* **What did you learn about move semantics?**

When you want to work with high performance, using memory intensively, it is very convenient to use movement operations, since they are significantly faster when working with a large amount of data.

* **In what kinds of projects do you think this would matter?**

In simulation projects, whether in video games, scientific computing, or any project that works with a large amount of data.

* **What are some risks of using std::move incorrectly?**

There are several ways to make mistakes if this operation is not used correctly. Some of these errors include:

1. Moving from an object that is still in use.
2. Accidentally moving when you intended to copy.
3. Double deletion or segmentation errors if the move logic is not implemented correctly in custom classes.