In Eclipse, when you implement the Serializable interface in a class, it provides two options for adding a serialVersionUID:

**1. Add Default serialVersionUID**

* This option assigns a fixed UID, typically 1L, to the serialVersionUID field.
* Example:
* private static final long serialVersionUID = 1L;
* It ensures that even if you make changes to the class (like adding new fields), the class can still deserialize old serialized objects without compatibility issues—unless you manually change the UID.
* Best suited when you manually control serialization compatibility.

**2. Add Generated serialVersionUID**

* This option generates a unique UID based on the class structure using Eclipse’s built-in algorithm.
* Example (generated by Eclipse):
* private static final long serialVersionUID = -1234567890123456789L;
* This UID changes if you modify the class (e.g., add/remove fields or methods), meaning old serialized objects may become incompatible with the new version.
* Best suited when you want strict version control and avoid deserialization of outdated objects.

**Which One to Choose?**

* If your class is stable and you want **backward compatibility**, choose the **default 1L or manually define a fixed value**.
* If you want **automatic detection of breaking changes**, choose the **generated serialVersionUID**, but be aware that changes to the class will make previously serialized objects invalid.