

>



what means this command?

```
docker build . -t qdos-devel:latest
```



This command is used to build a Docker image using the Docker build system. Let's break down the command:

1. **docker build**: This is the main command to build Docker images from a Dockerfile and a specified "context" (typically a directory containing the files and resources needed to build the image).
2. **.**: This is the build context, which is the current directory. The Dockerfile should be located in this directory, and any other files and resources required during the build process should also be here or within its subdirectories.
3. **-t**: This flag is used to specify a name and optionally a tag for the image. It stands for "tag."
4. **qdos-devel:latest**: This is the name and tag for the image you are building. **qdos-devel** is the name of the image, and **latest** is the tag. If you don't provide a tag, **latest** is used by default.

In summary, this command builds a Docker image using the Dockerfile and context found in the current directory and tags the resulting image with the name "qdos-devel" and the tag "latest."



what means this?

```
docker pull xora/qdos-devel
```



The **docker pull** command is used to download a Docker image from a remote repository, such as Docker Hub, to your local machine. In this case, the command is pulling the Docker image named **xora/qdos-devel**.

Let's break down the command:

1. **docker pull**: This is the main command to download a Docker image from a remote repository.
2. **xora/qdos-devel**: This is the name of the Docker image you want to download. It is composed of two parts:
 - xora**: This is the user or organization account on the remote repository (such as Docker Hub) that owns the image.
 - qdos-devel**: This is the name of the Docker image you want to download.

By default, if no specific tag is mentioned, the **latest** tag will be pulled. In this case, the command will download the **xora/qdos-devel:latest** image from the remote repository to your local machine.



so, previous build command is not required if I am downloading existing image? where it is stored in docker?



That's correct. If you are downloading an existing Docker image using the **docker pull** command, you don't need to build the image yourself with the **docker build** command. The image you pull from the remote repository will already be built and ready for use.

When you pull a Docker image, it is stored in your local Docker environment, specifically in the local Docker image registry. You can view all the images you have downloaded or built locally by running the following command:

Copy code

```
docker images
```

This command will list all the images you have in your local registry, along with their repository name, tag, image ID, and the date they were created.

Keep in mind that if you need to make any modifications or customizations to the image, you will need to have access to the Dockerfile and build the image using the **docker build** command. But if you're using an existing image that meets your requirements, simply pulling the image is sufficient.



how much commodore plus/4 was sold?



The Commodore Plus/4, released in 1984, was not as successful as its popular sibling, the Commodore 64. Exact sales figures for the Plus/4 are not available, but it is estimated that around 350,000 units were sold during its production run. This number pales in comparison to the Commodore 64, which sold over 17 million units and became the best-selling single personal computer model of all time. The Plus/4's relatively high price, lack of compatibility with Commodore 64 software, and competition from other computers contributed to its limited success.
