1. A common challenge that Docker can help solve is integration. When multiple developers program parts of the same application, there can frequently be issues when trying to piece all of the code together. What works one way on one machine may work differently on another, which can lead to issues when trying to integrate in to one happily working program. Docker allows developers to pass a container that holds all necessary code, which helps by ensuring all libraries or dependencies are passed from one developer to another. This prevents missing pieces that one developer may have that another doesn’t, which would lead to frustration and buggy code. DockerHub can also be used to benefit continuous integration.
2. Docker can also benefit developers with testing. Instead of the traditional test, fix, reintegrate cycle which frequently leads to more broken code elsewhere, with Docker, “when developers find bugs, they can fix them in the development environment and redeploy them to the test environment for testing and validation.” (Docker) Docker also can hold the entire testing suite as part of the image, so it is an easy transition as it is in the same spot as the rest of the code.
3. As for deployment, Docker can also benefit this step of development. Especially when using continuous deployment, there can be issues with version control and when to complete an update. Developers can use the same containers for every step of development, so any changes can be pushed directly to the production version. This allows for quick updates during the development lifecycle when need be since the entire container can be tested to ensure everything still works correctly. DockerHub can also be used to benefit continuous deployment.
4. Another aspect that Docker can help is customer support. Customer experience can decrease due to buggy code. Rather than needing to comb through all the code to find issues that customers may bring up, containers can allow easier access to related parts of the code. With well implemented containers, the developers would instead just need to look at one container instead of the entire code base for the project. Containers can also help by keeping applications running more frequently by reducing the need to take a site down to fix a problem. These problems can be solved faster and without interrupting the entire service using containers.
5. One of the most serious problems that comes with continuous integration and deployment is poor security. Many projects do not update their security frequently enough, and are left vulnerable to attacks that can compromise the entire project. Docker can help improve security because if the code has been compromised, that individual container can be replaced, rather than having to completely restart the program or spend a long time analyzing the code for the problem.

I used the suggested site (<https://dzone.com/articles/top-10-benefits-of-using-docker>) and Docker’s website (<https://docs.docker.com/get-started/overview/#:~:text=%F0%9F%94%97&text=Docker%20streamlines%20the%20development%20lifecycle,(CI%2FCD)%20workflows>) to formulate my answers.