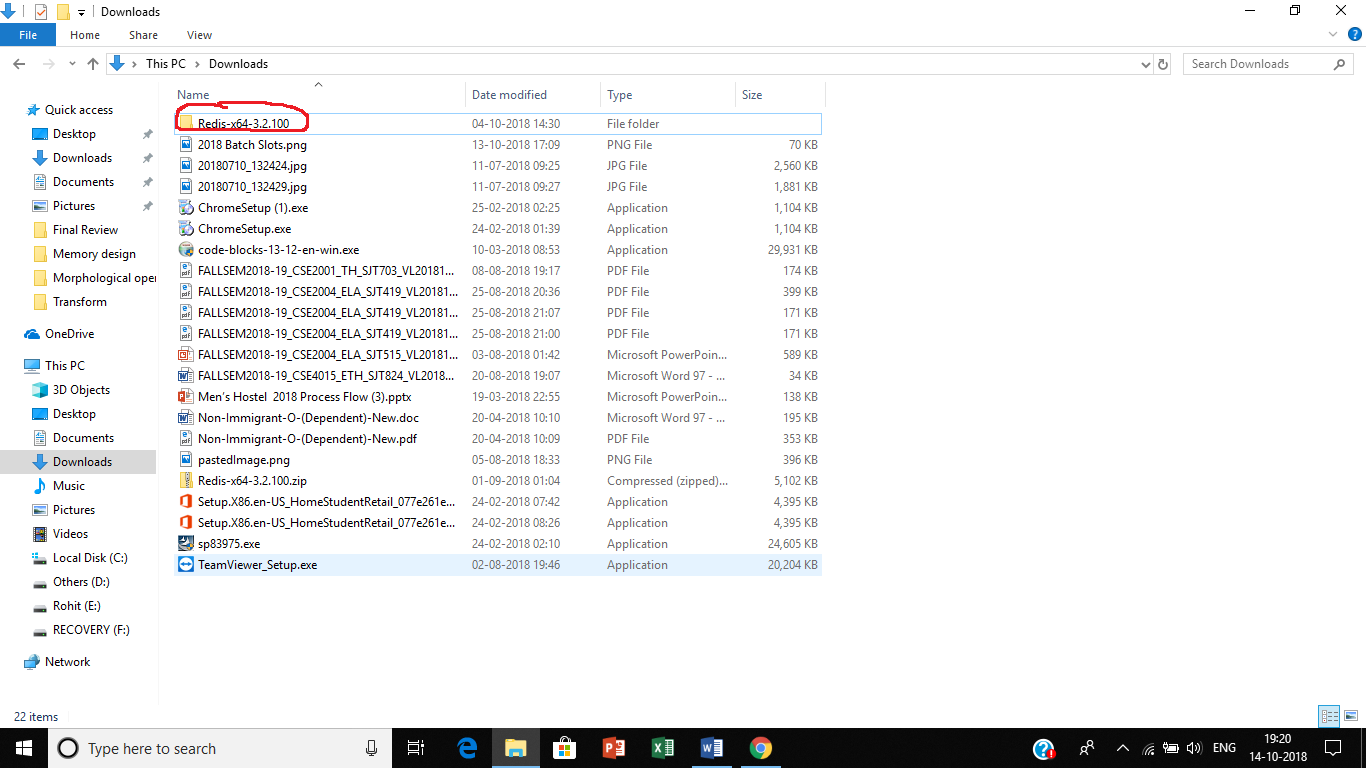
1. **On selecting the downloads option, you will get a list of redis zip files according to their versions**
2. **Select the redis-x64-3.2.100 zip file**

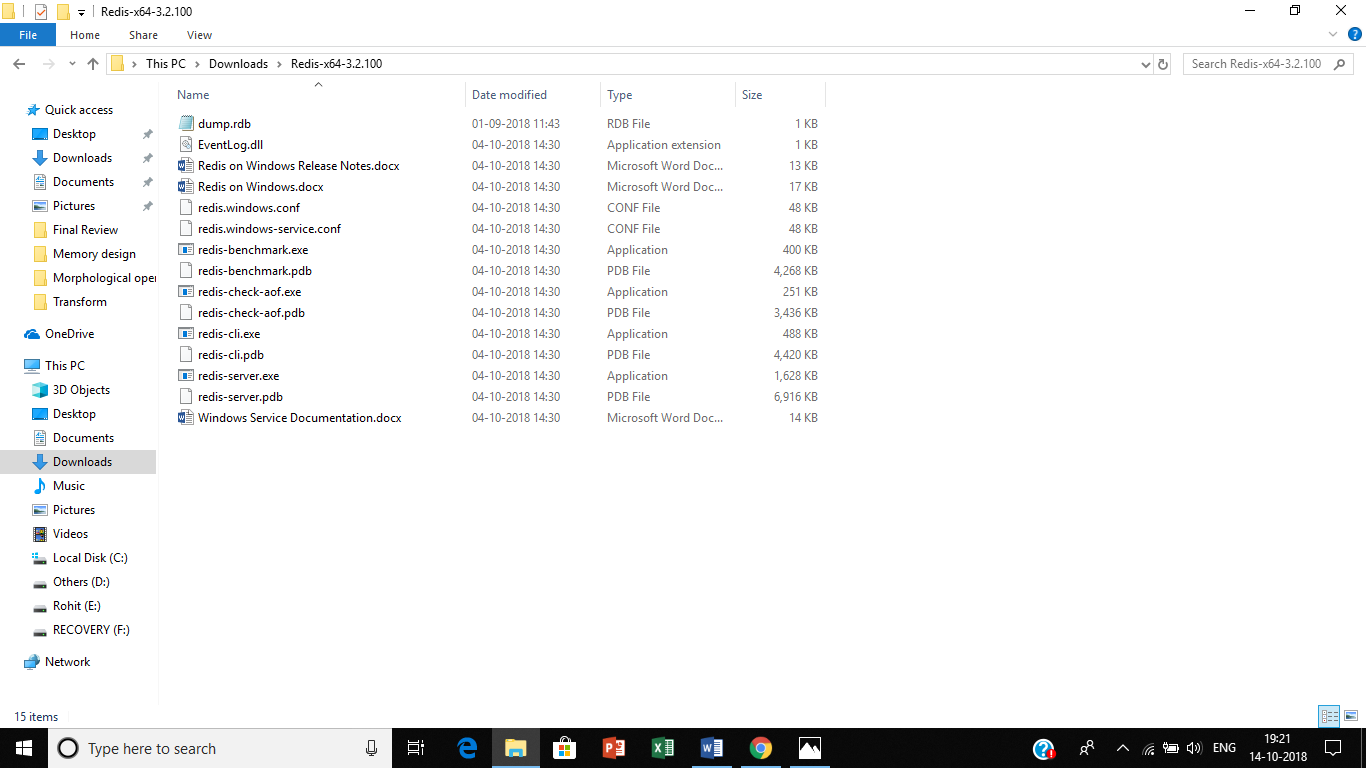


1. **Once downloaded, the zip file would look like this**

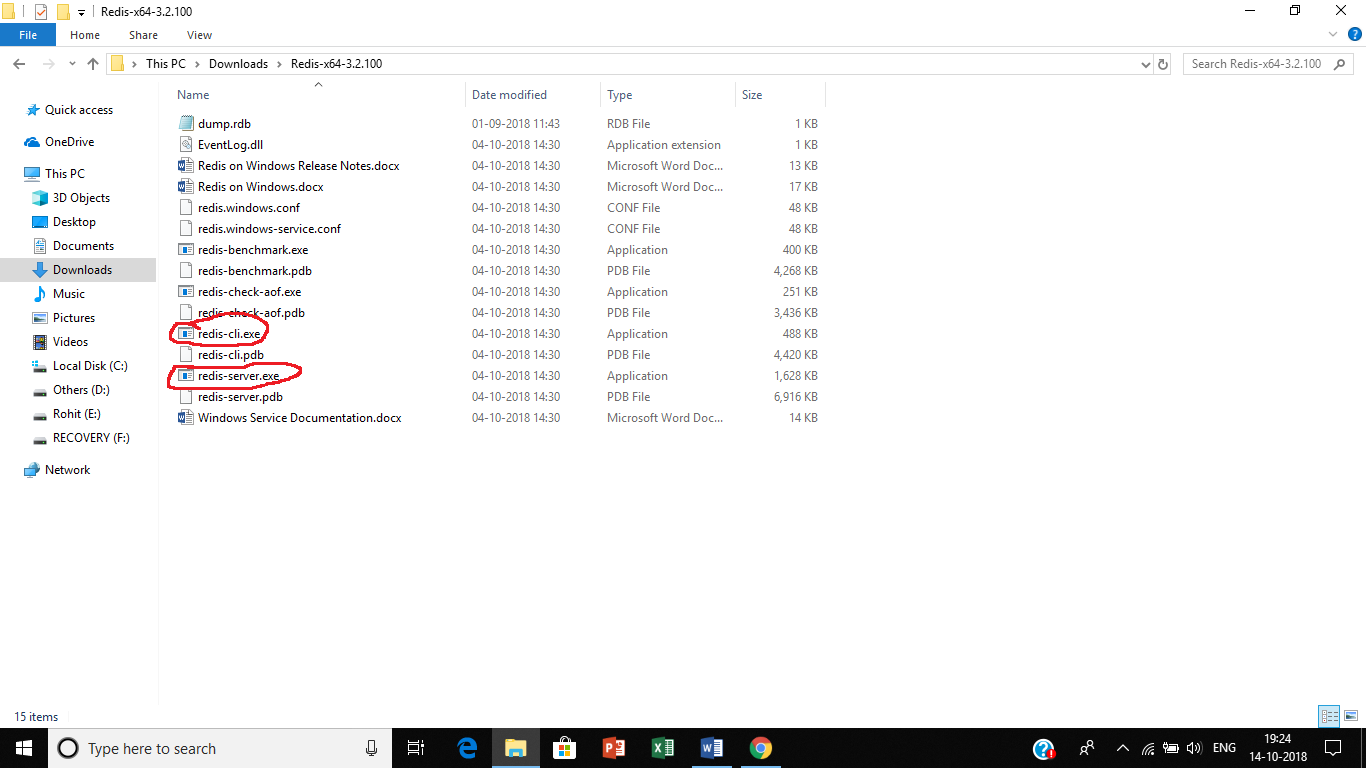


1. **After unzipping, the file would look like**

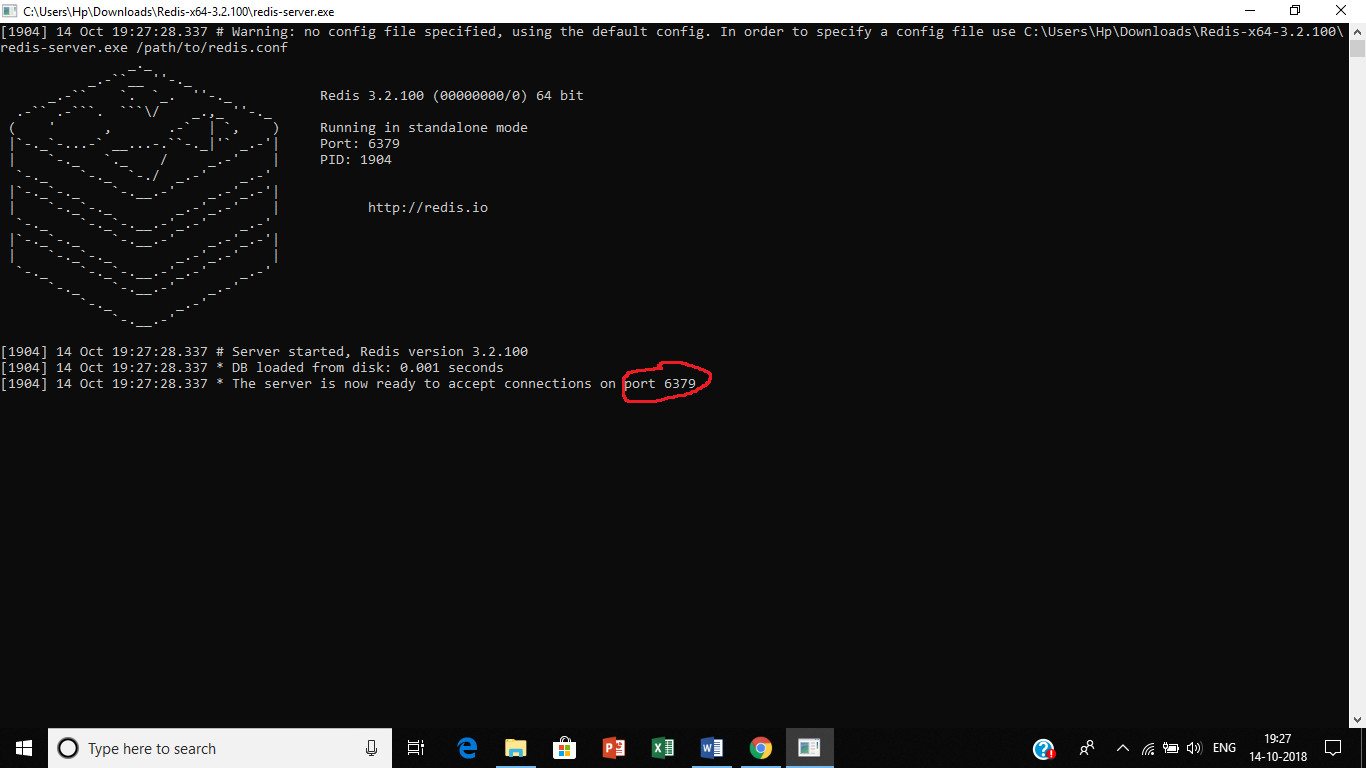




1. **Once you open the file, open the redis-server.exe and the redis-cli.exe files**

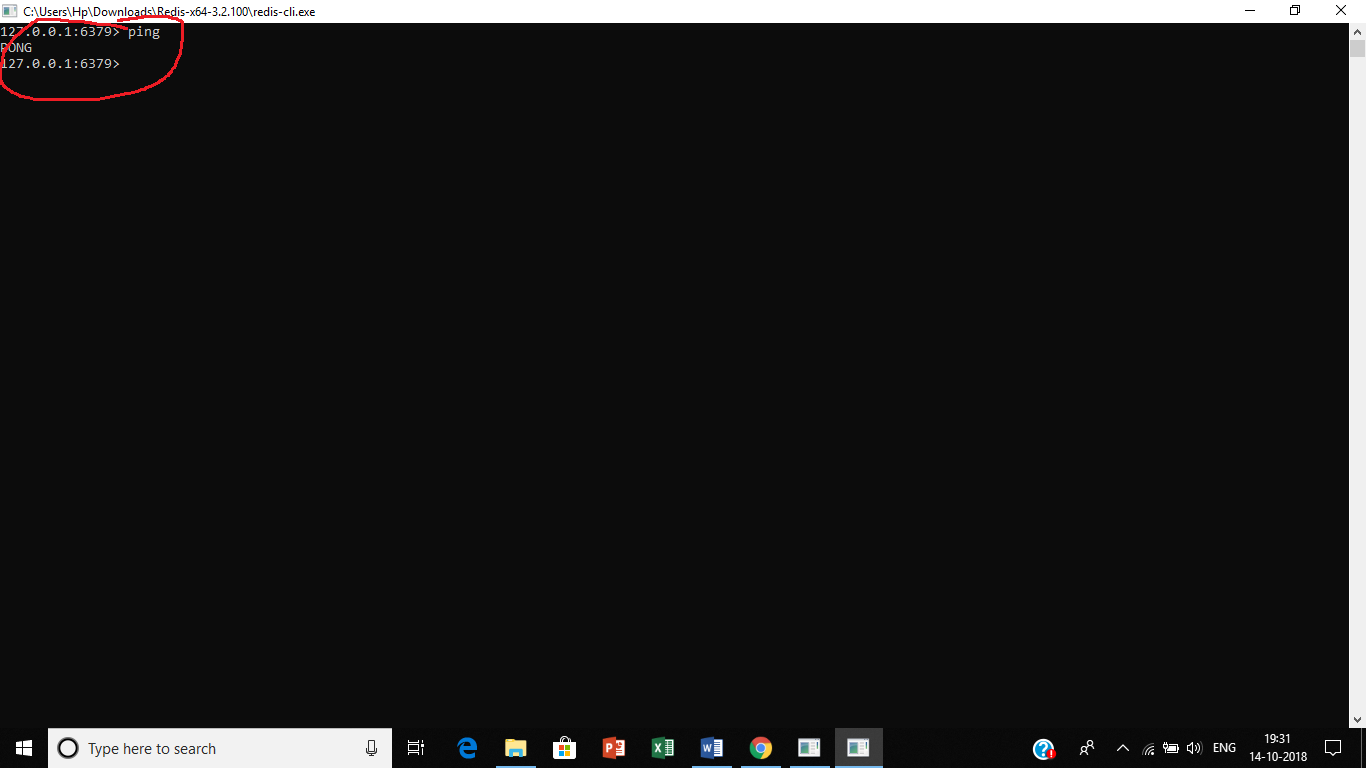


1. **The redis-server.exe file is the server file which looks like**



**Here port 6379 is the default port on the system. It can be changed if the user wants to**

1. **The redis-cli.exe file is the system where the commands are given and it looks like**



**In order to check the interactibility of the system type PING, of you get the reply PONG from the system, then the system is active**

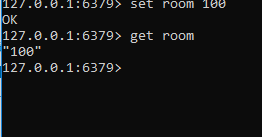
**Characteristics of our nosql database (Redis)**

**SQL**

According to the sql database that we studied in the course, data was stored in form of tables. Each table represented an entity in the entire data schema. Each column of the table represented an attribute related to that entity and each row consisted of a record. However in order to form a table one had to specify the type and length of each type of data in each row explicitly. This was time consuming and had to force the user on his end to remember the complex commands by heart in order to perform a successful query. Moreover tables were only useful when we were dealing with data of finite quantity and simpler relationships among attributes. In these days of vast and almost infinite amount of data flowing each second it is inappropriate to store data in form of tables.

**Redis**

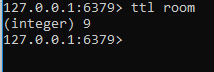
Redis has a huge advantage over the traditional method database management method. Redis belongs to what we call as a nosql database. The word **nosql** means **“Not Only Sql”** which means that the data is stored in a format different to the tabular method in sql. The data in redis is stored in a format of **key-value** pair which means that a key is assigned to a data value and the data is referred by the key similar to a linked list in data structures . An example can be shown as:



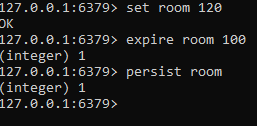
In this example a key room has been associated with a value 100. Furthermore, the data 100 is referred by the key ***room.*** Another observation made here is the fact that I didn’t have to mention the data type for the number 100. The system takes it on it’s own. This is another advantage over the sql database. We can even set the time limit for the key-value pair to expire if it’s use is temporary. For ex:



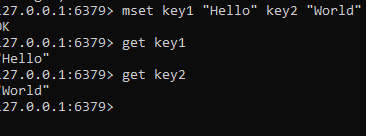
Here 100 represents the time left (in s) for the key **room** to expire. We can check the status of the key as:



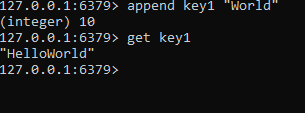
Here 9 represents time (in s) left for the key **room** to expire. If you don’t want to delete the key in the middle of it’s expiry then:



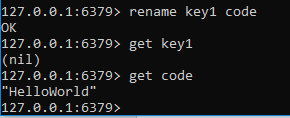
If you want to name more than one keys at the same time then:



If you want to add data to the value of a certain key then:



If you want to rename a key then:

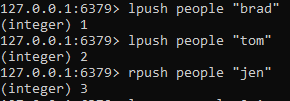


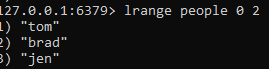
**The various data types in redis are:**

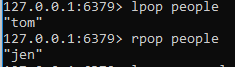
1. **Lists**
2. **Sets**
3. **Sorted Sets**
4. **Hashes**

**Lists**

Lists are basically data types which are similar to an array. The difference between a list and the other data types in redis is that the addition and deletion of data in a list is bi-directional. We can add/delete data from the right as well as from the left. For ex:

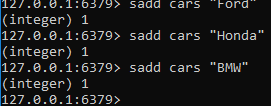






**Sets**

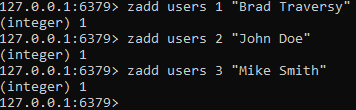
Sets is a form of a data type similar to key-value pair. It’s general form can be shown as:



Here **cars**  represents the field and **Ford,Honda,BMW**  are the data values associated with **cars**

**Sorted Sets**

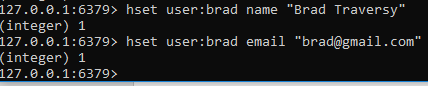
Sorted sets are similar to sets but the difference in sorted sets is that each value is associated with a score. This score helps in giving priority to a particular record. For ex:



Here **zadd** is the command, **users** is the field, **(1,2,3)**  are the scores and (**Brad, John and Mike)** are the values.

**Hashes**

Hashes are the most commonly used data types in redis. This is mainly because each record of a hash has a hash-key which uniquely determines a record and acts more so as a primary key in the case of sql database. The data format is a shown:



In this example **user:brad** is a hash-key, **(name, email)** are field names and the last entities are values.

***DEMO VEDIO FOR OUR PROJECT IS AVAILABLE ON YOUTUBE IN THIS PRIVATE LINK:-***

[**https://youtu.be/38clq2UDJD4**](https://youtu.be/38clq2UDJD4)

**NOTE:-Kindly open in Firefox for better video quality**